IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 25 March 2023, accepted 14 April 2023, date of publication 19 April 2023, date of current version 7 July 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3268524

RESEARCH ARTICLE

Online Series-Parallel Reinforcement-Learning-Based Balancing Control for Reaction Wheel Bicycle Robots on a Curved Pavement

XIANJIN ZHU[1], YANG DENG[2], XUDONG ZHENG[3], QINGYUAN ZHENG[2], ZHANG CHEN[2], BIN LIANG[2], (Senior Member, IEEE), AND YU LIU[1]

[1]School of Mechatronics Engineering, Harbin Institute of Technology, Harbin 150006, China
[2]Department of Automation, Tsinghua University, Beijing 100084, China
[3]School of Automation, Beijing University of Posts and Telecommunications, Beijing 100876, China

Corresponding author: Yu Liu (lyu11@hit.edu.cn)

This work was supported by the National Natural Science Foundation of China under Grant 62203252 and Grant 52205008.

ABSTRACT The reaction wheel bicycle robot is a kind of unmanned mobile robot with great potential. However, the control of such bicycle robots on a curved pavement under inaccurate model parameters, model uncertainties and disturbances is challenging due to the lateral instability and underactuated characteristic. Applying conventional control methods to this problem often results in brittle and inaccurate controllers. In this paper, an online serial-parallel combination reinforcement learning with conventional control methods is designed to achieve the path tracking and banlancing control for a reaction wheel bicycle robot on curved pavements. The parallel part of the controller refers to compensating the equilibrium point and the serial part of the controller refers to adjusting the parameters of a sliding mode controller that tracks the target roll equilibrium point. The comparison between the proposed controller and several existing controllers in experimental test built in Matlab Simscape illustrates stronger robustness and better control performances.

INDEX TERMS Reaction wheel bicycle robot, reinforcement learning, robustness, sliding model control.

I. INTRODUCTION

The bicycle robots (BRs) with only two ground contact points can provide flexible maneuverability and convenient deployment over rough terrain without relying on the complex active suspension control [1] or multi-wheel cooperative control [2] of four-wheel vehicles. In addition, the light weight of the BR also provides higher energy efficiency and faster acceleration [3], [4]. Research interests have been aroused on the balancing control of BR on a flat terrain [5], [6], [7], [8]. However, in order to realize the advantages of BR on rough terrain, it is necessary to investigate the balancing control of BR on a curved pavement due to the lateral instability and underactuated characteristic of BR.

The balancing control of BR can be divided into two categories. The first category only employs the steering and velocity as inputs [9], [10], [11], [12]. For such methods, it is challenging to keep balance and track a given path over rough terrain. As for the second category, it stays balanced through controlling the auxiliary balancing mechanisms, such as control moment gyroscopes [13], [14], mass balancers [15], [16], reaction wheel [17], [18], [19], etc. The reaction wheel bicycle robot (RWBR) has been studied extensively by virtue of its simple construction and the lightweight configuration of the system. In addition, to the best knowledge of the authors, few studies have concerned the balancing control and path tracking of the RWBR on a curved pavement. Therefore, RWBR is selected as the research object in this paper.

Previous studies have applied different methods for balancing control of RWBR, including linear control, nonlinear control and intelligent control. The linear control, such as proportional-integral-differential (PID) [20] and linear quadratic regulator (LQR) [17], [21], can achieve balancing control by using the local linearization around the equilibrium point. However, external disturbances and unmodeled characteristics might lead to degradation of the control

The associate editor coordinating the review of this manuscript and approving it for publication was Engang Tian.

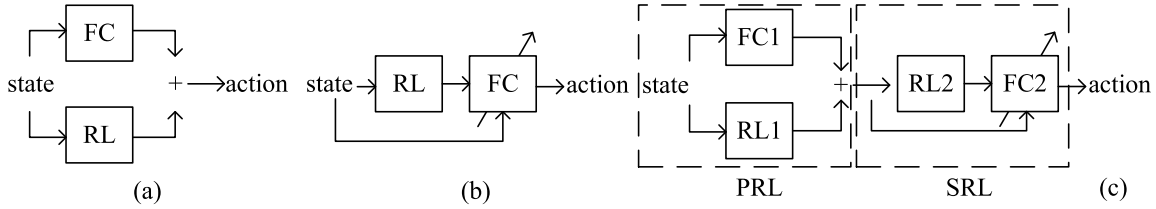

FIGURE 1. Combination of FC and RL: (a) The parallel model (b) The serial model (c) The serial-parallel model.

performance of linear controllers and even the instability of the system when the balancing control of RWBR is on a curved pavement. To improve the control performance, some studies have investigated the nonlinear control of BR. A sliding mode controller combined with low-pass filtering algorithm was proposed in [22], which is shown to be more robust than linear controllers. In another study, to deal with impulse disturbances and system uncertainties, a fuzzy sliding mode controller was designed, but the determination of fuzzy rules was rather complicated. An adaptive sliding mode controller was proposed in [23] for RWBR on flat terrain, in which a monotonic adaptation law was designed to tune the gains of the reaching control part. As to intelligent control, the neural network is applied to fit the system model or control strategy. The control strategy is tuned via continuous interactions between the system and the environment to maximize the expected return. As reinforcement learning (RL) achieved significant results in various tasks [24], [25], [26], some scholars have concerned the application of RL in BR tasks [27], [28], [29], [30].

In order to deal with the disadvantages in sampling efficiency of RL, the control framework that combines the stability guarantee of conventional feedback controls (FC) with the optimization ability of RL has been studied in recent years. In [31], an adaptive RL-based PID was tested in several simulated environments and in a real time robotic platform showing that RL was used to compensate or even adapt to changes in the uncertain environments. In [32], an improved backstepping control method was designed based on RL and fuzzy state observer for strict-feedback systems with unmeasurable states and proved to be bounded by the Lyapunov method. In [33], a performance-constrained fault-tolerant dynamic surface control (DSC) algorithm based on RL was proposed for nonlinear systems with unknown parameters and actuator failures. Although the combination between RL and FC can attenuate the problem of sampling efficiency, the application of RL in underactuated and unstable systems such as RWBR is limited.

In general, this combination between FC and RL can be divided into parallel (Fig. 1(a)) [34] and serial modes (Fig. 1(b)) [35], [36] for disturbed bicycle robots on a curved pavement strategies, the serial RL (SRL) uses the optimization capabilities of RL to tune the parameters of FC. In our previous work [36], the serial mode of RL and FC was used for the balancing control of RWBR driving straightly on a curved pavement. By comparing the performance of the three controllers and SRL, it was shown that SRL can effectively reduce the influence of matched and mismatched disturbances on the controller by adaptively adjusting the sliding mode surface parameters and reaching law parameters of the controller online. The current study tends to consider the balance of the RWBR under path tracking on a curved pavement with inaccurate model parameters, simplified dynamic model and disturbances. The exact equilibrium of the roll angle is very useful for improving the robustness and reducing the output torque of the controller, but it is difficult to be analytically computed on curved pavements under uncertainties and disturbances. Therefore, the controller proposed in this paper is shown in Fig. 1(c). PRL is used to estimate the target equilibrium point and SRL is used to realize the tracking control of the equilibrium point.

The main contributions of this paper are summarized as follows:

1) In order to improve the robustness of the balancing control, a serial-parallel reinforcement learning (SPRL) controller is proposed. The controller is consisted of a PRL and a SRL, the former is the compensation of equilibrium point based on RL and the latter is the online gain adaptation (including the parameters of the sliding surface) of sliding mode controller (SMC) based on RL. And the effectiveness of SPRL is demonstrated in Matlab Simscape by comparing with several existing control methods.

