mathematics

MDPI

Article

Combining Prior Knowledge and Reinforcement Learning for Parallel Telescopic-Legged Bipedal Robot Walking

Jie Xue [1,2,†], Jiaqi Huangfu [2,†], Yunfeng Hou [2,*] and Haiming Mou [2]

[1] School of Optical-Electrical and Computer Engineering, University of Shanghai for Science and Technology, Shanghai 200093, China; 221240078@st.usst.edu.cn
[2] Institute of Machine Intelligence, University of Shanghai for Science and Technology, Shanghai 200093, China; 223332488@st.usst.edu.cn (J.H.); 201440059@st.usst.edu.cn (H.M.)
* Correspondence: yunfenghou@usst.edu.cn
† These authors contributed equally to this work.

Abstract: The parallel dual-slider telescopic leg bipedal robot (L04) is characterized by its simple structure and low leg rotational inertia, which contribute to its walking efficiency. However, end-to-end methods often overlook the robot's physical structure, leading to difficulties in maintaining the parallel alignment of the dual sliders, which in turn compromises walking stability. One potential solution to this issue involves utilizing imitation learning to replicate human motion data. However, the dual telescopic leg structure of the L04 robot makes it difficult to perform motion retargeting of human motion data. To enable L04 walking, we design a method that integrates prior feedforward with reinforcement learning (PFRL), specifically tailored for the parallel dual-slider structure. We utilize prior knowledge as a feedforward action to compensate for system nonlinearities; meanwhile, the feedback action generated by the policy network adaptively regulates dynamic balance and, combined with the feedforward action, jointly controls the robot's walking. PFRL enforces constraints within the motion space to mitigate the chaotic behavior of the parallel dual sliders. Experimental results show that our method successfully achieves sim2real transfer on a real bipedal robot without the need for domain randomization techniques or intricate reward functions. L04 achieves omnidirectional walking with minimal energy consumption and exhibits robustness against external disturbances.

Keywords: parallel dual-slider structure; bipedal robot; prior knowledge; reinforcement learning; sim2real

MSC: 68T05

check for updates

Academic Editor: Héctor Vázquez-Leal

Received: 7 February 2025
Revised: 5 March 2025
Accepted: 14 March 2025
Published: 16 March 2025

Citation: Xue, J.; Huangfu, J.; Hou, Y.; Mou, H. Combining Prior Knowledge and Reinforcement Learning for Parallel Telescopic-Legged Bipedal Robot Walking. *Mathematics* **2025**, *13*, 979. https://doi.org/10.3390/math13060979

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The leg design of bipedal robots is mainly divided into bent-knee and telescopic [1,2], as shown in Figure 1. The straight-leg type draws on the results of biological evolution, imitating the human gait, and partly refers to the way birds move. It increases the degree of freedom of movement and allows the robot to adapt to different speeds and terrains, while the passive oscillation of the knee joint helps energy recovery. However, the straight-leg structure also introduces a larger torsional inertia, which increases control difficulty and leads to higher energy consumption of the knee motors [3,4]. In contrast, the telescopic leg uses a simplified joint design, driven primarily through the hip, with the knee function replaced by a linear telescopic leg [1]. Although it is not bionic in appearance, the control is more direct and reduces dynamic loading for simpler gait planning [5,6]. Telescopic

joints are less common in the biological world but are easier to implement in engineering, making robot leg length adjustment more efficient and meeting the need for simplified dynamics models. Control methods for telescopic-legged robots have made significant progress in recent research, with the central goal of balancing dynamic performance and stability. Approaches based on model simplification have been widely adopted, such as linear inverted pendulum models and single rigid body models, by simplifying multi-body dynamics for real-time gait planning [1]. In telescopic-legged robots such as Slider, the sliding joint design significantly reduces the leg inertia, which, combined with the center of mass *COM* trajectory generated by the linear inverted pendulum model and the feedback control, effectively reduces the vertical motion deviation and improves the walking stability. For highly dynamic motion, the study further proposes an inertia shaping model to adjust the *COM* inertia by optimizing the leg configuration, which enhances jumping efficiency and attitude control [7]. The combination of trajectory optimization algorithms (e.g., multi-stage nonlinear programming) and whole-body control makes it possible to generate and execute complex movements (e.g., twisting jumps), while contact detection methods based on mean spatial velocity are introduced to achieve soft landing in combination with force control strategies. In the L04 robot, the dual-slider telescopic leg design further optimizes the dynamic characteristics by simulating knee bending and leg height adjustment. The split-coupled design of its hip joint reduces the number of motors, lowering energy consumption and cost [8]. With a linear inverted pendulum model and angular momentum trajectory planning, the L04 robot achieves stable forward and lateral walking. These conventional methods rely on accurate state observation, and the methods in the literature [8] do not enable steered walking.

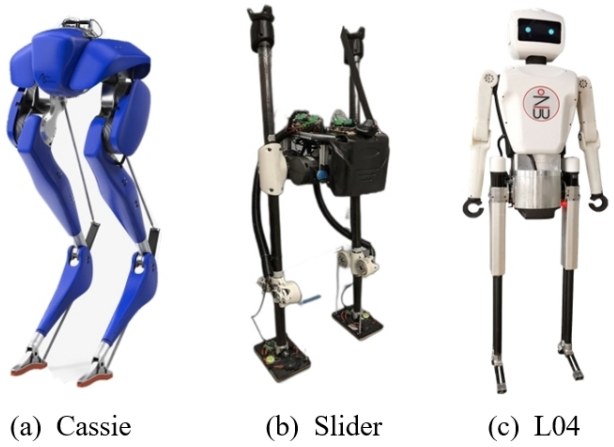


Figure 1. The figure shows, from left to right, the bent-knee robot (Cassie), the telescopic-legged robot (Slider), and the parallel dual-slider telescopic-legged robot (L04). Unlike the other two robots, L04 has no motors on its legs.

