

Research Article

Open-Closed-Loop PD Iterative Learning Control with a Variable Forgetting Factor for a Two-Wheeled Self-Balancing Mobile Robot

Jian Dong , Bin He , Chenghong Zhang, and Gang Li

College of Electronics and Information Engineering, Tongji University, Shanghai, China

Correspondence should be addressed to Bin He; hebin@tongji.edu.cn

Received 31 December 2018; Revised 21 March 2019; Accepted 24 April 2019; Published 8 August 2019

Academic Editor: Xianming Zhang

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

A novel iterative learning control (ILC) algorithm for a two-wheeled self-balancing mobile robot with time-varying, nonlinear, and strong-coupling dynamics properties is presented to resolve the trajectory tracking problem in this research. A kinematics model and dynamic model of a two-wheeled self-balancing mobile robot are deduced in this paper, and the combination of an open-closed-loop PD-ILC law and a variable forgetting factor is presented. The open-closed-loop PD-ILC algorithm adopts current and past learning items to drive the state variables and input variables, and the output variables converge to the bounded scope of their desired values. In addition, introducing a variable forgetting factor can enhance the robustness and stability of ILC. Numerous simulation and experimental data demonstrate that the proposed control scheme has better feasibility and effectiveness than the traditional control algorithm.

1. Introduction

Two-wheeled self-balancing mobile robots [1–3] are naturally unstable systems with characteristics such as nonlinear, multivariable, time-varying, and incomplete under-actuation, and these characteristics can be used to study variable control algorithms. As the abovementioned robots have advantages in terms of different aspects of strong environmental adaptation, low cost, and stable operation, these robots have many applications in many areas. Kazuo Yamafuji first introduced the concept of a two-wheeled self-balancing robot in the nineteen eighties. Since then, scholars have investigated numerous kinematic models and control algorithms for two-wheeled self-balancing mobile robots. Grasser [4] developed the DSP-based wheeled balancing robot “Joe”, which achieved good mobility through decoupling motion and simplified control but poor interference resistance. Miasa [5] designed the self-balancing robot “Equibot”, but it was only good for moving on flat ground and could fall over on uneven ground and thus has high requirements for road conditions. Additionally, the adaptability of the controlled object is undesirable. Thao [6] used the PID inverse step method to control a robot; a reverse step controller maintained

the balance of the system, and a PID controller controlled the motion direction. However, the algorithm is too complex to implement in practical applications. Wu [7] applied a state feedback controller and a fuzzy controller to a robot. Experiments showed that the combination of these two control algorithms could achieve a good control effect. Because of its small capacity, feasible rotation and strong ability to adapt to terrain changes, this combined algorithm is widely used in many fields [8–10], such as traffic, survey, rescue and entertainment. However, this method is cost prohibitive and time consuming because of its fuzzy rules.

Determining how to maintain balance and realize perfect tracking under the correlated interference condition is absolutely critical to research about two-wheeled self-balancing mobile robots. However, conventional modern control methods, including sliding mode control and intelligent control, need accurate knowledge of the robot dynamics and physical parameters of the system to ensure consistency and convergence. Thus, we introduced a novel iterative learning control (ILC) algorithm. The basic idea behind ILC [11–13] is that the controller should learn from previous executions and perform better every cycle. The idea of ILC is that a skill can be improved and subsequently perfected

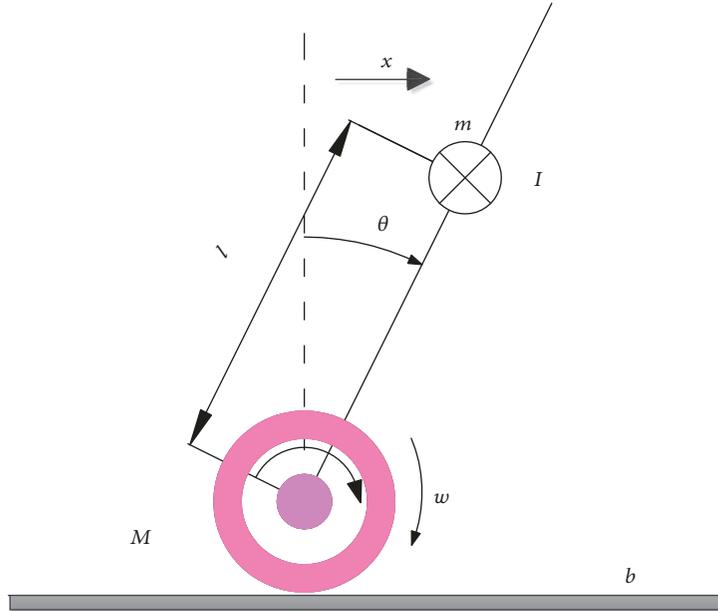


FIGURE 1: Model of a two-wheeled self-balancing mobile robot system in motion.

through constant practice. An ILC algorithm has already been designed for mobile robots in [14–16], Kyung [14] used ILC path-tracking algorithm to control mobile robot under state disturbances and state errors, but its trajectory tracking accuracy is not very ideal. Chong [17] adopted ILC with predictive, current, and past learning items to achieve high precision trajectory tracking. However, the tracking performance of this algorithm is poor in the real experimental environment with interference. There are quite a few other learning control methods. Zhou [18] design a robust observer-based learning control method with a prescribed H_∞ disturbance rejection for modified repetitive-control system to compensate the system delay. Zhou [19] presents an estimation and compensation learning control of state-dependent nonlinearity for EID-based modified repetitive-control system to improve the tracking performance.

This algorithm is extended to include the open-closed-loop PD-ILC scheme [20–22] for two-wheeled self-balancing mobile robot systems to improve the performance of the algorithm. Open-closed-loop PD-ILC not only ensures system stability but also improves tracking performance and vibration control. To reduce buffeting caused by robot system uncertainty and disturbances, this control law has been improved by the introduction of a variable forgetting factor, which can dynamically and precisely compensate for system errors. This combination not only stabilizes the robot's balance but also maintains the motion of the robot to track the desired trajectory.

The remainder of this paper is organized as follows. In Section 2, a mathematical model of a two-wheeled self-balancing mobile robot is derived from Newton's laws and Lagrangian mechanics. Section 3 describes the design of the proposed ILC controller schemes. A robust convergence analysis is performed in Section 4. In Section 5, simulations

are presented and discussed. Section 6 illustrates the experimental results for the control system. The paper is concluded in Section 7.

2. Two-Wheeled Self-Balancing Mobile Robot System Model

2.1. System Description of the Two-Wheeled Self-Balancing Mobile Robot System. As shown in Figure 1, the mechanical part of the two-wheeled self-balancing mobile robot [23–25] basically consists of two wheels and a body. The parameters of the two wheels of the robot are the same in terms of mass, diameter and rotational inertia; additionally, these wheels are coaxially installed and independently driven. The body moves in an upright fashion and maintains balance through two rounds of rotation. The fundamental equation and derivation of the robotic mathematical model use Newton's laws of motion and Lagrangian mechanics. Based on the model, an open-closed-loop controller, in combination with ILC, is deployed for equilibrium stability and drive control.