2) A simplified dynamic model of RWBR on a curved pavement is derived, and an online approximation of the equilibrium point is put forward based on this model. The error dynamics of the roll angle is obtained by an equivalent inertia wheel pendulum.

3) In this paper, an online serial-parallel combination reinforcement learning with conventional control methods is designed to achieve the path tracking and balancing control for a reaction wheel bicycle robot on curved pavements. This is different from existing related studies about the RWBR from the perspective of the task. First of all, previous studies have not considered the influence of an unstructured curved pavement on the balancing control of RWBR. Secondly, in our previous work [36], the balancing control of RWBR driving straightly on a curved pavement was considered while not considering path tracking on a curved pavement.

This paper is organized as follows: The RWBR is described in Section II, and the dynamics of the bicycle and the

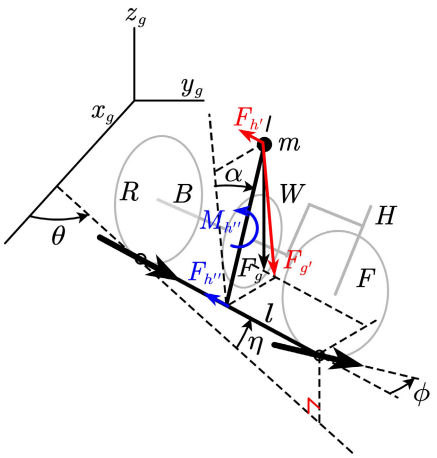

FIGURE 2. RWBR on a curved pavement.

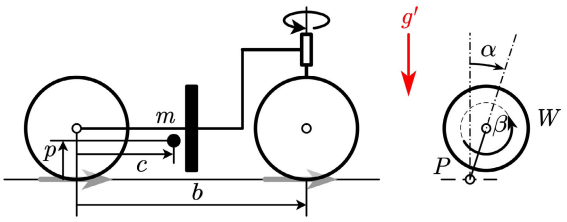

FIGURE 3. (a) Equivalent gravity field with $\alpha = 0$ and (b) Equivalent inertia wheel pendulum.

equivalent inertia wheel pendulum are established using some assumptions and simplifications. In Section III, the SPRL controller is designed. The Actor Net. and Critic Net. of PRL, the SMC and the SRL are described in detail in this part. In Section IV, the simulation environment in Matlab Simscape is built, and three test cases are designed; and then, the performance of four different controllers are compared. Finally, in Section V, a conclusion is addressed.

The video of the experiment is available at the following website: https://github.com/ZhuXianjinGitHub/SPRL. (accessed on 10th February 2023).

II. DYNAMICS

The RWBR on a curved pavement is illustrated in Fig. 2. The RWBR is consisted of five parts: rear wheel, body frame, reaction wheel, steering handlebar and front wheel (noted as R, B, W, H and F, respectively). The fork angle is zero. The inertia frame is defined as $o_g x_g y_g z_g$. The θ is the yaw angle, the η is the pitch angle, the α is the roll angle and the ϕ is the steering angle. In addition, p is the vertical distance between the center of gravity and the ground point of the rear wheel, c is the horizontal distance from the rear wheel ground point to the center of mass, and b is the distance between the front and rear wheel ground points in Fig. 3. The following assumptions are made: (1) The thickness of the rear and front wheels is negligible, and the contacts between wheels and the ground are regarded as point contacts. (2) These five parts are symmetrical with respect to the plane of the rear and front wheels, so the center of mass of these bodies are in the same plane. (3) The front and rear wheels are always in contact with the ground.

Remark 1: Assumption (1) is often considered in balancing control of RWBR. Assumption (2) is a favorable and easy assumption to be implemented in real RWBR. Assumption (1) and Assumption (2) are used to simplify the derivation of the nominal model in this paper. As for Assumption (3), it is mainly for the impact of the lifting of wheel-ground constraint on the dynamic model. The constraints in RWBR are the same as in [11].

The motion state of the inertia wheel will not affect the movement of the RWBR. Therefore, the dynamic modeling of RWBR can be divided into two parts for the design of the balancing controller, which are the dynamics of the bicycle and the dynamics of the equivalent inertia wheel pendulum. In the modeling of the bicycle dynamics, the bicycle is considered as a point of mass with two contacts with the ground. The gravitational force $F_g = mg$ of the bicycle robot on the slope can be converted to $F_{g'} = mg\cos(\eta)$, $F_{h''} = mg\sin(\eta)$ and $M_{h''} = pmg\sin(\eta)$, where $F_{g'}$, $F_{h''}$, and $M_{h''}$ are all in the plane of the body frame. Since the rear wheels in this paper are controlled by speed servo, we could ignore the effect of $F_{h''}$. Under assumption (3), $M_{h''}$ is also ignored. Then $g' = \cos(\eta)\, g$ is defined according to the equivalent gravity [37], [38]. As derived in [11], [12], and [39], $\sigma \triangleq \frac{c}{b}\tan(\phi)$ and the constrained Lagrangian of the bicycle is calculated as follows:

$$\begin{aligned} & mp^2\ddot{\alpha} - mcp\sigma\dot{v}_r cos(\alpha) \\ & \quad = mg' psin(\alpha) \\ & \qquad - mcp\sigma v_r sin(\alpha) + mpcos(\alpha)\sigma\upsilon_r^2\,(1 + psin(\alpha)\sigma) \\ & \qquad + \frac{1}{2}\frac{\partial}{\partial\alpha}J(\alpha,\sigma)\,\dot{\sigma}^2 + mcpcos(\alpha)v_r\dot{\sigma} + \tau_{rw}. \end{aligned} \tag{1}$$

We assume that the true value of the target roll angle α_{target} is the equilibrium point under zero dynamics, and the estimate value $\hat{\alpha} = \Psi(\phi, \upsilon_r, \eta)$ can be obtained by the formula (2) (from [11]).

$$0 = \left(g' - c\sigma v_r\right)\tan\left(\hat{\alpha}\right) + c\sigma\upsilon_r^2 + p(c\sigma)^2\upsilon_r^2\sin\left(\hat{\alpha}\right) \tag{2}$$

where υ_r is the component of the velocity of the rear-wheel contact along the contact-line as measured from the virtual inertia frame generated by the equivalent gravity. In general, $\hat{\alpha}$ does not equal to α_{target}.

As for the dynamics of the equivalent inertia wheel pendulum derived in [36] and [40], considering the body frame and rear and front wheels as a unit P, and the reaction wheel as another unit W. The mass and inertia matrix of the two parts with respect to the body-fixed reference frame are m_1, I_1, m_2 and I_2. L_1 and L_2 represent the distance between the centroids of P and W and the connection between front- and rear-wheel ground points on a flat road. The following

simplified model of equivalent inertia wheel pendulum can be derived:

$$\begin{aligned} d_{11}\ddot{\alpha} + d_{12}\ddot{\beta} - \bar{m}g' \sin(\alpha) &= d_1 \\ d_{21}\ddot{\alpha} + d_{22}\ddot{\beta} &= \tau_{rw} + d_2 \end{aligned} \tag{3}$$

where $d_{11} = m_1 l_1^2 + m_2 l_2^2 + I_1 + I_2$, $d_{12} = d_{21} = d_{22} = I_2$, $\bar{m} = m_1 l_1 + m_2 l_2$, β is the reaction wheel angle, τ_{rw} is the input torque of the reaction wheel's motor, and d_1 and d_2 are mismatched/matched disturbances. Define $e = \hat{\alpha} - \alpha$ as the tracking error of the roll angle, then from [41] it is possible to derive the following error dynamics:

$$\ddot{e} = \frac{d_{22}}{\det D}\bar{m}g' \sin(e) - \ddot{\hat{\alpha}} - \frac{d_{12}}{\det D}\tau_{rw} \tag{4}$$

where $\det D = d_{11}d_{22} - d_{12}d_{21} > 0$, and $\ddot{\hat{\alpha}}$ is estimated by a second-order low pass filter from $\hat{\alpha}$.