With recent advancements in RL, an increasing number of researchers are investigating its application to address the locomotion challenges of bipedal robots. RL is a machine learning paradigm in which an agent learns optimal behavior through interactions with its environment [9]. The agent explores various behaviors to maximize cumulative rewards, thereby learning to operate optimally in complex environments. This data-driven approach obviates the need for manual controller design in bipedal robots [10]. Current learning frameworks for bipedal locomotion are predominantly categorized into two approaches: reference trajectory imitation and cyclic gait encouragement [11,12]. Trajectory imitation necessitates high-quality data and the formulation of an imitation reward function. This approach incentivizes the robot to adhere to the reference data. Xue et al. combined motion

imitation with task-specific objectives [11]. The robot is trained using motion capture data, yielding high-quality and robust movements. This approach further facilitates a wider array of complex maneuvers, such as dynamic rotations, kicks, and flips with intermittent ground contact. Li et al. developed robust walking controllers by emulating reference motions and validated their performance on the Cassie robot at varying speeds [13]. In [14], a framework that emulates human motion data allowed a bipedal robot to effectively learn environment-appropriate movements. The approach of promoting periodic walking entails guiding the robot to adopt a periodic gait through the use of periodic signals. Jonah et al. utilized an RL-based approach to analyze Cassie's walking gait by designing a reward function grounded in clock signals, thereby achieving a periodic gait [12]. In [15], curriculum learning was introduced to enhance periodic walking. Omnidirectional walking in bipedal robots was accomplished through the progressive escalation of task difficulty. Wu et al. expedited the training of periodic gaits on the Cassie robot by utilizing custom sinusoidal curves [16]. In [17], periodic joint references were generated using state machines and cosine curves, enabling bipedal walking through a hybrid approach combining feedforward and RL techniques. However, trajectory imitation necessitates extensive high-quality data [18]. Acquiring such data is challenging, and the quality of training is highly dependent on it. Promoting periodic walking may lead to unnatural gait patterns, including persistently open lateral joints or one foot consistently positioned ahead of the other. None of these methods take into account the special structure of the L04, which makes it difficult to maintain the parallel attitude of the dual telescopic rods. The modeled data also cannot be redirected to the mechanical structure of the L04.

To address the aforementioned issues, our contributions are as follows:

1. We propose a novel walking learning framework that synergistically combines prior feedforward knowledge with RL.
2. We derive the forward and inverse kinematics of the parallel dual-slider telescopic leg robot and design prior knowledge and feedforward actions.
3. We implement sim2real for physical L04 and design smooth regularization terms to optimize the sim2real effect.

2. Background

The L04 robot is depicted in Figure 2. The robot comprises 17 joints and is structured into four main components: the head, arms, hips, and legs [8]. The head incorporates two rotary joints, each arm is equipped with three, the hips accommodate five, and each leg features two linear joints. All joints are independently driven by motors. The left and right hip traversing roll joints utilize a motor-driven split configuration. Similarly, the left and right hip steering joints adopt a split-pair design. Four motors are integrated within the hip region. The coupled split design improves efficiency and minimizes energy consumption, thereby enhancing the robot's overall performance.

The control relationship between the linear drive and the corresponding joint angles within the linkage-driven mechanism is depicted in Figure 2. The lateral bifurcation five-bar linkage mechanism comprises the crossbar A, upper linkage B, lower linkage C, bottom linkage D, and a central axis, which is implemented as a rolling screw rod. The crossbar A moves vertically along the central axis, causing the linkage C to swing laterally. The rotational, linear drive at the base moves the crossbar A along the bottom centerline, enabling forward and backward motion, which in turn drives the linkage C to produce hip rotational swinging. The lateral screw rod, oriented vertically downward and perpendicular to the horizontal rotational screw axis, facilitates the synchronous coupling of lateral and steering movements in the hip of bipedal robots. Furthermore, this design ingeniously integrates all degrees of freedom functionalities into a compact structure,

enabling hip rotation and lateral swinging through a dual-coupling bifurcation mechanism, thereby significantly reducing power consumption during robot locomotion.

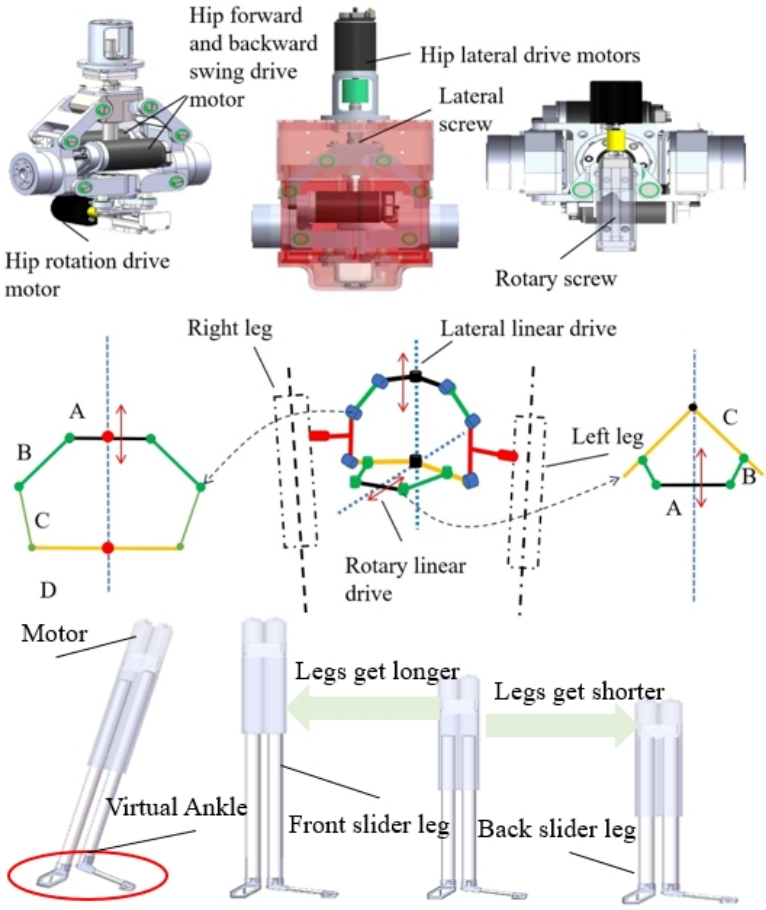


Figure 2. Mechanical structure diagram of the L04 robot's hip and legs. The **top** of the figure shows the mechanical structure of the L04. The **middle** section illustrates the principles of lateral split, dual-coupled split, and rotational division. The **bottom** section depicts the motion pattern of the parallel dual-slider mechanism. The motion of the dual sliders simulates the effects of knee flexion and ankle articulation.