When the two wheels move at different speeds, the robot body can change its direction and rotate around its center by setting different rotation directions for the two wheels. The steering principle is exactly the same as the tank steering principle shown in Figure 2.

2.2. Dynamic Modeling of the Two-Wheeled Self-Balancing Mobile Robot System. According to Newton's second law of motion and the momentum balance, the following differential equation is obtained after force analysis of the robot body and two-wheeled bottom in the horizontal direction:

$$(M + m) \ddot{x} + b\dot{x} + ml\ddot{\theta} \cos \theta - ml\dot{\theta}^2 \sin \theta = u \quad (1)$$

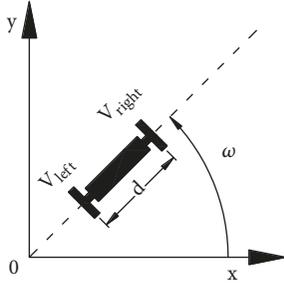


FIGURE 2: Steering diagram of the robot.

The following differential equation is obtained by analyzing the force and momentum balance of the robot body in the vertical direction:

$$(I + ml^2)\ddot{\theta} + mgl \sin \theta = -ml\ddot{x} \cos \theta \quad (2)$$

Formulas (1) and (2) are the mathematical model in the form of the system's nonlinear differential equations. Since keeping the robot in balance is the primary purpose of control under appropriate external conditions, the exact model is treated as follows:

$$\begin{aligned} \cos \theta &\approx 1, \\ \sin \theta &\approx -\theta, \end{aligned} \quad (3)$$

$$\left(\frac{d\theta}{dt}\right)^2 \approx 0$$

The linearization of the equation of motion is as follows:

$$(M + m)\ddot{x} + b\dot{x} - ml\ddot{\theta} = u \quad (4)$$

$$(I + ml^2)\ddot{\theta} - mgl\theta = ml\ddot{x} \quad (5)$$

After taking the Laplace transform [26, 27], the transfer function is as follows:

$$\begin{aligned} \frac{X(s)}{U(s)} &= \frac{((I + ml^2)/q)s^2 - mgl/q}{s^4 + (b(I + ml^2)/q)s^3 - ((M + m)mgl/q)s^2 - (bmg/l/q)s} \end{aligned} \quad (6)$$

$$\begin{aligned} \frac{\theta(s)}{U(s)} &= \frac{(ml/q)s^2}{s^4 + (b(I + ml^2)/q)s^3 - ((M + m)mgl/q)s^2 - (bmg/l/q)s} \end{aligned} \quad (7)$$

where $q = (M + m)(I + ml^2) - m^2l^2$.

According to formulas (4) and (5), the equation of state in which the displacement x , velocity \dot{x} , pitch angle θ , and angular velocity $\dot{\theta}$ of the robot are state variables is

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix}$$

$$\begin{aligned} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & \frac{-(I + ml^2)b}{I(M + m) + Mml^2} & \frac{mgl^2}{I(M + m) + Mml^2} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & \frac{-mlb}{I(M + m) + Mml^2} & \frac{mgl(M + m)}{I(M + m) + Mml^2} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} \\ &+ \begin{bmatrix} 0 \\ \frac{I + ml^2}{I(M + m) + Mml^2} \\ 0 \\ \frac{ml}{I(M + m) + Mml^2} \end{bmatrix} u \end{aligned} \quad (8)$$

The output equation with the displacement x and pitch angle θ of the robot as the output variables is

$$y = \begin{bmatrix} x \\ \theta \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} u \quad (9)$$

where x is the displacement of the robot, \dot{x} is the velocity of the robot, \ddot{x} is the acceleration of the robot, θ is the angle between the robot body and the vertical upward direction, u is the input, M is the mass of the two-wheeled bottom part of the robot, m is the upper body mass of the robot, b is the coefficient of friction between the robot's bottom wheel and the ground, l is the linear distance between the axis of the robot's bottom wheel and the center of its body mass, and I is the upper body inertia of the robot.

Formulas (8) and (9) can simplify the system state equation as follows:

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx + Du \end{aligned} \quad (10)$$

where A is the state matrix, B is the input matrix, C is the output matrix, and $D = 0$ is the feedforward matrix.

3. Designed Controller and Variable Forgetting Factor

The designed control algorithm, which includes the tracking error of i previous iterations and $i + 1$ current iterations, as shown in Figure 3, is

$$\begin{aligned} u_{i+1}(t) &= a(i, t)u_0(t) + (1 - a(i, t))u_i(t) \\ &+ \varphi_1(t)\dot{e}_i(t) + \varphi_2(t)\dot{e}_{i+1}(t) + \Psi_1(t)e_i(t) \\ &+ \Psi_2(t)e_{i+1}(t) \end{aligned} \quad (11)$$

where i indicates the iteration number; $\varphi_1(t)$ is the open-loop differential learning gain matrix; $\varphi_2(t)$ is the closed-loop differential learning gain matrix; $\Psi_1(t)$ is the open-loop

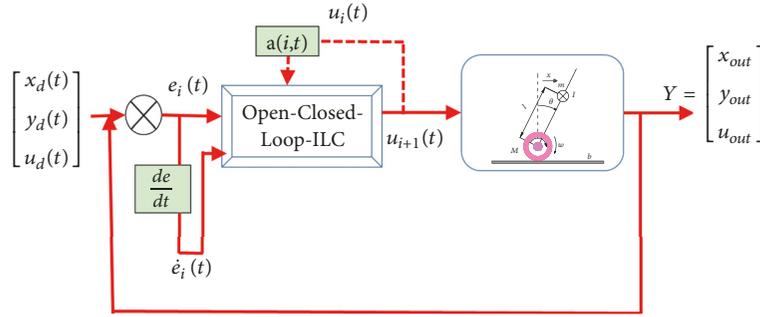


FIGURE 3: Block diagram of the designed ILC law.

proportional learning gain matrix; $\Psi_2(t)$ is the closed-loop proportional learning gain matrix; $u_{i+1}(t)$ and $u_i(t)$ are the control variable; $u_0(t)$ is the initial value of the input; and $e_i(t) = y_d(t) - y_i(t)$.

Scholars generally use a fixed forgetting factor, but a cannot vary with changes in the system features. In this article, $0 \leq a(i, t) < 1$ is a multivariable forgetting factor that is introduced to the designed controller. This multivariable forgetting factor $a(i, t)$ can vary automatically according to changing system deviation. The factor $a(i, t)$ is used to balance perfect learning and robustness, which can increase the robustness of ILC against uncertainty, disturbance, initialization error, and fluctuation of system dynamics:

$$a(i, t) = \frac{(1 - \delta)}{5^{\max[\Lambda e_i^2(t)]}} \quad (12)$$

where Λ coordinates gain changes and Λ is used to control the rate at which δ approaches 1. The smaller the value of $a(i, t)$, the better the system tracking performance.