III. CONTROLLER DESIGN

In traditional RL, the standard Markov decision process framework is often considered for picking optimal actions to maximize the rewards over discrete timesteps. The state at time t only depends on the state at the time $t-1$ and the corresponding action, which is independent of other historical states and inputs. The agent learns a policy $u_t = \pi(s_t)$ by maximizing the return $R_t = \sum_{i=t}^{T} \gamma^{(i-t)} r_i$, where T is the horizon that the agent optimizes over, γ is the discount factor and r is the reward. RL often does not attempt to model or identify the dynamics. Instead, it finds actions that maximize rewards as the estimation of the state-action value.

As for FC with RL discussed in [32] and [42], using RL to obtain the approximate optimal solution of Hamilton-Jacobi-Bellman (HJB) equation can avoid to solve the analytical solution of HJB. And it is often difficult to obtained. In addition, RL is a data-driven approach that effectively deal with unknown dynamics and disturbances through gradient updating of neural networks in interactions with environment online [24], [43].

In this paper, two different kinds of RL and conventional feedback control are adopted in this paper for the balancing control of the RWBR as shown in Fig. 1c. The first one is the compensation of the $\hat{\alpha}$ by PRL. The second one is the adjustment of the parameters of the SMC by SRL. In this section, the PRL, the SRL and the SPRL will be described. For the choice of the RL algorithm, proximal policy optimization (PPO) [44], which is a model-free, online, on-policy, actor-critic framed [45], policy gradient RL method, is selected. PPO uses conservative policy iterations based on an estimator of the advantage function to guarantee the monotonic improvement for general stochastic policies. The monotonic improvement guarantee for general stochastic policies can be found in [47]. The update of the parameters of the Actor Net. and Critic Net. in the PPO can be found in [36] and [44], which will not be discussed in this paper.

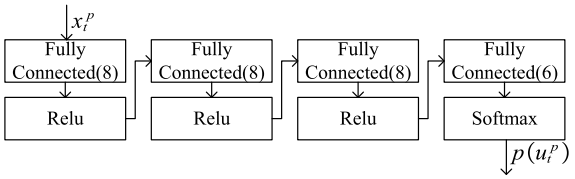

FIGURE 4. Actor Net. in Parallel RL.

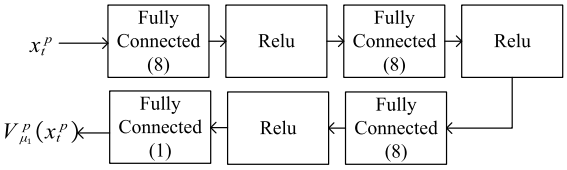

FIGURE 5. Critic Net. in Parallel RL.

A. PRL

The exact value of α_{target} is difficult to be obtained analytically. As proposed in [19] and [34], RL methods allow agents to learn $\Delta\alpha = \alpha_{target} - \hat{\alpha}$ through interaction with environments. In order to reduce the need for interactive samples, PPO with discrete output is adopted in the PRL and the output at time t is:

$$\Delta\alpha_t = K_1 u_t^p \tag{5}$$

where K_1 is the output coefficient of the PRL and $u_t^p \in \{-1, -0.75, -0.5, -0.25, 0.25, 0.5, 0.75, 1\}$.

The Actor Net. (Fig. 4) of PRL is used to map the states $x_t^p = \left(\alpha, \phi, \eta, v_r, \tau_{rw}, \hat{\alpha}\right)_t$ to the actions $p(u_t^p)$, which presents the probability of each discrete action u_t^p, $p(u_t^p) \sim \pi_{\theta_1}^p\left(x_t^p\right)$ in which θ_1 represents the parameters of the policy π^p.

The Critic Net. implements a value function approximator that is used to map the states x_t^p to a scalar value $V_{\mu_1}^p\left(x_t^p\right)$, in which μ_1 means the parameters in the Critic Net. The scalar value represents the predicted discounted cumulative long-term reward when the agent starts from the given state and takes the best possible action. The Critic Net. of the RPL is shown in Fig. 5, which is composed of a deep neural network with ReLU nonlinear activation function. The gradient descent calculation of the Critic Net. is to update μ_1 in the policy V^p.

Reward r_{t+1} can be calculated as:

$$r_t = \begin{cases} 0 & r_t < 0, \\ 1 - \kappa^p e^{|\tau_{rw}|} & 0 \le r_t < 1 \\ 1 & r_t \ge 1 \end{cases} \tag{6}$$

where $\kappa^p > 0$ is a constant.

Remark 2: PRL is introduced to compensate the equilibrium point as for the nominal analytical model with uncertainties. The only difference between PRL and [34] is that the latter is directly used to compensate the output of the controller. The motivation using PRL to compensate the equilibrium point other than the output of the controller is to take

full advantage of RWBR's prior knowledge of dynamic models. This is obviously conducive to improving the efficiency of RL.

B. SRL

Firstly, a terminal sliding mode controller is constructed. Then, the actor-critic framework, SMC Net., and the key components of the SRL are presented. The process and convergence of the SRL are described and the analysis of the PPO optimization process for SMC is performed in our previous work [36]. By defining $x_1 = e$ and $x_2 = \dot{e}$, the tracking model can be expressed as follows:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{d_{22}}{\det D}\bar{m}g' \sin(e) - \ddot{\alpha}_{target} - \frac{d_{12}}{\det D}\tau_{rw} \end{aligned} \tag{7}$$

The SMC is designed according to the method in [37]. The recursive sliding surface is defined as follows:

$$\begin{aligned} s_0 &= x_1 \\ s_1 &= \dot{s}_0 + \alpha_0 s_0 + \beta_0 |s_0|^{p0} sign(s_0) \end{aligned} \tag{8}$$

where $\alpha_0, \beta_0, 0 < p_0 < 1$ and the sign function is defined as follows:

$$sign(s) = \begin{cases} 1, & s > 0 \\ -1, & s < 0 \end{cases} \tag{9}$$

and sign(0) $\in [-1, 1]$. Next, based on the sliding surfaces (8), the following output of the mean part of the SMC Net. is defined as follows:

$$\begin{aligned} u = &\frac{1}{c_2}(c_1 g' \sin(x_1) - \ddot{\hat{\alpha}} + \alpha_0 \dot{s}_0 + p_0 \beta_0 |s_0|^{(p_0-1)} \dot{s}_0 \\ &+ \varphi s_1 + \vartheta |s_1|^{p_1} sign(s_1)) \end{aligned} \tag{10}$$

where $\varphi, \vartheta > 0$ and $\theta_2 = [\alpha_0, \beta_0, \varphi, \vartheta]^T$. The reward function in the SRL is of the same form as for it in [36].

$$r_t = \begin{cases} 0 & r_t < 0, \\ 1 - \kappa^s \bar{X}(t)^2 & 0 \le r_t < 1 \\ 1 & r_t \ge 1 \end{cases} \tag{11}$$

where $\kappa^s = [a_1, a_2, a_3]$, $a_i > 0$ and $\bar{X} = [x_1, x_2, u]^T$.