3. Design of Prior Knowledge

Conventional RL methods utilize the representational power of neural networks to develop walking strategies in an end-to-end fashion. Most data-driven walking approaches neglect the robot's underlying physical principles and prior knowledge, instead focusing on training policies end-to-end, from sensor data to joint position commands. However, such data-driven methods necessitate meticulously designed reward functions and substantial training time, yet frequently fail to produce natural and graceful gaits. To enable bipedal robots to generate natural and graceful walking patterns, researchers commonly design imitation-based rewards, incentivizing robots to replicate a predefined reference motion. Conventional methods for generating reference motions include video-based learning or the utilization of costly motion capture systems. However, the dual-telescopic-leg structure of the L04 poses significant challenges in retargeting human motion data. Therefore, we fully leverage the robot's prior knowledge and integrate it with RL to achieve stable walking.

We use the Cartesian coordinate system based on the midpoint $O(0, 0, 0)$ of the line AA' connecting the points of lateral symmetric rotation of the hip, as shown in Figure 3. The COM is positioned 0.17 m directly above O. This setting is based on the actual physical structure and mass distribution of the L04 robot. The positive direction of the X-axis is

defined as the forward positive direction of the robot. The positive direction of the Y-axis is defined as the positive direction in which the robot moves to the left. The positive direction of the Z-axis is defined as the direction in which the robot's center of mass moves vertically downward. The vertical distance from the origin O to the ground is set to h, and the hip split-rotation joint angles are labeled as j_{tl} and j_{tr}. The pair-fractional structure of the robot's hip joint design ensures that j_{tl} and j_{tr} are equal. Point A and point A' are the centers of rotation of the left lateral and right hip joints, respectively. The corresponding lateral joint rotation angles are labeled j_{sl} and j_{sr}. Due to the nature of the pairwise split structure design, $j_{sl} = j_{sr}$. The left key points O, A, B, and C are always in the same plane. Similarly, the right side points O, A', B', and point C' are always in the same plane. The rotation angles of the right and left hip joints are denoted as j_{hl} and j_{hr}, respectively. The l_L and l_R denote the lengths of the left and right legs, respectively. In addition to the changes in leg length, the length of the hip linkage is fixed. OA represents the horizontal distance from the steering center O to the rotation center A of the hip lateral rotation joint, with a length of a; AB represents the vertical distance from the rotation center A of the hip lateral rotation joint to the rotation center C of the hip joint, with a length of b; BC represents the horizontal distance from the rotation center A of the hip lateral rotation joint to the rotation center C of the hip joint, with a length of c.

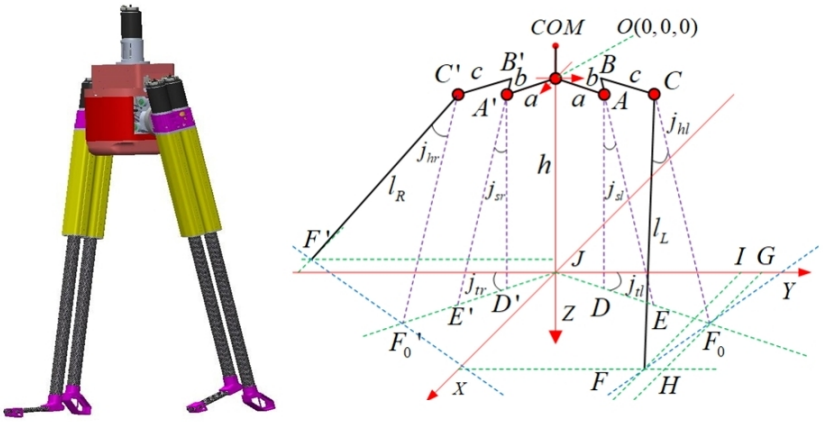


Figure 3. The spatial schematic diagram of the kinematic analysis of L04.

Forward kinematics refers to determining the position and orientation of the virtual leg end effector in Cartesian space given the known joint angles (j_t, j_s, j_h, j_a) in joint space. Here, the geometric method of solution is employed. For the left leg, assuming HG is parallel to the X-axis. Let the length of the virtual leg be l_L, then the position in Cartesian space along the X, Y, and Z directions for the left leg end-point F can be respectively expressed as $F_x = HG$, $F_y = HG$, $F_z = AD$.

$$JF_0 = a + l_L \times \sin(j_{sl}) + \frac{c}{\cos(j_{sl})} \tag{1}$$

$$GF_0 = JF_0 \times \sin(j_{tl}) \tag{2}$$

$$FF_0 = l_L \times \sin(j_{hl}) \tag{3}$$

$$HF_0 = FF_0 \times \cos(j_{tl}) \tag{4}$$

$$\begin{aligned} F_x =& HG = HF_0 + GF_0 = FF_0 \times \cos(j_{tl}) + JF_0 \times \sin(j_{tl}) \\ =& l_L \times \sin(j_{hl}) \times \cos(j_{tl}) + (a + l_L \times \sin(j_{sl}) + \frac{c}{\cos(j_{sl})}) \times \sin(j_{tl}) \end{aligned} \tag{5}$$

$$\begin{aligned} F_y =& JI = JG - HF = JF_0 \times \cos(j_{tl}) - FF_0 \times \sin(j_{tl}) \\ =& (a + l_L \times \sin(j_{sl}) + \frac{c}{\cos(j_{sl})}) \times \cos(j_{tl}) - l_L \times \sin(j_{hl}) \times \sin(j_{tl}) \end{aligned} \tag{6}$$

$$F_z = AD = l_L \times \cos(j_{sl}) \tag{7}$$

Combined with the virtual ankle angle θ formed by the front and back legs, here defined as $j_a = \theta$, the pose at the end point F of the virtual leg can be derived:

$$F_{pitch} = \arcsin(\sin(j_a - j_{hl}) - \cos(j_{sl})) \tag{8}$$

$$F_{yaw} = \arctan(\tan(j_a - j_{hl}) \times \sin(j_{sl})) \tag{9}$$

Virtual leg inverse kinematics refers to solving the joint angles (j_t, j_s, j_h, j_a) of each joint with the known position and pose of the virtual leg end F in Cartesian space. To solve the virtual leg inverse kinematics, the forward kinematics is used to find the equational constraint equations containing the known position information $(F_x, F_y, F_z, F_{pitch}, F_{yaw})$ of the virtual leg end F point and the steering j_t variables:

$$0 = \tan(j_s) \times \tan(F_{pitch}) + \sin(j_t - F_{yaw}) \tag{10}$$

Newton's iterative method is used to solve for j_t and j_s, which are brought into the equation to solve for the steering angle j_h and the virtual ankle joint j_a.

$$j_h = \arctan(-\frac{F_x}{\cos(j_s)} + c \times \tan(j_s - b)) \tag{11}$$

$$j_a = \arcsin(\sin(F_{pitch}) + \cos(j_s)) + j_h \tag{12}$$

$$l_L = \frac{F_z}{j_s} \tag{13}$$

Using the above derivation, the swing angle j_h of the hip joint, the angle j_a of the virtual ankle joint, and the geometric length l_L of the virtual leg can be obtained.

By analyzing fundamental human walking patterns, we observe that the foot's position relative to the COM follows a sinusoidal trajectory. Human-like alternating foot-lifting gaits are achieved by regulating the foot placement of the virtual leg using a sinusoidal function. The frequency and amplitude of the sinusoidal function are dynamically adjusted to control the foot-lifting motion during walking. However, relying exclusively on the sinusoidal strategy may lead to instability during foot contact and lift-off, including a bouncing effect. To mitigate this issue, we incorporate a double-support phase into the sinusoidal function. During the double-support phase, both feet maintain contact with the ground, ensuring seamless gait transitions. A sinusoidal function regulates the Z-axis variations of the foot landing points. The foot-lifting height is dynamically calculated and adjusted in real time according to the gait cycle phase.

$$z_{ref_1} = z_{max} - max(0, z\sin(\frac{2\pi}{T}t + \phi_0) - \triangle z) \tag{14}$$

$$z_{ref_2} = z_{max} - max(0, z\sin(\frac{2\pi}{T}t + \phi_0 + \pi) - \triangle z) \tag{15}$$

where z_{max} is the maximum position that the virtual leg can reach in the Z-direction; $z_{max} - z + \triangle z$ is the minimum position that the virtual leg can reach in the Z-direction; and the $-\triangle z$ term is a parameter used to adjust the duration of the double-supported standing phase.

During the foot lifting phase, the COM undergoes an offset in the y-direction towards the supporting foot. As the foot makes contact with the ground, the COM offsets back towards the origin in the Y-direction. In the frame of the COM, the changes in the foot's landing points in the Y-direction also conform to a sinusoidal curve, exhibiting the same periodic characteristics as the changes in the Z-direction of the foot's landing points.

$$y_{ref_1} = y_{ref_2} = -max(0, y\sin(\frac{2\pi}{T}t + \phi_0 + \pi) - \triangle y) \tag{16}$$

Feedforward control has enabled the robot to perform alternating leg movements while airborne. However, upon landing, the robot can only maintain stationary, stepping briefly before it falls. This issue arises because the feedforward reference is unable to accommodate changes in body posture. In the initial state, there is the problem of not being able to distinguish which foot to swing first. If a certain foot is fixed to swing first, it will cause the problem of a single gait. Therefore, our method of generating t in the reset phase is

$$t = \begin{cases} 0, & random > 0.5 \\ \omega \times T, & otherwise, \end{cases} \tag{17}$$

4. Walking Learning Framework Design

During the leg-lifting phase of L04, it is crucial to maintain consistent ground clearance for the bottoms of the dual telescopic legs on the supporting leg to ensure a natural and aesthetically pleasing gait. End-to-end methods typically rely on reward functions to encourage uniform ground clearance of the parallel telescopic legs. However, this approach does not guarantee the desired policy and significantly increases exploration time. To enable bipedal robots to achieve natural and graceful walking patterns, researchers often design imitation-related rewards to encourage robots to mimic a given reference motion. Common methods for generating reference motions include video-based learning or the use of expensive motion capture systems. However, the dual-telescopic leg structure poses challenges for retargeting human motion data. To address these issues, we designed an L04 walking learning method based on PFRL. The PFRL method integrates feedforward and feedback mechanisms, as shown in Figure 4. This method extends traditional neural network-based RL by introducing a novel feedforward control component. The feedforward control operates independently and in parallel with the neural network. This design enables the simultaneous processing of prior knowledge and feedback information, thereby enhancing the robustness of the entire framework. PFRL ensures the aesthetic quality of the parallel telescopic legs during the swing phase using a virtual leg model while also reducing the action space. Notably, we set the virtual leg's F_{pitch} to zero. This not only keeps the foot parallel to the ground in real time but also avoids excessive and unnecessary exploration of the ankle joint. Unlike common gait learning methods, our policy network outputs foot placement positions. The robot's actions are derived from the sum of the feedforward reference and the policy's foot placement output, which are converted into joint positions through the virtual leg model.

4.1. Policy Construction

The state space S_t of the policy network input is defined as follows:

$q_f \in \mathbb{R}^{10\times1}$: feedforward landing points are converted to joint positions by virtual leg models.

$\Phi \in \mathbb{R}^{2\times1}$: the roll and pitch of the body's COM.

$\omega \in \mathbb{R}^{3\times1}$: the roll, pitch, and yaw rate of the body's COM.

$q_t \in \mathbb{R}^{10\times1}$: current joint position of the robot's lower body.

$\dot{q}_t \in \mathbb{R}^{10\times1}$: current joint velocity of the robot's lower body.
$phase \in \mathbb{R}^{2\times1}$: phase signal.
$a_{t-1} \in \mathbb{R}^{5\times1}$: last action output by the policy.
$cmd \in \mathbb{R}^{3\times1}$: speed command.