4. Convergence Analysis

Assumption 1. Function f is uniformly Lipschitz with respect to x $\Omega \in \mathbb{R}^m \times \mathbb{R}^m \times [1, T]$,

$$\|f(x_{i+1}, t) - f(x_i, t)\| \leq k_f \|x_{i+1} - x_i\| \quad (13)$$

where k_f is the Lipschitz constant and $f(\bullet)$ is a nonlinear function.

Assumption 2. The initial value of the system satisfies the following criterion:

$$x_i(0) = x_d(0), \quad \forall i \quad (14)$$

Assumption 3. For $y_d(t)$, unique $u_d(t)$ and $x_d(t)$ exist that satisfy the following equations:

$$\begin{aligned} \dot{x}_d(t) &= f(x_d, t) + B(t)u_d(t) \\ y_d(t) &= C(t)x_d(t) \end{aligned} \quad (15)$$

where $B(\bullet)$ is a nonlinear function, $u_d(t)$ is the desired control input, and $x_d(t)$ is the desired state.

Assumption 4. The uncertainty, disturbance and their derivatives are bounded.

$$\begin{aligned} b_\nu &\triangleq \max \left\{ \sup_{t \in [0, T]} \|\nu_i(t)\|, \sup_{t \in [0, T]} \|\nu_{i+1}(t)\| \right\} \\ b_{d\nu} &\triangleq \max \left\{ \sup_{t \in [0, T]} \|\dot{\nu}_i(t)\|, \sup_{t \in [0, T]} \|\dot{\nu}_{i+1}(t)\| \right\} \\ b_\omega &\triangleq \max \left\{ \sup_{t \in [0, T]} \|\omega_i(t)\|, \sup_{t \in [0, T]} \|\omega_{i+1}(t)\| \right\} \end{aligned} \quad (16)$$

Let

$$\begin{aligned} \Delta \dot{x}_i &= \dot{x}_d - \dot{x}_i \\ &= f(x_d, t) + B(t)u_d(t) - f(x_i, t) - B(t)u_i(t) \\ &\quad - \omega_i(t) = \Delta f_i(t) + B(t)\Delta u_i(t) - \omega_i(t) \end{aligned} \quad (17)$$

where x_d is the desired state of the system and \triangleq indicates a formula that is defined as another formula.

Let us consider nonlinear multivariable systems [28, 29] with uncertainty and disturbances as follows:

$$\begin{aligned} \dot{x}_i(t) &= f(x_i, t) + B(t)u_i(t) + \omega_i(t) \\ y_i(t) &= C(t)x_i(t) + \nu_i(t) \end{aligned} \quad (18)$$

where i and t are the number of iterations and the running time, respectively. $x_i(t) \in \mathbb{R}^m$ is the state variable, $u_i(t) \in \mathbb{R}^m$ is the control variable, $y_i(t) \in \mathbb{R}^m$ is the system output, $\omega_i(t)$ and $\nu_i(t)$ are the uncertainty item and interferential term, respectively, and $w(\bullet)$ is a nonlinear function.

Theorem 5. *The sufficient convergence conditions of the ILC algorithm are*

$$\lim_{i \rightarrow \infty} \|\Delta u_i(t)\|_\lambda \leq d < 1 \quad (19)$$

Then, we choose a sufficiently large constant λ to reach the conclusion of this theorem: the error $e_i(t)$ approaches uniformly bounded with $[0, T]$ by ILC.

Proof. From (11) and (18), we obtain

$$\begin{aligned}
\Delta u_{i+1} &= u_d - u_{i+1} \\
&= u_d - au_0 - (1-a)u_i - \varphi_1 \dot{e}_i - \varphi_2 \dot{e}_{i+1} - \Psi_1 e_i \\
&\quad - \Psi_2 e_{i+1} \\
&= a\Delta u_0 + (1-a)\Delta u_i - \varphi_1 \dot{e}_i - \varphi_2 \dot{e}_{i+1} - \Psi_1 e_i \\
&\quad - \Psi_2 e_{i+1} \\
&= a\Delta u_0 + (1-a)\Delta u_i \\
&\quad - \varphi_1 [\dot{C}(t)\Delta x_i(t) + C(t)\Delta \dot{x}_i(t) - \dot{v}_i(t)] \\
&\quad - \varphi_2 [\dot{C}(t)\Delta x_{i+1}(t) + C(t)\Delta \dot{x}_{i+1}(t) - \dot{v}_{i+1}(t)] \\
&\quad - \Psi_1 [C(t)\Delta x_i(t) - v_i(t)] \\
&\quad - \Psi_2 [C(t)\Delta x_{i+1}(t) - v_{i+1}(t)] \\
\Delta \dot{x}_i &= \dot{x}_d - \dot{x}_i \\
&= f(x_d, t) + B(t)u_d(t) - f(x_i, t) - B(t)u_i(t) \\
&\quad - \omega_i(t) = \Delta f_i(t) + B(t)\Delta u_i(t) - \omega_i(t)
\end{aligned} \tag{20}$$

Inserting (21) into (20) yields

$$\begin{aligned}
\Delta u_{i+1} &= a\Delta u_0 + (1-a)\Delta u_i - \varphi_1 [\dot{C}(t)\Delta x_i(t) \\
&\quad + C(t)[\Delta f_i(t) + B(t)\Delta u_i(t) - \omega_i(t)] - \dot{v}_i(t)] \\
&\quad - \varphi_2 [\dot{C}(t)\Delta x_{i+1}(t) \\
&\quad + C(t)[\Delta f_{i+1}(t) + B(t)\Delta u_{i+1}(t) - \omega_{i+1}(t)] \\
&\quad - \dot{v}_{i+1}(t)] - \Psi_1 [C(t)\Delta x_i(t) - v_i(t)] \\
&\quad - \Psi_2 [C(t)\Delta x_{i+1}(t) - v_{i+1}(t)]
\end{aligned} \tag{22}$$