In order to adaptively adjust the coefficients θ_2 of SMC using automatic differential software, the SMC controller is represented as a neural network. The weights of the neural networks represent the coefficients of the SMC. Then, the SMC controller represented by a neural network is used to replace the mean part of the PPO's actor network. Finally, the optimization framework based on PPO with an actor-critic is explored to adaptively adjust the coefficients of SMC. These steps are described in [36].

Remark 3: In [46], the adaptive terminal sliding mode was designed for RWBR by tuning parameters φ, ϑ online. The online adaptation of φ, θ can attenuate the matched disturbance, and well-adjusted parameters such as $\alpha_0, \beta_0, \varphi, \vartheta$ can lead to a more robust sliding surfaces against mismatched disturbance. The main motivation of SRL is to improve the robustness with respect to matched and mismatched disturbances by tuning parameters $\alpha_0, \beta_0, \varphi, \vartheta$ online. Moreover, the adaptive gains of [46] are monotonic, which may cause more serious chattering in practice. The comparison between SRL and adaptive terminal sliding mode is shown in our previous work [36].

TABLE 1. The parameters of RWBR.

	True	Seed 1	Seed 2	Seed 3
d_{11}	0.033	0.0358	0.0368	0.0391
d_{22}	0.004	0.0041	0.0035	0.0033
$\bar{m}$	0.2742	0.2609	0.2544	0.3267

C. SPRL

Finally, the SPRL in this paper is similar to the hierarchical RL for that the output of PRL is a dimension of the input of SRL in Fig. 6. The PRL is used to compute an approximated equilibrium of target roll angle and attenuation of the inaccurate parameters and the model uncertainties. The SRL is used to compensate matched disturbance and attenuate mismatched disturbances as proposed in [36]. Learning two levels of policies simultaneously is problematic due to non-stationary transition and reward functions that naturally emerge. To make it easier for the PRL to learn, PRL needs to act at longer time scales than the SRL [48], [49].

IV. SIMULATION EXPERIMENT

In order to demonstrate the effectiveness of the SPRL controllers proposed in this paper, three test cases built in Matlab Simscape [50] are formed by the combination of different rear-wheel angle velocities and height of the terrain. And several comparative tests are designed. First, performances of SMC, Model Predictive Control (MPC) [51] and SRL are compared in the roll angle tracking control task. Next, the performances of SRL and SPRL compensation are compared. In addition, all the comparison trials are conducted under three different random seeds.

A. SIMULATION PLATFORM

To verify the performance of the controller proposed in this paper, the simulation environment is built in Matlab Simscape, as shown in Fig. 7.

In the simulation environment, the RWBR is placed on a curved pavement. According to the physical parameters of the RWBR, the parameters in Formula (1) are calculated as shown in Table 1. The first column in the table represents the true values of BR. The other columns represent estimates of BR under different random seeds. The other parameters of the BR are $p = 0.1\ m$, $c = 0.15m$ and $b = 0.3m$. The function relationship between the height of the curved pavement and the direction of x of the inertia frame is $y = K_2 \sin(K_3 x) + K_4$, where K_2 is a given constant, K_3 is generated by $K_3 = 0.8 + 0.4\gamma_1$ and $K_4 = 0.1\gamma_2 K_2$, both γ_1 and γ_2 are random numbers betweent 0 and 1. The surface plot of the terrain for case 3 is shown in Fig. 8.

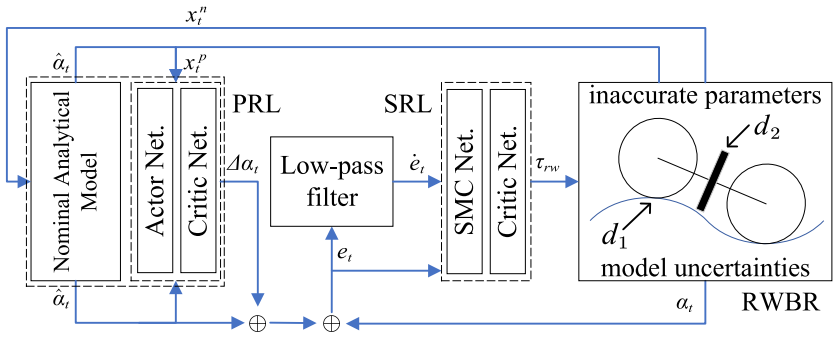

FIGURE 6. The SPRL Controller.

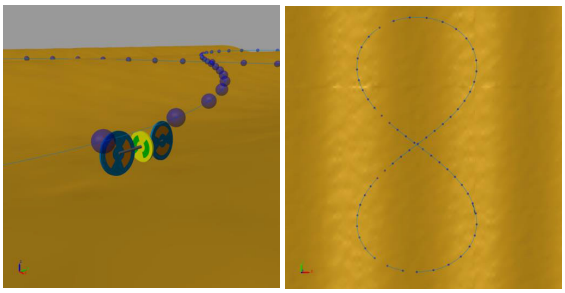

FIGURE 7. (a) RWBR on the curved pavement in Matlab Simscape. (b) Given "8"-shape path.

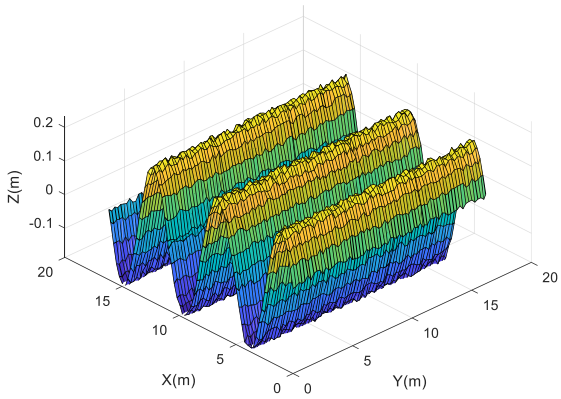

FIGURE 8. The three-dimensional surface plot with K_2=0.2.

TABLE 2. The parameters of RWBR.

	Case 1	Case 2	Case 3
$v(rad/s)$	10	10	20
$K_2(m)$	0.1	0.2	0.2

Three test cases are formed by the combination of different rear-wheel angle velocities v and terrain parameter K_2 as shown in Table 2. In addition, an overturning moment perturbation τ_p is added at the center of the RWBR. The perturbation is shown in Fig. 9.

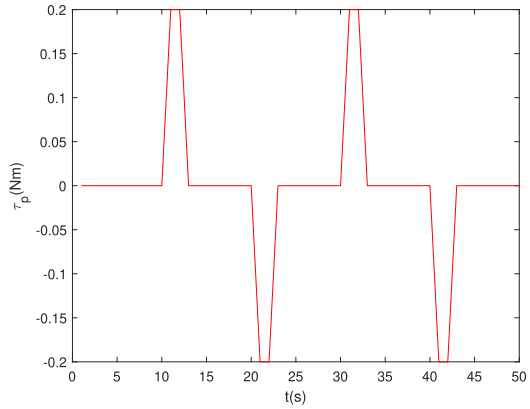

FIGURE 9. The perturbation over time.