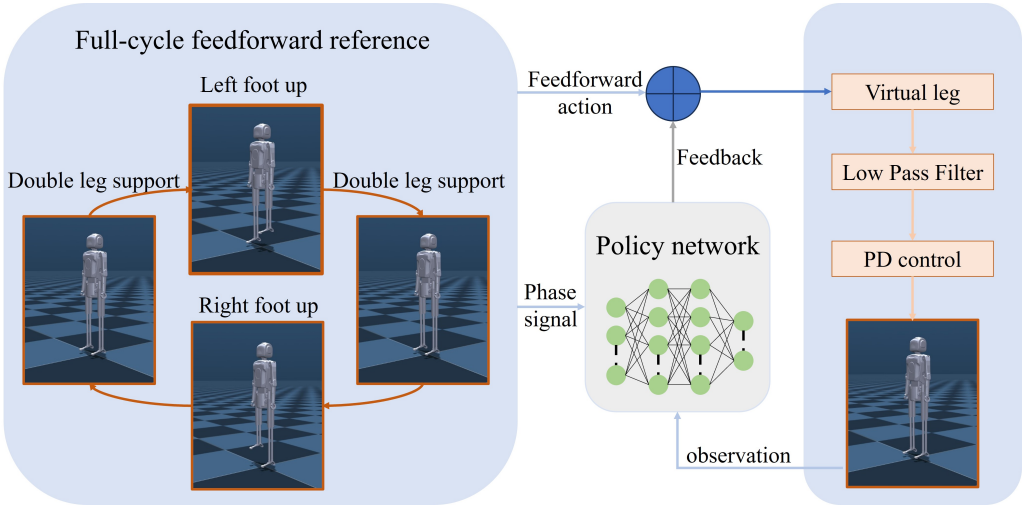


Figure 4. Framework diagram of PFRL. The PFRL framework employs a weighted summation between the feedforward reference and the landing point of the policy output, which is solved by the virtual leg model for the joint position. The feedforward and policy networks operate at 40 Hz, while the virtual leg model, filter, and PD controller operate at 1000 Hz.

Unlike common end-to-end approaches, the output of the virtual leg model-based framework is a six-dimensional vector representing the real-time position of the robot's feet. This foot position is relative to the center of the mass coordinate system. It is worth noting that due to the structural advantages of the L04, the left and right lateral target positions of the policy network outputs are set to the same value, so the action space contains only five vectors.

$$a_t = (p_x^l, p_y^l, p_z^l, p_x^r, p_z^r, hip_z) \tag{18}$$

The policy network outputs the locations of the landing points and then expands to conform to the hip pair division mechanism, i.e., $(p_x^l, p_y^l, p_z^l, p_x^r, p_y^r, p_z^r)$. Note that p_y^l and p_y^r are equal. The extended landing points are added to the feedforward landing points, and the joint target position is solved by the virtual leg model. The steering joints are learned directly by the end-to-end method. The target positions of the left and right steering joints are equal in real time. The strategy network outputs target landing point positions at 40 Hz. The solved target position of the virtual leg model is filtered by a low-pass filter and fed to the PD controller running at a frequency of 1000 Hz. The sampling period of the low-pass filter is 0.001 s, and the cutoff frequency is 4 Hz. The PD controller uses a low-stiffness controller output torque to control the biped robot's walking through a position control method. We use position control to control biped robot walking. With the PD controller, the target position is converted into motor torque. This is calculated as $\tau = k_p \times (q_{tar} - q) + k_d \times (\dot{q}_{tar} - \dot{q})$. The $\dot{q}_{tar}$ is set to 0.

4.2. Rewards

Most of the existing studies have established complex reward engineering to constrain the training process. However, this also brings a large number of reward function coefficients, which further adds difficulty to the training process. In order to maximize the demonstration of the effect of structure on sim2real results, we implemented the walking motion using only three of the most common reward functions used in bipedal training.

These three reward functions are velocity tracking reward, stability reward, and periodic foot lift reward.

The velocity tracking reward is

$$r_v = \exp(-4||cmd_{xy} - v_{xy}||^2) + 0.5\exp(-4||cmd_\omega - \omega_z||^2) \tag{19}$$

where, cmd_{xy} is the robot's velocity command in x and y directions, v_{xy} is the robot's linear velocity in x and y directions during the actual walking process, cmd_ω is the robot's angular velocity command, and ω_z is the robot's actual angular velocity in z direction.

The stability reward is

$$r_\omega = -0.015(\omega_x^2 + \omega_y^2) \tag{20}$$

where ω_x and ω_y are the roll and pitch of the robot's COM, respectively.

In order to ensure that the virtual leg length change can conform to the characteristic of periodic change and change within a reasonable range, a periodic reward mechanism is designed based on the virtual leg model. The purpose of this mechanism is to guide the robot during the walking process so that the change in its virtual leg length can follow a predetermined periodic pattern and, at the same time, to ensure that the change in the leg length does not exceed a reasonable range. Specifically, the periodic reward is defined as

$$r_p = exp(-40\sum_{i=1}^{2}(l_{ref_i} - l_{foot_i})^2) \tag{21}$$

where l_{foot_i} is the virtual leg leg length of the current feedforward and feedback summed. The reward function is designed as

$$r = 0.3 * r_v + 0.2 * r_\omega + 0.5 * r_p \tag{22}$$

4.3. Smooth Regularization

In order to achieve the real-virtual migration of the policy, Reference [19] improves the robustness by making the action smoother after being disturbed by noise through regularization techniques. Compared with drones, bipedal robots have greater noise and external interference, such as fluctuations in IMU data when the robot touches the ground and disturbances from different terrains. Therefore, using only the CAPS method [19], bipedal robots will still transform noise interference into small-range oscillations of joint positions. However, this method can suppress large-range mutations of movements, thereby protecting the hardware. The regularization term in CAPS is divided into the abrupt change smoothing term L_T and the oscillation smoothing term L_S. L_T requires that the current action be similar to the next action under the same policy. On the other hand, L_S demands that the mapping states of similar input operations be similar under the same policy. The regularization term used in the CAPS method is

$$L_T = D_T(\pi_\theta(s_t), \pi_\theta(s_{t+1})) \tag{23}$$

$$L_S = D_S(\pi_\theta(s_t), \pi_\theta(\bar{s}_t)) \tag{24}$$

$$D_T(\pi_\theta(s_t), \pi_\theta(s_{t+1})) = \|\pi_\theta(s_t) - \pi_\theta(s_{t+1})\|^2 \tag{25}$$

$$D_S(\pi_\theta(s_t), \pi_\theta(\bar{s}_t)) = \|\pi_\theta(s_t) - \pi_\theta(\bar{s}_t)\|^2 \tag{26}$$

where $\bar{s}$ is from a distribution around s.