Using Assumption 1 yields the following:

$$\begin{aligned}
\Delta u_{i+1} &= a\Delta u_0 + (1-a)\Delta u_i - \varphi_1 \dot{C}(t)\Delta x_i(t) \\
&\quad + \varphi_1 C(t)[k_{f_1}\Delta x_i + B(t)\Delta u_i(t) - \omega_i(t)] \\
&\quad - \varphi_1 \dot{v}_i(t) - \varphi_2 \dot{C}(t)\Delta x_{i+1}(t) \\
&\quad - \varphi_2 C(t)[k_{f_2}\Delta x_{i+1} + B(t)\Delta u_{i+1}(t) - \omega_{i+1}(t)] \\
&\quad - \varphi_2 \dot{v}_{i+1}(t) - \Psi_1 C(t)\Delta x_i(t) + \Psi_1 v_i(t) \\
&\quad - \Psi_2 C(t)\Delta x_{i+1}(t) + \Psi_2 v_{i+1}(t)
\end{aligned} \tag{23}$$

$$\begin{aligned}
&[1 + \varphi_2 C(t)B(t)]\Delta u_{i+1} \\
&= [(1-a) - \varphi_1 C(t)B(t)]\Delta u_i \\
&\quad - [\varphi_1 \dot{C}(t) + \Psi_1 C(t) + \varphi_1 C(t)k_{f_1}]\Delta x_i(t) \\
&\quad - [\varphi_2 \dot{C}(t) + \Psi_2 C(t) + \varphi_2 C(t)k_{f_2}]\Delta x_{i+1}(t) \\
&\quad + \varphi_1 C(t)\omega_i(t) - \varphi_1 \dot{v}_i(t) + \varphi_2 C(t)\omega_{i+1}(t) \\
&\quad - \varphi_2 \dot{v}_{i+1}(t) + \Psi_1 v_i(t) + \Psi_2 v_{i+1}(t) + a\Delta u_0
\end{aligned} \tag{24}$$

Let us take the norm on both sides of (24), where λ -norm is defined to simplify the formula as follows:

$$\begin{aligned}
\gamma_1 &= \sup \| (1-a)I - \varphi_1 C(t)B(t) \|, \\
\gamma_2 &= \sup \| [I + \varphi_2 C(t)B(t)]^{-1} \| \\
b_{\varphi_1} &= \sup_{t \in [0, T]} \| \varphi_1(t) \|, \\
b_{\Psi_1} &= \sup_{t \in [0, T]} \| \Psi_1(t) \|, \\
b_C &= \sup_{t \in [0, T]} \| C(t) \|, \\
b_{dC} &= \sup_{t \in [0, T]} \| \dot{C}(t) \|, \\
b_{\varphi_2} &= \sup_{t \in [0, T]} \| \varphi_2(t) \|, \\
b_{\Psi_2} &= \sup_{t \in [0, T]} \| \Psi_2(t) \|, \\
b_B &= \sup_{t \in [0, T]} \| B(t) \|, \\
b_{k_{f_1}} &= \sup_{t \in [0, T]} \| k_{f_1} \|, \\
b_{k_{f_2}} &= \sup_{t \in [0, T]} \| k_{f_2} \|,
\end{aligned} \tag{25}$$

Then, (24) implies the following.

$$\begin{aligned}
\|\Delta u_{i+1}\| &= \gamma_1 \gamma_2 \|\Delta u_i\| + \gamma_2 [b_{\varphi_1} b_{dC} + b_{\Psi_1} b_C + b_{\varphi_1} b_{k_{f_1}}] \\
&\quad \cdot \|\Delta x_i(t)\| + \gamma_2 [b_{\varphi_2} b_{dC} + b_{\Psi_2} b_C + b_{\varphi_2} b_{k_{f_2}}] \\
&\quad \cdot \|\Delta x_{i+1}(t)\| + \gamma_2 [b_{\varphi_1} b_C b_\omega + b_{\varphi_2} b_C b_\omega + b_{\varphi_1} b_{d\nu} \\
&\quad + b_{\varphi_2} b_{d\nu} + b_{\Psi_1} b_\nu + b_{\Psi_2} b_\nu] + \gamma_2 a \|\Delta u_0\|
\end{aligned} \tag{26}$$

Multiplying both sides of (13) by $e^{-\lambda t}$ and taking the λ -norm yield

$$\begin{aligned}
\|\Delta u_{i+1}\|_\lambda &\leq \gamma_1 \gamma_2 \|\Delta u_i\|_\lambda + \gamma_2 \xi_1 \|\Delta x_i(t)\|_\lambda \\
&\quad + \gamma_2 \xi_2 \|\Delta x_{i+1}(t)\|_\lambda + \mu_0
\end{aligned} \tag{27}$$

where $\xi_1 = b_{\varphi_1} b_{dC} + b_{\Psi_1} b_C + b_{\varphi_1} b_{k_{f_1}}$, $\xi_2 = b_{\varphi_2} b_{dC} + b_{\Psi_2} b_C + b_{\varphi_2} b_{k_{f_2}}$, and $\mu_0 = \gamma_2 [b_{\varphi_1} b_C b_\omega + b_{\varphi_2} b_C b_\omega + b_{\varphi_1} b_{d\nu} + b_{\varphi_2} b_{d\nu} + b_{\Psi_1} b_\nu + b_{\Psi_2} b_\nu] + \gamma_2 a \|\Delta u_0\|_\lambda$.

TABLE 1: Variables and parameters of the two-wheeled self-balancing mobile robot.

Symbol	Parameter	Value	Unit
M	Mass of the wheel	1.12	kg
m	Mass of the self-balancing robot body	0.51	kg
r	Wheel diameter	0.042	m
l	Distance of barycenter to axle center	0.084	m
I	Rotational inertia	0.0032	kg.m ²
g	Gravitational acceleration	9.87	m/s ²
b	Friction coefficient of wheel and ground	0.1	N/m/sec

From Assumption 2, we obtain

$$\Delta x_i(t) = \int_0^t [f[x, t] \Delta x_i(\tau) - \omega_i(\tau)] d\tau \quad (28)$$

Taking the norm yields

$$\|\Delta x_i(t)\| = b_\omega T + \int_0^t [\|\Delta x_i(\tau)\| + b_B \|\Delta u_i(\tau)\|] d\tau \quad (29)$$

Using the Bellman-Gronwall Lemma condition gives

$$\|\Delta x_i(t)\| = b_\omega T e^t + \int_0^t [e^{(t-\tau)} b_B \|\Delta u_i(\tau)\|] d\tau \quad (30)$$

Multiplying both sides of (30) by $e^{-\lambda t}$ to compute the λ -norm, we have

$$\|\Delta x_i(t)\|_\lambda \leq b_\omega T + b_B \frac{1 - e^{-\lambda T}}{\lambda} \|\Delta u_i(\tau)\|_\lambda \quad (31)$$

The same lesson applies as follows:

$$\|\Delta x_{i+1}(t)\|_\lambda \leq b_\omega T + b_B \frac{1 - e^{-\lambda T}}{\lambda} \|\Delta u_{i+1}(\tau)\|_\lambda \quad (32)$$

Substituting (31) and (32) into (27) gives

$$\begin{aligned} \|\Delta u_{i+1}\|_\lambda &\leq \gamma_1 \gamma_2 \|\Delta u_i\|_\lambda \\ &+ \gamma_2 \xi_1 \left[b_\omega T + b_B \frac{1 - e^{-\lambda T}}{\lambda} \|\Delta u_i(\tau)\|_\lambda \right] \\ &+ \gamma_2 \xi_2 \left[b_\omega T + b_B \frac{1 - e^{-\lambda T}}{\lambda} \|\Delta u_{i+1}(\tau)\|_\lambda \right] \\ &+ \mu_0 \end{aligned} \quad (33)$$