In Simscape, the contact force models between tire and ground includes normal force f_n and friction f_f. During the contact, a contact frame is located at the contact point. The z-direction of the contact frame is an outward normal vector for the one geometry, and inward normal vector for the other. The f_n is listed as follows,which is aligned with the z-axis of the contact frame:

$$f_n = s(d)\left(k \cdot d + b \cdot d'\right) \tag{12}$$

where d is the penetration depth between two contacting geometries, d' is the first time derivative of the penetration depth, k is the normal-force stiffness specified in the block, b is the normal-force damping specified in the block and $s(d)$ is the smoothing function. As for the frictional force, the f_f is directly opposed to the direction of the relative velocity.

$$\left|f_f\right| = \mu \cdot |f_n| \tag{13}$$

where μ is the effective coefficient of friction. It is a function of the values of the coefficient of static friction, coefficient of dynamic friction, and critical velocity parameters, and the magnitude of the relative tangential velocity. The parameters used in the contact force model are chosen in [52].

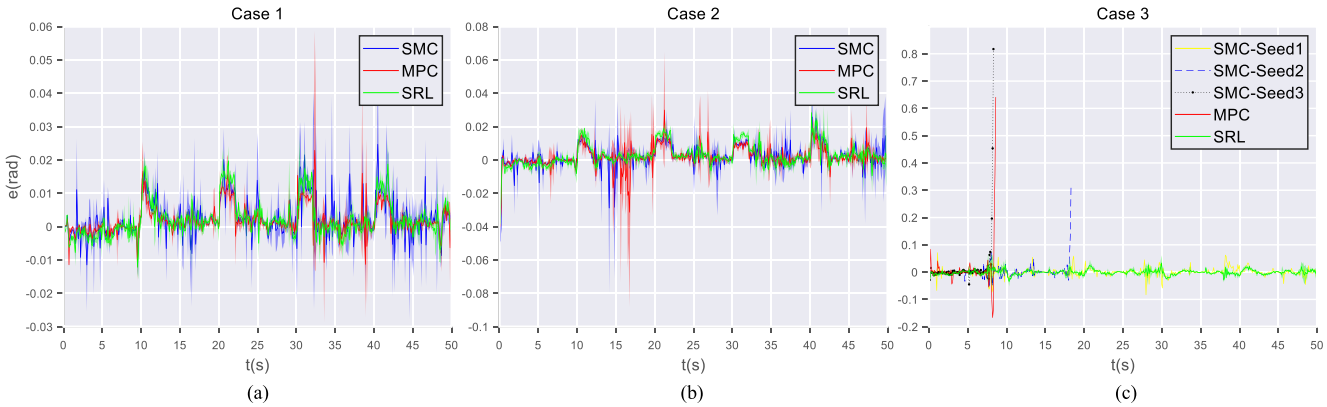

FIGURE 10. Tracking error over time.

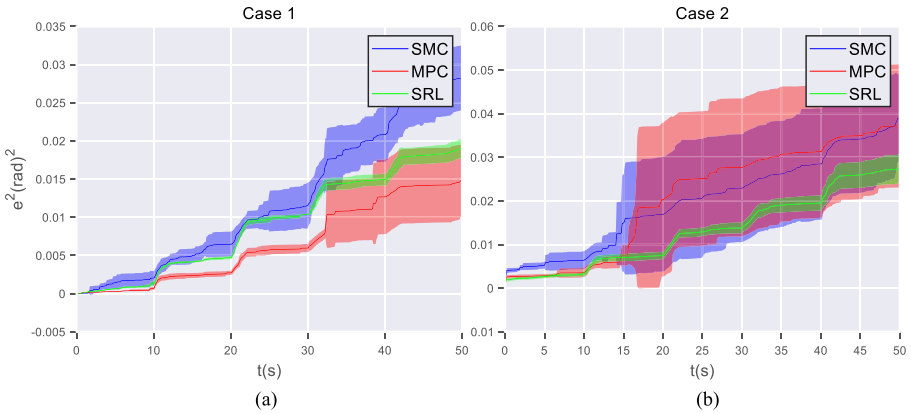

FIGURE 11. The cumulative sum of squares of tracking error over time.

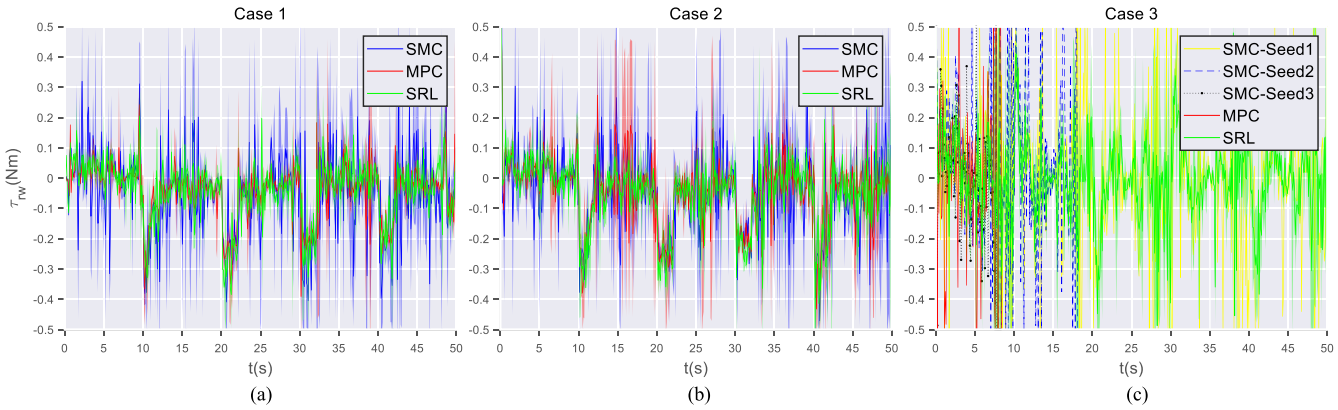

FIGURE 12. Torque of reaction wheel τ_{rw} over time.

B. EXPERIMENTAL RESULTS

To illustrate the effect of SRL in roll angle tracking control, the performances of SMC, MPC and SRL are compared in each case and each random seed. In all cases, a Stanley controller [53] is used to track the "8"-shape path. The control frequency of SRL, SMC and MPC is set to 100 Hz. The control frequency of PRL is set to 10 Hz. All cases in the experiments are set to 50 s. The parameters $\kappa^p = 10$ and $\kappa^s = [80, 10, 0.1]$ are chosen for the reward functions. The initial coefficients of all controllers are tuned to obtain an acceptable control performance. The implementation of SMC is shown in Section III-B in addition to parameters $\theta_2(0) = [30, 1, 20, 1]$. The Nonlinear MPC Controller block in Maltab is adopted to design the MPC controller, in which

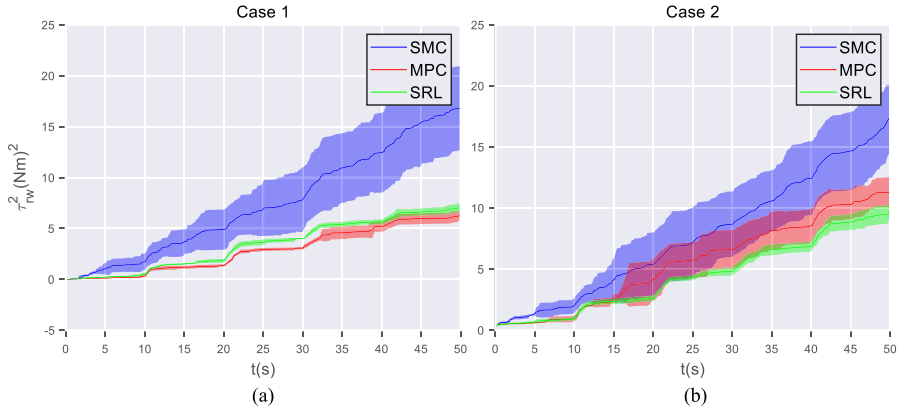

FIGURE 13. The cumulative sum of squares of torque of reaction wheel τ_{rw} over time.