In contrast to the CAPS approach, we define the noise as a spherical space $\mathbb{B}_d = \{\tilde{s} : d(s, \tilde{s}) \leq \epsilon\}$. The radius of this spherical space $\mathbb{B}_d$ is ϵ. $d(s, \tilde{s})$ is computed by the L2 paradigm, i.e., $d(s, \tilde{s}) = ||s - \tilde{s}||^2$. All states in the spherical perturbation space should

have similar strategies π. However, we cannot compute strategies for all states in the space. Therefore, we pick the most variable $\tilde{s}$ in $\mathbb{B}_d$ by

$$\tilde{s} = \arg \max_{\hat{s} \in \mathbb{B}_d(s,\epsilon)} \mathcal{D}(\pi_\theta(s), \pi_\theta(\tilde{s})) \tag{27}$$

where $D(\pi_\theta(s), \pi_\theta(\tilde{s}))$ is the distributional measure of the two strategies under noise perturbation. We use the L2 paradigm to compute D.

Restricting the fluctuations of the policies $\pi_\theta(s_{t+1})$ and $\pi_\theta(s_t)$ to the next moment only cannot suppress the small oscillations caused by the noise of the bipedal robot. Therefore, we also add a computation of the fluctuations between $\tilde{s}$ and s_{t+1}. We suppress small-scale oscillations in the strategies by limiting the fluctuations between the three policies. The state-valued function V_θ will become unstable when the training exploration process is perturbed by state noise. We impose a smoothing constraint on the state-valued function V_θ to increase the robustness to noise.

$$L_V = \|\pi_\theta(\tilde{s}) - 2 * \pi_\theta(s_{t+1}) + \pi_\theta(s_t)\|^2 + ||V_\theta(s) - V_\theta(\tilde{s})||^2 \tag{28}$$

The total noise robust regularization term is

$$L = \lambda_T L_T + \lambda_S L_S + \lambda_V L_V \tag{29}$$

where λ_T, λ_S, and λ_V are regularization weights.

5. Experiments

We utilize the Proximal Policy Optimization (PPO) algorithm to train a bipedal robot for locomotion. The policy network comprises two components: an actor and a critic, both implemented as multi-layer perception (MLP) networks. Due to the reduced output dimensionality, the actor and critic networks employ a compact 128 $\times$ 128 architecture. Notably, no randomization is applied during the training process. Our method is also compared with a commonly used bipedal walking learning framework (baseline) [20]. To enable a quantitative comparison, we disable the history stacking in the baseline.

5.1. Omnidirectional Walking Test

The rewards during training are shown in Figure 5. Our method starts converging at around 1200 rounds. The baseline method, on the other hand, takes about 7800 rounds. As a priori knowledge is introduced, the convergence speed of our method is improved by 84.6%. The final reward value is also improved by 7.7%.

We first test the omnidirectional traveling on a flat surface. The omnidirectional walking is divided into forward walking, side walking, and steering walking. The cycle command is set to 36, and *cmd* is set to [0.6,0,0] for forward travel. The *cmd* for sideways travel is set to [0,0.3,0]. The *cmd* for steering walk is set to [0,0,0.6]. Omnidirectional walking on flat ground is shown in Figure 6. A total of 20 policies are employed for the sim2real conversion test for each of the aforementioned walking types. The aforementioned policies are all saved from the same training session. The success rate of migration is shown in Table 1. Both forward and lateral walking achieve a 100% success rate while turning reaches a 90% success rate. The success rates of the baseline method are 85%, 90%, and 70%, respectively. The success rate of our method walk is higher than the baseline method. The fluctuation of the Euler angle roll of L04 during a forward walk is shown in Figure 7. The variance of our method is 0.0000438, and the variance of the baseline method is 0.000498. Our method of walking is more stable.

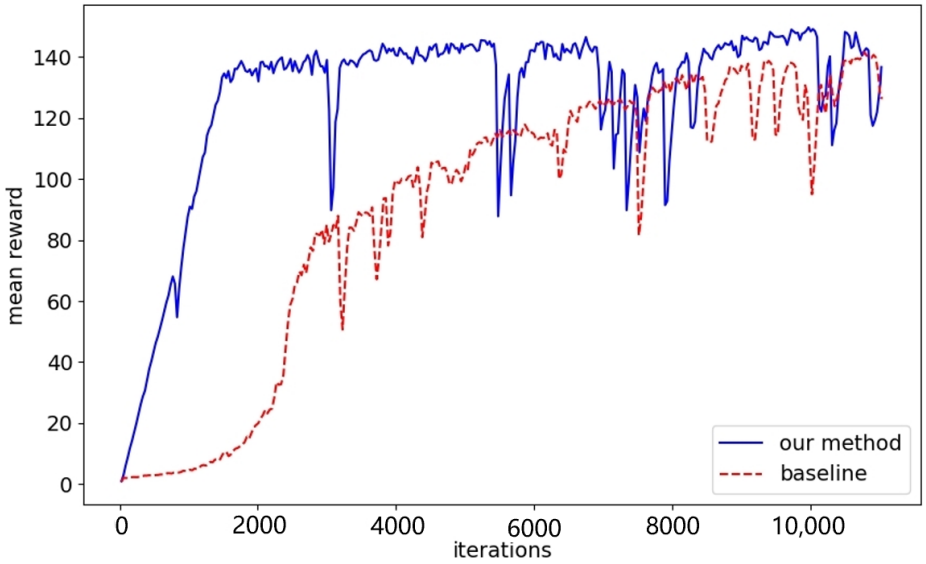


Figure 5. Reward plots of different methods. Blue is our method and red is baseline.