We have

$$\begin{aligned} &\left[1 - \gamma_2 \xi_2 b_B \frac{1 - e^{-\lambda T}}{\lambda} \right] \|\Delta u_{i+1}\|_\lambda \\ &\leq \left(\gamma_1 \gamma_2 + \gamma_2 \xi_1 b_B \frac{1 - e^{-\lambda T}}{\lambda} \right) \|\Delta u_i\|_\lambda + \gamma_2 \xi_1 b_\omega T \\ &+ \gamma_2 \xi_2 b_\omega T + \mu_0 \end{aligned} \quad (34)$$

This equation means that

$$b \|\Delta u_{i+1}\|_\lambda \leq a \|\Delta u_i\|_\lambda + c \quad (35)$$

where $a = \gamma_1 \gamma_2 + \gamma_2 \xi_1 b_B ((1 - e^{-\lambda T})/\lambda)$, $b = 1 - \gamma_2 \xi_2 b_B ((1 - e^{-\lambda T})/\lambda)$, $c = \gamma_2 \xi_1 b_\omega T + \gamma_2 \xi_2 b_\omega T + \mu_0$.

We choose a sufficiently large constant λ to satisfy $b > 0$ and $a/b < 1$, and we can reach the conclusion $\lim_{i \rightarrow \infty} \|\Delta u_i(t)\|_\lambda \leq c/(b - a)$. Thus, it is concluded that the error $e_i(t)$ approaches uniformly bounded with $[0, T]$ by ILC. \square

5. Simulation

In the simulation, open-closed-loop PD-ILC is used for comparison to verify the effectiveness of the proposed algorithm. The physical parameters of the two-wheeled self-balancing mobile robot are displayed in Table 1. The desired trajectory is $P_d = [\cos(\pi t) \ \sin(\pi t) \ \pi t + \pi/2]^T$. The interference terms are

$\omega_i(t) = \nu_i(t) = 0.01 \text{diag}[\sin(40\pi t)]$. The data used in the simulation are chosen as $\delta = 0.85$, $\Lambda = 6$, and $u_0 = [0.1 \ 0.1]^T$.

The simulation results are shown in Figures 4–6. Substantial error reductions take place over 40 iterations. Figure 4 shows the trajectory tracking of the wheeled self-balancing mobile robot in the process of iterative operation, and this tracking can illustrate the effectiveness of ILC laws. The angle and position are converged after 30 iterations.

The maximum tracking errors of the mobile robot are presented in Figure 5. It is obvious that the errors greatly decrease with an increasing number of iterations. In addition, this figure shows that the proposed algorithm has very small and monotonically decreasing tracking errors, and the convergence rate of this algorithm is much greater than that of the traditional open-closed-loop PD-ILC controller. The dynamic performance of the system can be greatly improved.

Figure 6 shows the pitch angle of the mobile robot. These graphs show that the pitch angle decreases as the number of iterations increases. The controlled object is a two-wheeled self-balancing robot. To maintain the balance in the dynamic movement, pitch angle cannot be zero. Compared to traditional algorithms, the proposed algorithm has a smaller pitch angle. This contribution is due to the multivariable forgetting factor. Thus, the work developed and reported in this paper forms the basis of the design and development of open-closed-loop PD-ILC with variable forgetting factor algorithm schemes for output trajectory tracking and self-balancing control of robot systems and can be extended to and adopted in practical applications.

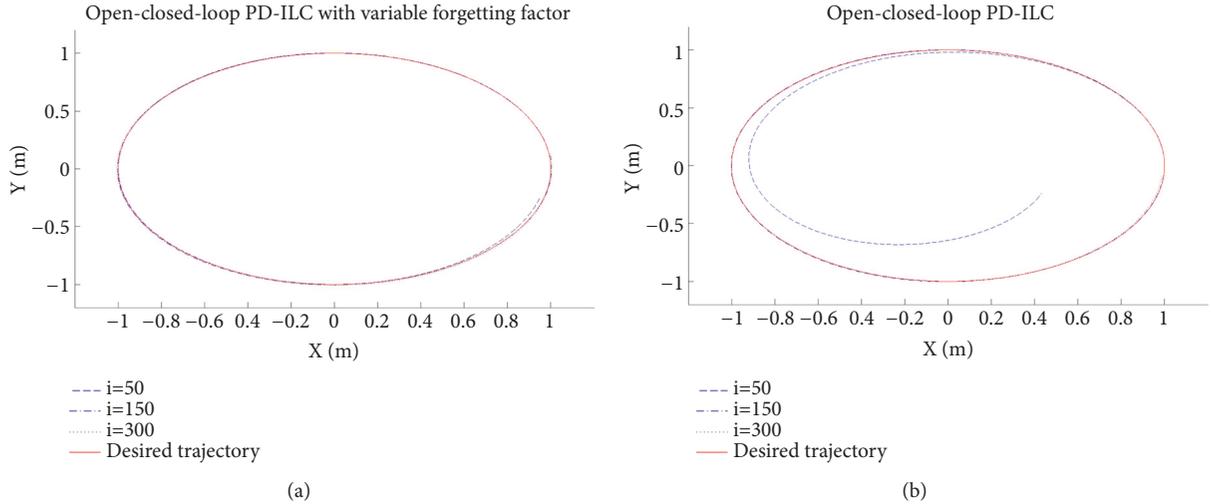


FIGURE 4: Desired and actual position trajectories.

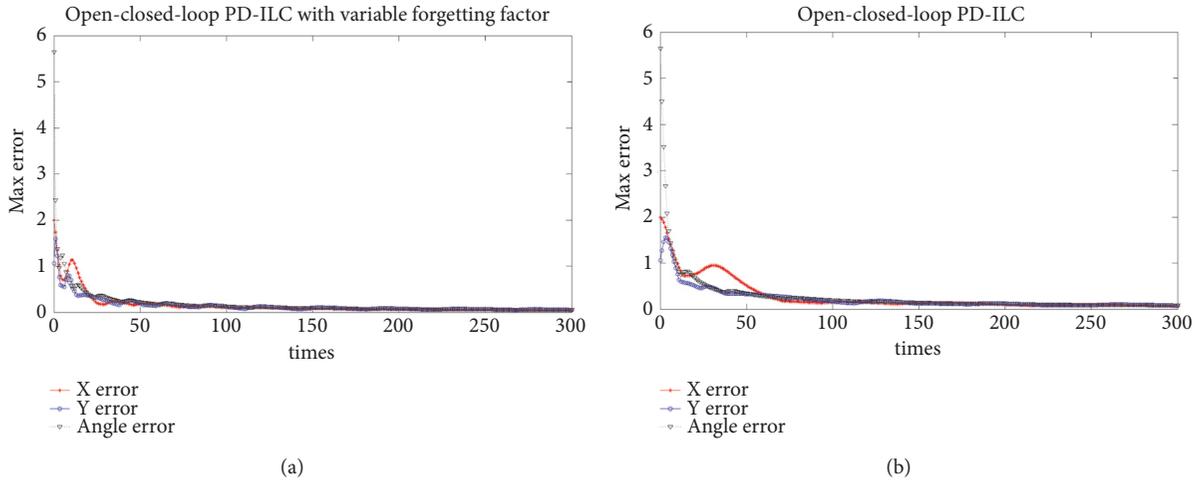


FIGURE 5: Max absolute values of the tracking error.