FIGURE 14. Torque of reaction wheel τ_{rw} over time.

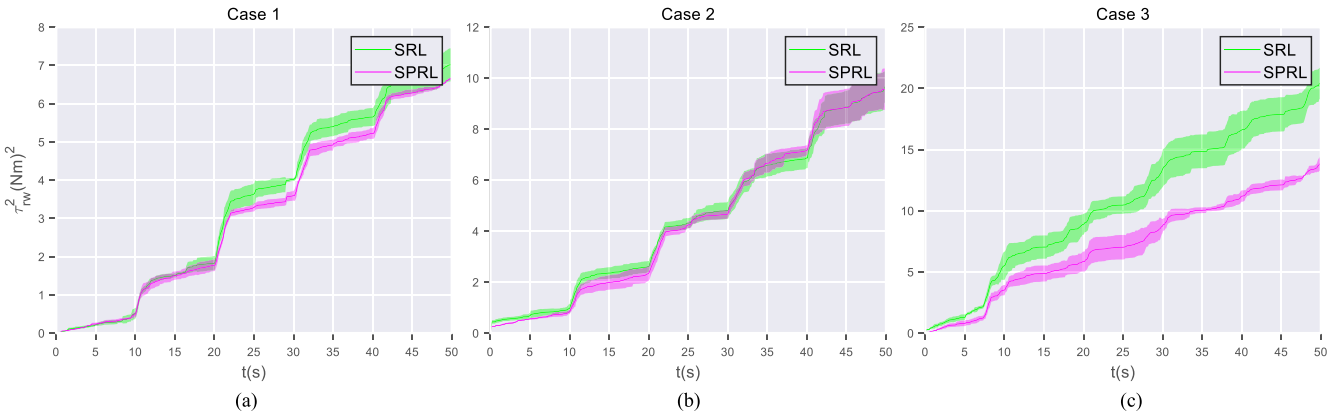

FIGURE 15. The cumulative sum of squares of torque of reaction wheel τ_{rw} over time.

the prediction model has 2 states (e, $\dot{e}$), 2 outputs (e, $\dot{e}$), and 1 input (τ_{rw}). The parameters of the MPC controller are listed as follows: the prediction horizon is 10 steps, the control horizon is 5 steps, and the tuning weights is [2, 0] for states and 0.01 for input.

1) COMPARISON OF SRL WITH SMC AND MPC

The evolution of the tracking error versus time is shown in Fig. 10. For a clearer comparison with the performance of each controller, the cumulative sum of tracking error squares is shown in Fig. 11. The solid line in the Fig. 11 shows the

TABLE 3. RMS of the four controllers (Nm).

Case	Controller	Seed 1	Seed 2	Seed 3	Mean
1	SMC	0.1590	0.1836	0.2053	0.1826
	MPC	0.1164	0.1095	0.1106	0.1122
	SRL	0.1204	0.1210	0.1141	0.1185
	SPRL	0.1150	0.1153	0.1157	0.1153
2	SMC	0.1684	0.1958	0.1948	0.1863
	MPC	0.1565	0.1407	0.1530	0.1501
	SRL	0.1393	0.1351	0.1433	0.1392
	SPRL	0.1448	0.1355	0.1342	0.1382
3	SMC	0.2456	-	-	-
	MPC	-	-	-	-
	SRL	0.1952	0.2096	0.2034	0.2027
	SPRL	0.1650	0.1722	0.1650	0.1674

mean, while the area part shows the standard deviation under different random seeds. The following conclusions can be drawn from experiments. Firstly, the RWBR with SRL is the only controller that completes the full 50 s test scenario in all of the three cases and the three seeds, which shows that it has the best robustness over the others. Secondly, in both case 1 and case 2, the tracking error of SRL is smaller than the one of SMC. In case 2, SRL has the smallest cumulative sum of squares of tracking errors among the three controllers. Finally, the standard deviation of SRL is significantly lower than the others, which also indicates that SRL improves the robustness of controllers. Similar to the tracking error, the torques and the cumulative sum of squares of torques under different controllers are also plotted in Fig. 12 and Fig. 13. The inertia wheel torque of the RWBR with SRL controller is significantly smaller than SMC. All tests exhibit that RL improves the performances of the SMC.

2) COMPARISON BETWEEN SPRL AND SRL

In order to show the influence of PRL on equilibrium compensation on RWBR control, Fig. 14 and Fig. 15 show SRL and SPRL under different cases and different seeds. In case 1 and case 2, SPRL is slightly superior to SRL. In case 3, SPRL is significantly superior to SRL. The reason for this phenomena is the higher speed of case 3. Higher velocity results in a larger approximation error when computing the target equilibrium point from the conventional analytical model, whereas the SPRL controller can reduce the approximation error through optimization. Thus, it can be concluded that the proposed SPRL controller achieves superior balancing performances, smaller inertia wheel torque and better robustness compared to the three other controllers.

For further data comparisons, the root mean square (RMS) of the torque for reaction wheel are compared, which are defined as follows.

$$RMS\left(q\left(i\right)\right) = \sqrt{\sum_{i=1}^{N} \frac{q^2\left(i\right)}{N}}. \tag{14}$$

In Table 3, we compare the RMS of the reaction wheel torque of the four controllers in all tests. Even though the average MPC in case 1 is slightly smaller than SPRL, the MPC in case 2 is larger than SPRL, and in case 3 the MPC crashed early before accomplishing the full 50 s simulation. In addition, the RMSs of SPRL are less than SRL, and the ones of SRL are less than SMC in all three cases. In conclusion, SPRL has a good balance compensation and tracking control effect for RWBR.

V. CONCLUSION

In this paper, a SPRL controller is developed for the balancing purpose of an RWBR system on a curved pavement under path tracking task with inaccurate model parameters, simplified dynamic model, matched disturbances (the dry friction of inertia wheels, etc.) and composite mismatched disturbances (including unmodeled wheel–ground contact and gust disturbances and topographically introduced periodic disturbances, etc.). By connecting FC and RL in serial-parallel, the comparison between the proposed SPRL and SMC, MPC and SRL illustrates stronger robustness and better control performance. The comparison between SPRL and SRL shows that PRL significantly reduces the control torque of inertia wheel for balancing equilibrium point compensation. The comparison between SRL and SMC shows that online adjustment of the parameters of the sliding mode surface and reaching control by serial reinforcement learning can effectively improve the control performance. In addition, MPC also serves as a comparison algorithm to demonstrate the effectiveness of SPRL.

In addition, improving the sample efficiency of the proposed SPRL will be considered as the main future research direction of this paper. The convergence of Actor Net. occurs later than that of Critic Net. according to [54]. It is not very ideal for the practicability of the algorithm and limits the effectiveness of SPRL in a rapidly changing environment or when there are more random disturbances. Therefore, our future work aims to replace the critic deep neural network with Gaussian process regression, radial basis functions, or some other components of model-based reinforcement learning to reduce the demand for data samples. Next, a real-time physical deployment is also planned.

REFERENCES

[1] H. E. Tseng and D. Hrovat, "State of the art survey: Active and semi-active suspension control," *Vehicle Syst. Dyn.*, vol. 53, no. 7, pp. 1034–1062, 2015.