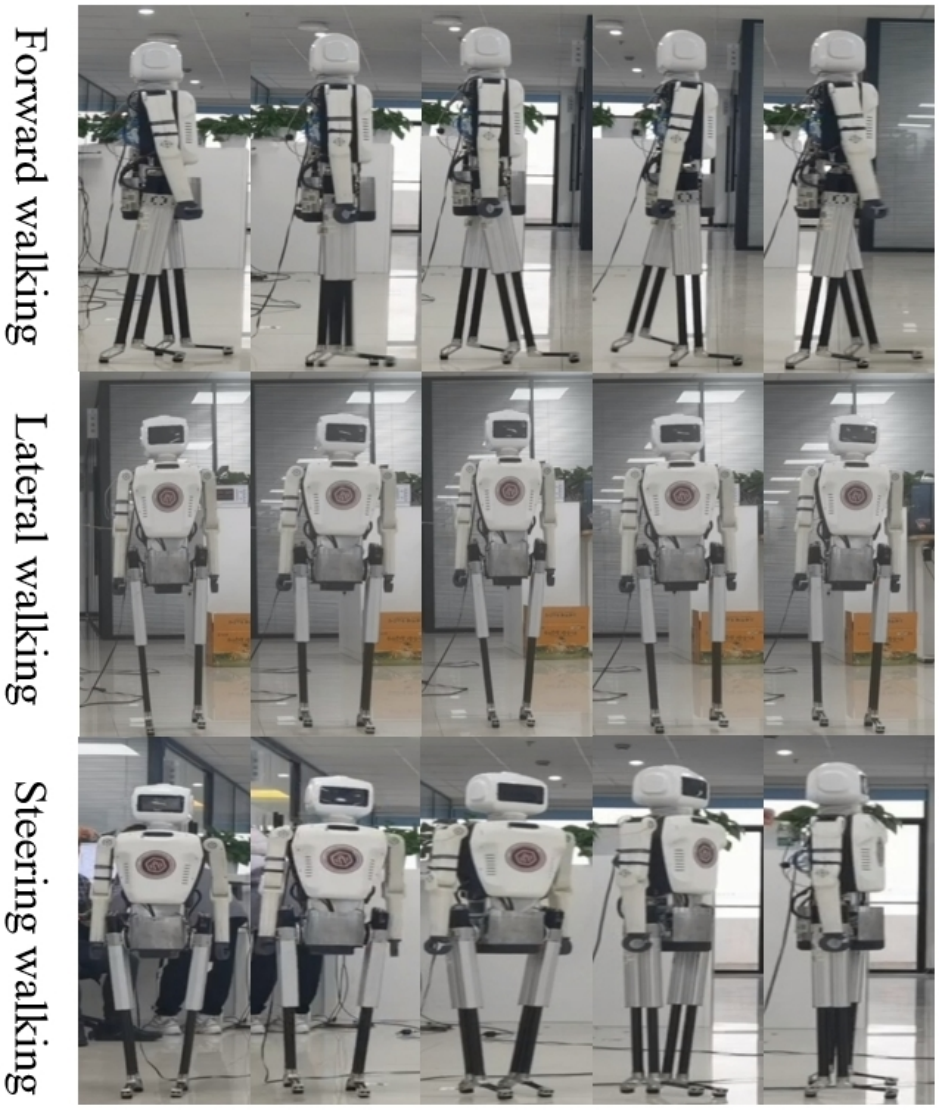


Figure 6. L04 omnidirectional walking diagram. Bipedal robot uses PFRL method to perform forward walking, lateral walking, and steering walking processes.

Table 1. The sim2real success rate of our method.

Walking Style	Forward Walking	Lateral Walking	Steering Walking
our method	100%	100%	90%
baseline	85%	90%	70%

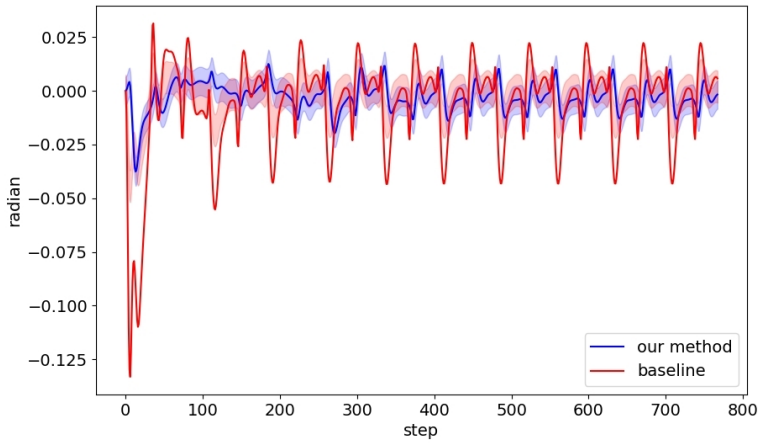


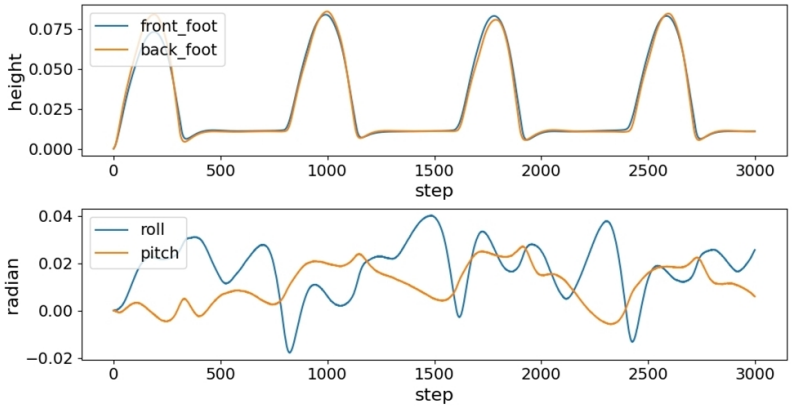
Figure 7. Euler angle variation diagram of forward walking.