6. Experiment

The self-balancing robot is shown in Figure 7. We choose STM32 MCU [30–32] as the main control unit. To verify the reliability and advantages of the proposed algorithm, two groups of experiments are conducted to compare the control performance of the two algorithms with the same conditions in a real two-wheeled self-balancing mobile robot.

The motion trajectory of the robot is shown in Figures 8 and 9 indicate the desired and actual position trajectories in the last iteration and the pitch angle is shown in Figure 10. The self-balancing mobile robot travels roughly along the black line of the road, especially while turning. The results of the experiment show that the proposed algorithm can effectively track the desired trajectory and yaw angle and stabilize the pitch angle.

The comparative results indicate that, with the same parameters, the performance of open-closed-loop PD-ILC with a variable forgetting factor is better than that of traditional open-closed-loop PD-ILC in terms of robot control.

Figures 9 and 10 indicate that the designed control algorithm for two-wheeled self-balancing mobile robot systems has better trajectory tracking control ability, faster convergence speed and maintains variation in the lower pitch angle despite a variety of disturbances. The proposed algorithm has potential value for practical applications.

7. Conclusions

After adopting an ILC strategy combining open-closed-loop PD and a variable forgetting factor, a new iterative approach that can effectively solve two-wheeled self-balancing mobile robot kinematics is constructed in this paper. The self-balancing robot is a complicated dynamic system that has highly nonlinear and coupled characteristics. Acceptable trajectory tracking control and self-balancing features have been achieved with both control strategies. Simulation and experimental results validate the effectiveness and feasibility of the designed algorithm. In the future, we will apply this algorithm in other control fields.

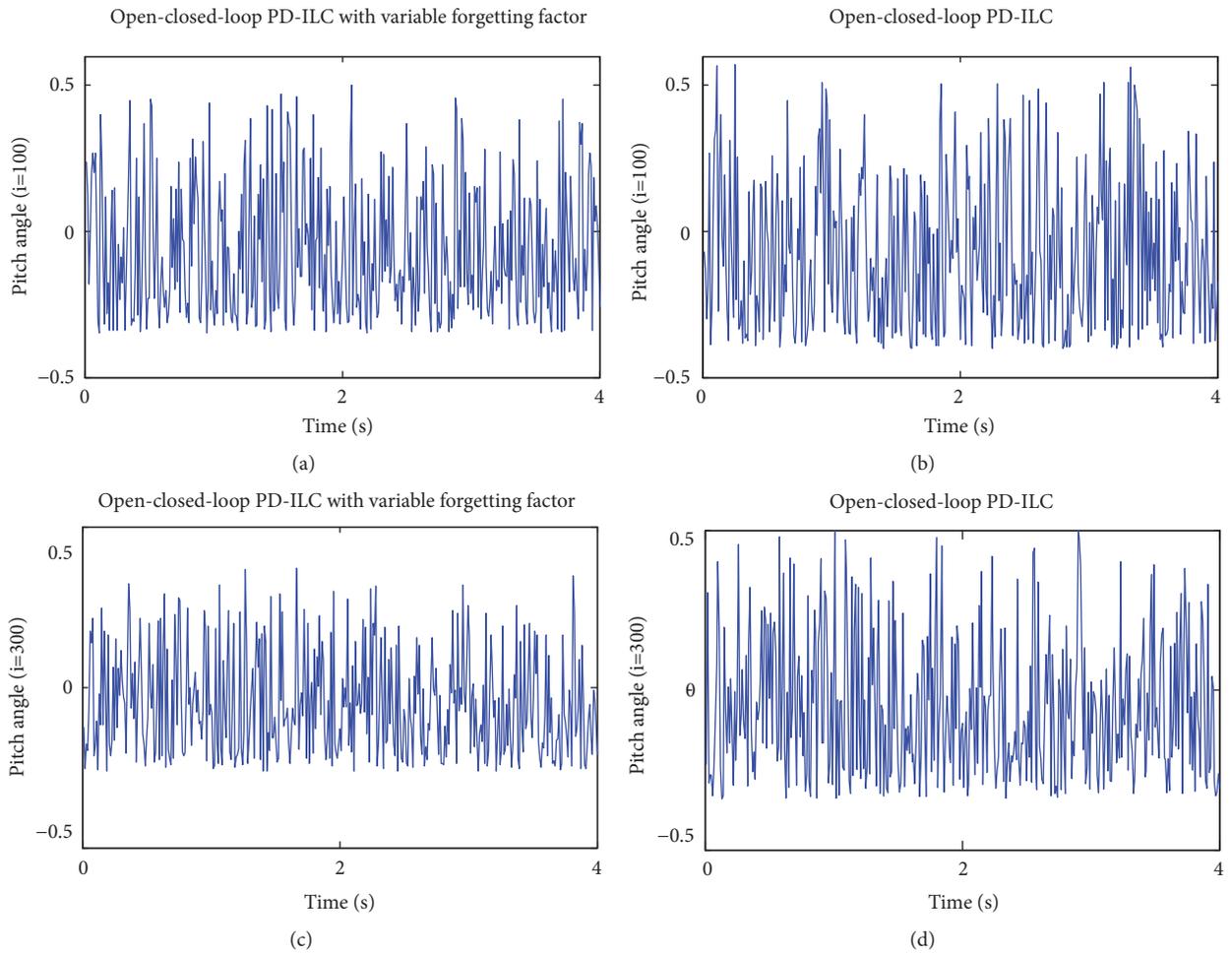


FIGURE 6: Response curves of the pitch angle.



FIGURE 7: Two-wheeled self-balancing robot prototype.