[2] H. Qi, L. Ding, B. You, L. Huang, X. An, S. Li, and G. Liu, "Velocity following control of a pseudo-driven wheel for reducing internal forces between wheels," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4337–4344, Apr. 2022.

[3] S. Stasinopoulos, M. Zhao, and Y. Zhong, "Simultaneous localization and mapping for autonomous bicycles," *Int. J. Adv. Robotic Syst.*, vol. 14, no. 3, 2017, Art. no. 1729881417707170.

[4] D. Rodriguez-Rosa, I. Payo-Gutierrez, F. J. Castillo-Garcia, A. Gonzalez-Rodriguez, and S. Perez-Juarez, "Improving energy efficiency of an autonomous bicycle with adaptive controller design," *Sustainability*, vol. 9, no. 5, pp. 1–16, 2017.

[5] C.-H. Chiu and C.-Y. Wu, "Bicycle robot balance control based on a robust intelligent controller," *IEEE Access*, vol. 8, pp. 84837–84849, 2020.

[6] Y. Sun, H. Zhao, Z. Chen, X. Zheng, M. Zhao, and B. Liang, "Fuzzy model-based multi-objective dynamic programming with modified particle swarm optimization approach for the balance control of bicycle robot," *IET Control Theory Appl.*, vol. 16, no. 1, pp. 7–19, 2022.

[7] C. Badgujar and S. Mohite, "Design, analysis and implementation of control moment gyroscope (CMG) mechanism to self-balance a moped bike," *Mater. Today, Proc.*, vol. 72, pp. 1517–1523, Jan. 2023.

[8] S.-I. Lee, I.-W. Lee, M.-S. Kim, H. He, and J.-M. Lee, "Balancing and driving control of a bicycle robot," *J. Inst. Control, Robot. Syst.*, vol. 18, no. 6, pp. 532–539, Jun. 2012.

[9] Y. Zhang, J. Li, J. Yi, and D. Song, "Balance control and analysis of stationary riderless motorcycles," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2011, pp. 3018–3023.

[10] Y. Yu and M. Zhao, "Steering control for autonomously balancing bicycle at low speed," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2018, pp. 33–38.

[11] N. H. Getz and J. E. Marsden, "Control for an autonomous bicycle," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1995, pp. 1397–1402.

[12] M. Araki, K. Akimoto, and T. Takenaka, "Study of riding assist control enabling self-standing in stationary state," *SAE Int. J. Vehicle Dyn., Stability, NVH*, vol. 3, no. 1, pp. 47–56, Dec. 2018.

[13] X. Zheng, X. Zhu, Z. Chen, Y. Sun, B. Liang, and T. Wang, "Dynamic modeling of an unmanned motorcycle and combined balance control with both steering and double CMGs," *Mechanism Mach. Theory*, vol. 169, Mar. 2022, Art. no. 104643.

[14] H. Yetkin, S. Kalouche, M. Vernier, G. Colvin, K. Redmill, and U. Ozguner, "Gyroscopic stabilization of an unmanned bicycle," in *Proc. Amer. Control Conf.*, Jun. 2014, pp. 4549–4554.

[15] P. Seekhao, K. Tungpimolrut, and M. Parnichkun, "Development and control of a bicycle robot based on steering and pendulum balancing," *Mechatronics*, vol. 69, Aug. 2020, Art. no. 102386.

[16] K. He, Y. Deng, G. Wang, X. Sun, Y. Sun, and Z. Chen, "Learning-based trajectory tracking and balance control for bicycle robots with a pendulum: A Gaussian process approach," *IEEE/ASME Trans. Mechatronics*, vol. 27, no. 2, pp. 634–644, Apr. 2022.

[17] A. Owczarkowski, D. Horla, and J. Zietkiewicz, "Introduction of feedback linearization to robust LQR and LQI control—Analysis of results from an unmanned bicycle robot with reaction wheel," *Asian J. Control*, vol. 21, no. 2, pp. 1028–1040, Mar. 2019.

[18] S. R. Larimi, P. Zarafshan, and S. A. A. Moosavian, "A new stabilization algorithm for a two-wheeled mobile robot aided by reaction wheel," *J. Dyn. Syst., Meas., Control*, vol. 137, no. 1, Jan. 2015, Art. no. 011009.

[19] X. Zhu, X. Zheng, Q. Zhang, Z. Chen, Y. Liu, and B. Liang, "Natural residual reinforcement learning for bicycle robot control," in *Proc. IEEE Int. Conf. Mechatronics Autom. (ICMA)*, Aug. 2021, pp. 1201–1206.

[20] H.-W. Kim, J.-W. An, H. dong Yoo, and J.-M. Lee, "Balancing control of bicycle robot using PID control," in *Proc. 13th Int. Conf. Control, Autom. Syst. (ICCAS)*, Oct. 2013, pp. 145–147.

[21] K. Kanjanawanishkul, "LQR and MPC controller design and comparison for a stationary self-balancing bicycle robot with a reaction wheel," *Kybernetika*, vol. 51, no. 1, pp. 173–191, Mar. 2015.

[22] J. Yi, D. Song, A. Levandowski, and S. Jayasuriya, "Trajectory tracking and balance stabilization control of autonomous motorcycles," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2006, pp. 2583–2589.

[23] C.-L. Hwang, H.-M. Wu, and C.-L. Shih, "Fuzzy sliding-mode underactuated control for autonomous dynamic balance of an electrical bicycle," *IEEE Trans. Control Syst. Technol.*, vol. 17, no. 3, pp. 658–670, May 2009.

[24] V. Mnih, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, 2015.

[25] S. Aradi, "Survey of deep reinforcement learning for motion planning of autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 740–759, Feb. 2022.

[26] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Sci. Robot.*, vol. 5, no. 47, Oct. 2020.

[27] J. Randløv AND p. Alstrøm, *Learning to Drive a Bicycle Using Reinforcement Learning and Shaping*. Madison, WI, USA: ICML, Jul. 1998.

[28] S. Choi, T. Le, Q. Nguyen, M. Layek, S. Lee, and T. Chung, "Toward self-driving bicycles using state-of-the-art deep reinforcement learning algorithms," *Symmetry*, vol. 11, no. 2, p. 290, Feb. 2019.

[29] Q. Zheng, D. Wang, Z. Chen, Y. Sun, and B. Liang, "Continuous reinforcement learning based ramp jump control for single-track two-wheeled robots," *Trans. Inst. Meas. Control*, vol. 44, no. 4, pp. 892–904, Feb. 2022.

[30] S. Turakhia, "A deep reinforcement learning approach for robotic bicycle stabilization," Ph.D. dissertation, Ira A. Fulton School Eng., Arizona State Univ., Phoenix, AZ, USA, 2020.

[31] I. Carlucho, M. D. Paula, and G. G. Acosta, "An adaptive deep reinforcement learning approach for MIMO PID control of mobile robots," *ISA Trans.*, vol. 102, pp. 280–294, Jul. 2020.

[32] D. Li and J. Dong, "Fuzzy control based on reinforcement learning and subsystem error derivatives for strict-feedback systems with an observer," *IEEE Trans. Fuzzy Syst.*, early access, Dec. 9, 2022, doi: 10.1109/TFUZZ.2022.3227993.

[33] D. Li and J. Dong, "Performance-constrained fault-tolerant DSC based on reinforcement learning for nonlinear systems with uncertain parameters," *Appl. Math. Comput.*, vol. 443, Apr. 2023, Art. no. 127759.