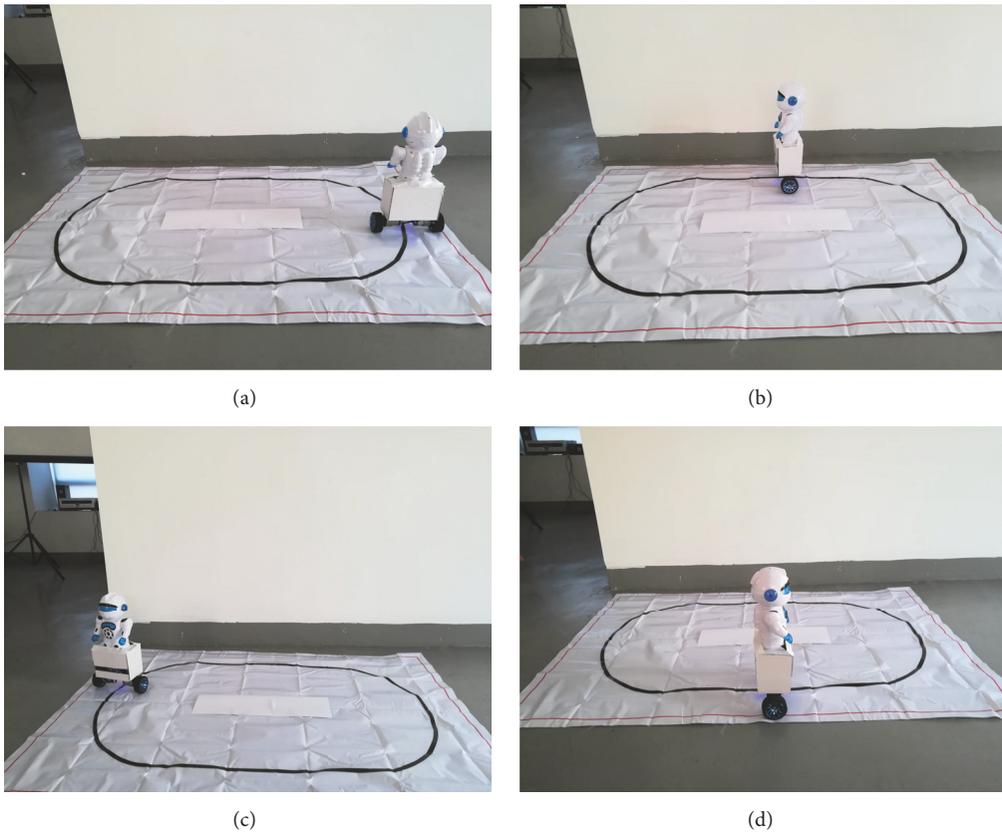


FIGURE 8: Process diagram of the two-wheeled self-balancing robot in motion.

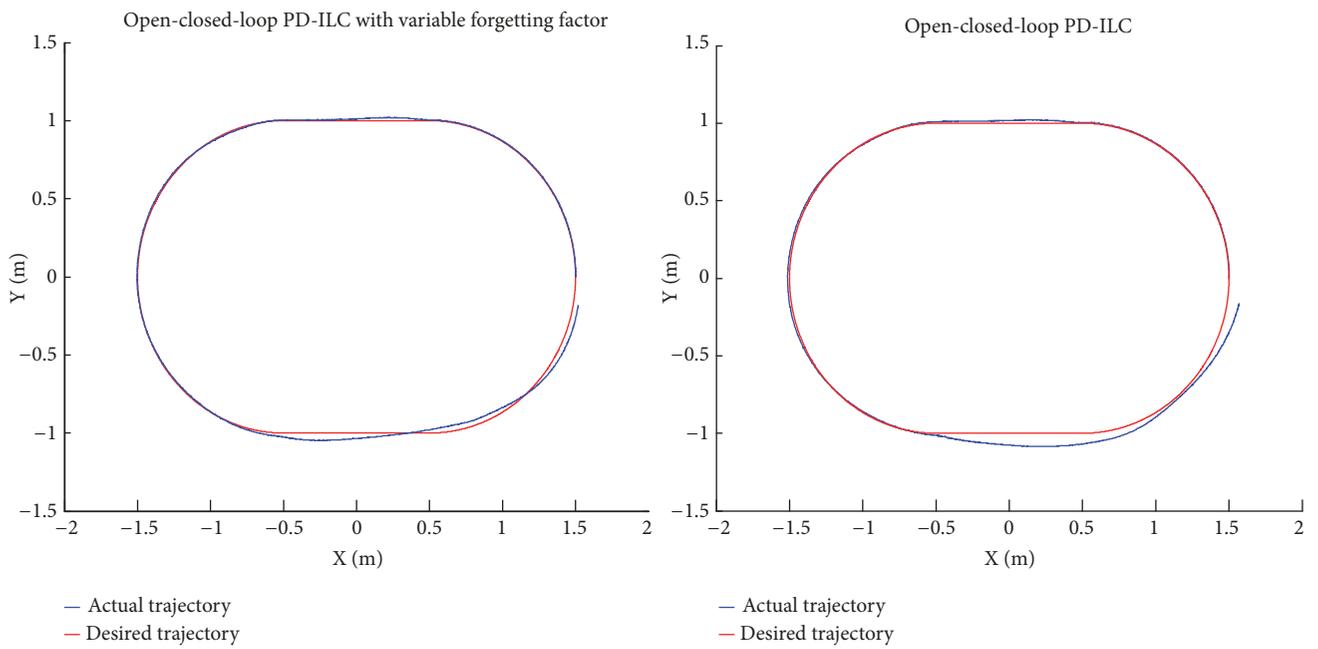


FIGURE 9: Desired and actual position trajectories in the last iteration.

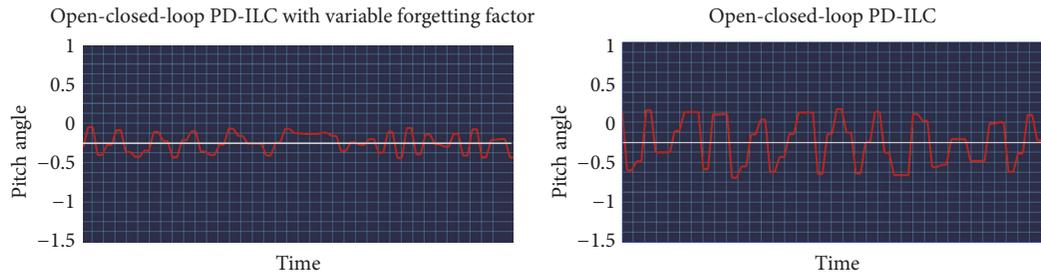


FIGURE 10: Curve of the pitch angle of the robot attitude.

Data Availability

All data and metadata used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Jian Dong designed the experiment. Jian Dong and Gang Li were involved in designing and conceiving the article. Bin He and Gang Li performed the experiment. Chenghong Zhang crunched the data. Jian Dong wrote the paper. Chenghong Zhang and Gang Li revised the paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants nos. 61801330, 61825303, 2018YFB1305300, and U1713215. It was supported by key innovation team program of innovation talents promotion plan by MOST of China (no. 2016RA4059).