[34] T. Johannink, S. Bahl, A. Nair, J. Luo, A. Kumar, M. Loskyll, J. Aparicio Ojea, E. Solowjow, and S. Levine, "Residual reinforcement learning for robot control," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 6023–6029.

[35] O. Dogru, K. Velswamy, F. Ibrahim, Y. Wu, A. S. Sundaramoorthy, B. Huang, S. Xu, M. Nixon, and N. Bell, "Reinforcement learning approach to autonomous PID tuning," *Comput. Chem. Eng.*, vol. 161, May 2022, Art. no. 107760.

[36] X. Zhu, Y. Deng, X. Zheng, Q. Zheng, B. Liang, and Y. Liu, "Online reinforcement-learning-based adaptive terminal sliding mode control for disturbed bicycle robots on a curved pavement," *Electronics*, vol. 11, no. 21, p. 3495, Oct. 2022.

[37] P. Xu, C. Xu, and A. Zheng, "A speed regulating controller for virtual passive walking," in *Proc. IEEE Int. Conf. Mechatronics Autom. (ICMA)*, Oct. 2020, pp. 838–843.

[38] H. Xie, X. Zhao, Q. Sun, K. Yang, and F. Li, "A new virtual-real gravity compensated inverted pendulum model and Adams simulation for biped robot with heterogeneous legs," *J. Mech. Sci. Technol.*, vol. 34, no. 1, pp. 401–412, Jan. 2020.

[39] J. He, M. Zhao, and S. Stasinopoulos, "Constant-velocity steering control design for unmanned bicycles," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2015, pp. 428–433.

[40] M. W. Spong, P. Corke, and R. Lozano, "Nonlinear control of the reaction wheel pendulum," *Automatica*, vol. 37, no. 11, pp. 1845–1851, Nov. 2001.

[41] J. Moreno-Valenzuela, C. Aguilar-Avelar, and S. Puga-Guzmán, "On trajectory tracking control of the inertia wheel pendulum," in *Proc. Int. Conf. Control, Decis. Inf. Technol. (CoDIT)*, Nov. 2014, pp. 572–577.

[42] L. An and G.-H. Yang, "Optimal transmission power scheduling of networked control systems via fuzzy adaptive dynamic programming," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 6, pp. 1629–1639, Jun. 2021.

[43] Q. Wei, R. Song, and P. Yan, "Data-driven zero-sum neuro-optimal control for a class of continuous-time unknown nonlinear systems with disturbance using ADP," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 2, pp. 444–458, Feb. 2016.

[44] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.

[45] V. Konda and J. Tsitsiklis, "Actor-critic algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 12. Denver, CO, USA, Nov. 1999, pp. 1–7.

[46] L. Chen, J. Liu, H. Wang, Y. Hu, X. Zheng, M. Ye, and J. Zhang, "Robust control of reaction wheel bicycle robot via adaptive integral terminal sliding mode," *Nonlinear Dyn.*, vol. 104, no. 3, pp. 2291–2302, May 2021.

[47] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proc. Int. Conf. Mach. Learn.*, Lille, France, 2015, pp. 1889–1897.

[48] A. Levy, R. Platt, and K. Saenko, "Hierarchical reinforcement learning with hindsight," 2018, *arXiv:1805.08180*.

[49] A. Levy, G. Konidaris, R. Platt, and K. Saenko, "Learning multi-level hierarchies with hindsight," Dec. 2017, *arXiv:1712.00948*.

[50] S. Documentation. (2022). *Simscape Model and Simulate Multidomain Physical Systems*. [Online]. Available: https://www.mathworks.com/products/simscape.html

[51] H. C. Documentation. (2022). *Nonlinear MPC Controller*. [Online]. Available: https://www.mathworks.com/help/mpc/ref/nonlinearmpccontroller.html

[52] S. Miller. (Feb. 2021). *Simscape Multibody Contact Forces Library*. Accessed: Feb. 2021. [Online]. Available: https://github.com/mathworks/Simscape-Multibody-Contact-Forces-Library

[53] S. Thrun, "Stanley: The robot that won the DARPA grand challenge," *J. Field Robot.*, vol. 23, no. 9, pp. 661–692, 2006.

[54] M. Holzleitner, L. Gruber, J. Arjona-Medina, J. Brandstetter, and S. Hochreiter, "Convergence proof for actor-critic methods applied to PPO and rudder," in *Transactions on Large-Scale Data- and Knowledge-Centered Systems XLVIII*, 2021, pp. 105–130.

XIANJIN ZHU received the B.S. degree in machinery and automation from the Wuhan University of Science and Technology, Wuhan, China, in 2012, and the M.S. degree from the School of Mechanical Engineering, Dalian University of Technology, Dalian, China, in 2016. He is currently pursuing the Ph.D. degree with the Harbin Institute of Technology, Harbin, China.

His current research interests include the design of robotics systems, the intelligent control of robots, and the application of deep reinforcement learning to robots.

YANG DENG received the B.S. degree in information and computational science (applied mathematics) and the M.S. degree in control engineering from Beihang University, Beijing, China, in 2014 and 2017, respectively, and the Ph.D. degree in automatic control engineering from École Centrale de Nantes, Nantes, France, in 2020.

Since 2021, he has been with the Department of Automation, Tsinghua University, Beijing, as a Postdoctoral Research Associate. His research interests include time-delay systems, networked control systems and their applications, and robotic systems.

XUDONG ZHENG received the Ph.D. degree in dynamics and control from Beihang University, Beijing, China, in 2019.

He was a Postdoctoral Fellow with the Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, China, from 2019 to 2022. He is currently an Associate Professor with the School of Automation, Beijing University of Posts and Telecommunications, Beijing. His current research interests include robot dynamics and control, and multibody systems dynamics.

QINGYUAN ZHENG received the B.S. degree from the School of Electrical Engineering, Chongqing University, China, in 2015. He is currently pursuing the Ph.D. degree with Tsinghua University, China.

His current research interests include reinforcement learning and the intelligent control of wheeled mobile robots.

ZHANG CHEN received the Ph.D. degree in control science and engineering from Tsinghua University, Beijing, China, in 2015.

He was a Postdoctoral Fellow with the Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, China, from 2016 to 2018. He is currently a Research Assistant Professor with the Department of Automation, Tsinghua University. His current research interests include control systems, robotics, and teleoperation.

BIN LIANG (Senior Member, IEEE) was born Jiangxi, China, in 1968. He received the B.Sc. and M.Sc. degrees in control engineering from Northwestern Polytechnical University, Xi'an, China, in 1988 and 1991, respectively, and the Ph.D. degree in precision instruments from Tsinghua University (THU), Beijing, China, in 1994. He was recommended for admission to the 1st "Educational Reform Pilot Class" with Northwestern Polytechnical University. From December 1994 to October 2007, he was a Researcher with China Aerospace Science and Technology Corporation (CASC). In 2007, he joined THU, where he is currently a Professor with the Department of Automation and the leading Director of the Institute of Navigation and Control. His research interests include robotics, teleoperation, and intelligent control.

YU LIU received the Ph.D. degree in mechatronics engineering from the Harbin Institute of Technology, Harbin, China, in 2004.

From September 2007 to December 2013, he was an Associate Professor with the School of Mechatronics Engineering, Harbin Institute of Technology, where he has been a Professor, since 2013. His research interests include space robot, parallel robot, mobile robot, theory of mechanism, and intelligent control.