References

- [1] J. Han, X. Li, and Q. Qin, "Design of Two-Wheeled Self-Balancing Robot Based on Sensor Fusion Algorithm," *International Journal of Automation Technology*, vol. 8, no. 2, pp. 216–221, 2014.
- [2] Junfeng Wu, Wanying Zhang, and Shengda Wang, "A Two-Wheeled Self-Balancing Robot with the Fuzzy PD Control Method," *Mathematical Problems in Engineering*, vol. 2012, Article ID 469491, 13 pages, 2012.
- [3] H. Yau and Q. Liang, "Sliding Mode Controller Design of a Self-Balancing Two-Wheeled Robot," *Advanced Science Letters*, vol. 9, no. 1, pp. 609–613, 2012.
- [4] G. Felix, A. Aldo, C. Silvio et al., "JOE:A mobile, inverted pendulum," *IEEE Trans on Industrial Electronic*, vol. 49, no. 1, pp. 107–114, 2002.
- [5] M. Samer, M. Al-Mjali, I. Anas et al., "Fuzzy control of a two-wheel balancing robot using DSPIC," in *Proceedings of the 2010 7th International Multi-Conference on Systems Signals and Devices (SSD)*, pp. 1–6, Jordan, 2010.
- [6] T. Nguyen, N. Duong, and P. Nguyen, "A PID backstepping controller for two-wheeled self-balancing robot," in *Proceedings of the 2010 International Forum on IEEE Strategic Technology (IFOST)*, pp. 76–81, 2010.
- [7] J. Wu and W. Zhang, "Design of fuzzy logic controller for two-wheeled self-balancing robot," in *Proceedings of the 6th International Forum on Strategic Technology, IFOST 2011*, pp. 1266–1270, China, August 2011.
- [8] X. Ruan, J. Chen, and L. Dai, "Motor learning based on the cooperation of cerebellum and basal ganglia for a self-balancing two-wheeled robot," *Intelligent Control and Automation*, vol. 2, no. 3, pp. 214–225, 2011.
- [9] J. W. Zhao and X. G. Ruan, "Modelling and control of a flexible two-wheeled self-balancing mobile robot," *International Journal of Systems, Control and Communications*, vol. 3, no. 3, p. 330, 2011.
- [10] H. Saryanto and A. Wardoyo, "An investigation on the application of fuzzy and PID algorithm in the two wheeled robot with self-balancing system using microcontroller," in *Proceedings of the International Conference on Control, Automation and Robotics: International Conference on Control, Automation and Robotics (ICCAR)*, pp. 64–68, Singapore, 2015.
- [11] F. Bouakrif, "Iterative learning control for MIMO nonlinear systems with arbitrary relative degree and no states measurement," *Complexity*, vol. 19, no. 1, pp. 37–45, 2013.
- [12] J. de Best, L. Liu, R. van de Molengraft, and M. Steinbuch, "Second-Order Iterative Learning Control for Scaled Setpoints," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 805–812, 2015.
- [13] B. Cichy, L. Hladowski, K. Galkowski, A. Rauh, and H. Aschemann, "Iterative Learning Control of an Electrostatic Microbridge Actuator With Polytopic Uncertainty Models," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 5, pp. 2035–2043, 2015.
- [14] K.-L. Han and J. S. Lee, "Iterative path tracking of an omnidirectional mobile robot," *Advanced Robotics*, vol. 25, no. 13-14, pp. 1817–1838, 2011.
- [15] K. L. Moore, M. Ghosh, and Y. Q. Chen, "Spatial-based iterative learning control for motion control applications," *Meccanica*, vol. 42, no. 2, pp. 167–175, 2007.
- [16] W. Wang, J. Chen, and L. Mao, "Two-Wheeled Mobile Robot Tracking Based on Iterative Learning Control," *Advanced Materials Research*, vol. 433-440, pp. 5866–5870, 2012.
- [17] C. Yu and X. Chen, "Trajectory tracking of wheeled mobile robot by adopting iterative learning control with predictive, current, and past learning items," *Robotica*, vol. 33, no. 7, pp. 1393–1414, 2015.
- [18] L. Zhou, J. She, and S. Zhou, "Robust H_{∞} control of an observer-based repetitive-control system," *Journal of The Franklin Institute*, vol. 355, no. 12, pp. 4952–4969, 2018.

- [19] L. Zhou, J. She, S. Zhou, and C. Li, "Compensation for state-dependent nonlinearity in a modified repetitive control system," *International Journal of Robust and Nonlinear Control*, vol. 28, no. 1, pp. 213–226, 2018.
- [20] Z. Feng, Z. Zhang, D. Pi et al., "Open-closed-loop PD-type iterative learning controller for nonlinear systems and its convergence," in *Proceedings of the WCICA 2004 - Fifth World Congress on Intelligent Control and Automation, Conference Proceedings*, pp. 1241–1245, China, June 2004.
- [21] S. Yu, J. Wu, X. Yan et al., "A PD-type open-closed-loop iterative learning control and its convergence for discrete systems," in *International Conference on Machine Learning and Cybernetics*, vol. 2, pp. 5659–5662, 2002.
- [22] Y. Hu, B. Zuo, J. Li et al., "Fast algorithm of open-closed loop pd-type iterative learning control for dc motor , intelligent control and automation," in *The Sixth World Congress on 2006 Intelligent Control and Automation*, pp. 3834–3838, WCICA, 2006.
- [23] J. Hu, J. Wang, G. Sun et al., "Self-balancing control and manipulation of a glove puppet robot on a two-wheel mobile platform," in *Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robotics and Systems*, pp. 424–425, 2009.
- [24] I. Nail, M. Omer, S. Uluc et al., "Path following with an underactuated self-balancing spherical-wheel mobile robot," in *Proceedings of the 17th International Conference on Advanced Robotics*, pp. 194–199, Istanbul, Turkey, July 2015.
- [25] D. Choi and J. Oh, "Four and two wheel transformable dynamic mobile platform," in *Proceedings of the 2011 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6404–6407, Shanghai, China, May 2011.
- [26] H. Gzyl, A. Tagliani, and M. Milev, "Laplace transform inversion on the real line is truly ill-conditioned," *Applied Mathematics and Computation*, vol. 219, no. 18, pp. 9805–9809, 2013.
- [27] F. Yin, J. Song, and F. Lu, "A coupled method of Laplace transform and Legendre wavelets for nonlinear Klein-Gordon equations," *Mathematical Methods in the Applied Sciences*, vol. 37, no. 6, pp. 781–792, 2014.
- [28] Y. Tan, H. Dai, D. Huang, and J. Xu, "Unified iterative learning control schemes for nonlinear dynamic systems with nonlinear input uncertainties," *Automatica*, vol. 48, no. 12, pp. 3173–3182, 2012.
- [29] C. Yin, J.-X. Xu, and Z. Hou, "A high-order internal model based iterative learning control scheme for nonlinear systems with time-iteration-varying parameters," *IEEE Transactions on Automatic Control*, vol. 55, no. 11, pp. 2665–2670, 2010.
- [30] S. Sun, J. Ni, Z. Chen et al., "The design of a DeviceNet - SPI converter module based on the STM32 MCU," in *Proceedings of the 2015 IEEE International Conference on Mechatronics and Automation*, pp. 843–847, China, August 2015.
- [31] T. Zhang and Y. Li, "Data acquisition design based on the STM32," in *Proceedings of the 2012 international conference on manufacturing engineering and automation*, pp. 1527–1530, Guangzhou, China, 2012.
- [32] W. Huajie and S. Juan, "The design and research of infrared sensor monitoring system based on STM32," in *Proceedings of the 2012 Fourth international conference on computational and information sciences*, pp. 876–879, 2012.