We compare the PFRL method with the conventional direct joint position control approach. Figures 8b and 9b illustrate the posture variations and the off-ground height changes in the dual telescopic legs of the L04 robot during two walking modes. The figures demonstrate that the Euler angle fluctuations under the PFRL method are minimal, indicating high stability during complex omnidirectional walking. This not only helps reduce posture deviations but also ensures precise control of the robot across various walking modes, preventing accidental falls or operational errors caused by instability. The off-ground height of the parallel dual telescopic legs is derived from encoder-read joint positions using kinematic calculations. The dual telescopic leg heights of the left leg under the PFRL method are largely consistent. This ensures stable transitions of the virtual ankle joint and posture during walking while preventing unnecessary motion of the telescopic legs in the air. In contrast, the conventional method fails to maintain consistency in the dual telescopic leg heights, as illustrated in Figures 8a and 9a. This further results in posture instability of the L04.

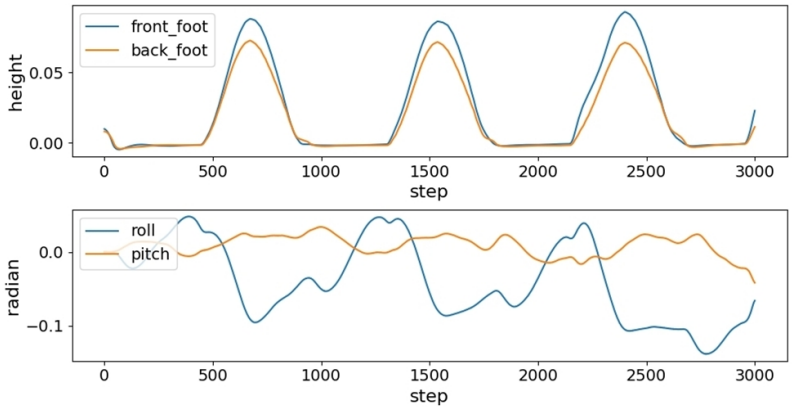
During the turning test, the conventional method failed to maintain stability. The aforementioned results indicate that the low-inertia structure of the L04 robot provides a distinct advantage in RL algorithm testing. The low-inertia design of the robot not only mitigates inertial effects during operation but also improves motion flexibility and stability. More importantly, the concentration of mass in the waist area of the L04 further reduces the sim2real gap, aligning the robot's real-world performance more closely with its simulated results. The PFRL method further demonstrated superior performance over the conventional method in the walking test.

The velocity command is set to [0,0,0] in order to facilitate a step-in-place motion. In the case of in situ stepping, our method is compared with the CAPS method. Both the CAPS method and our method employ the same reward function and feedforward. The position of the hip joints of the physical robot is recorded during in situ stepping, as illustrated in Figure 10. As illustrated in Figure 10, the proposed method yields smoother movements and circumvents the occurrence of oscillations in the policy output. In contrast, the CAPS method displays a tendency to oscillate when the hip joint is at 0 radians. This

phenomenon may be attributed to the noise generated by the IMU sensor upon contact with the ground.

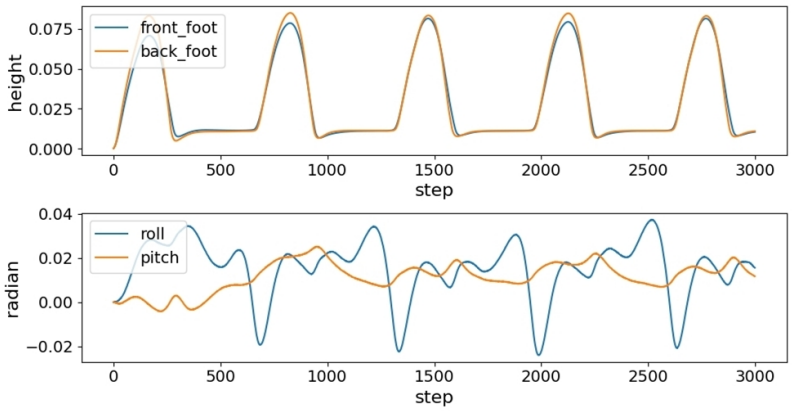


(**a**)



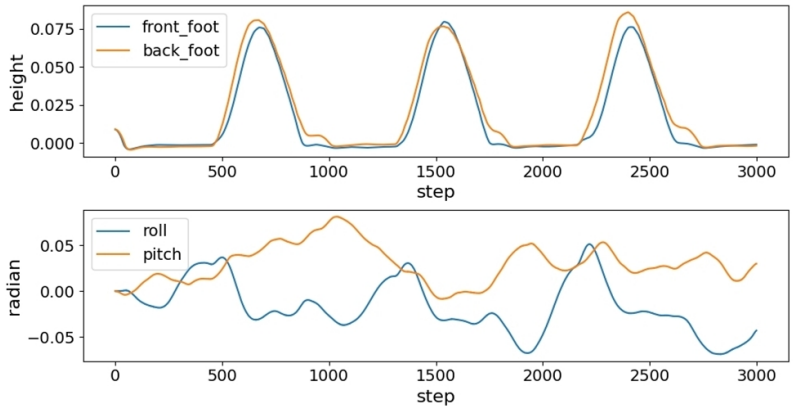
(**b**)

Figure 8. Forward walking. (**a**) Shows the change in height of the left leg dual telescopic pole off the ground and the change in body posture for the common method. (**b**) Shows the change in height of the left leg dual telescopic pole off the ground and the change in body posture for PFRL method.



(**a**)

Figure 9. *Cont*.



(**b**)

Figure 9. Lateral walking. (**a**) Shows the change in height of the left leg dual telescopic pole off the ground and the change in body posture for the common method. (**b**) Shows the change in height of the left leg dual telescopic pole off the ground and the change in body posture for PFRL method.

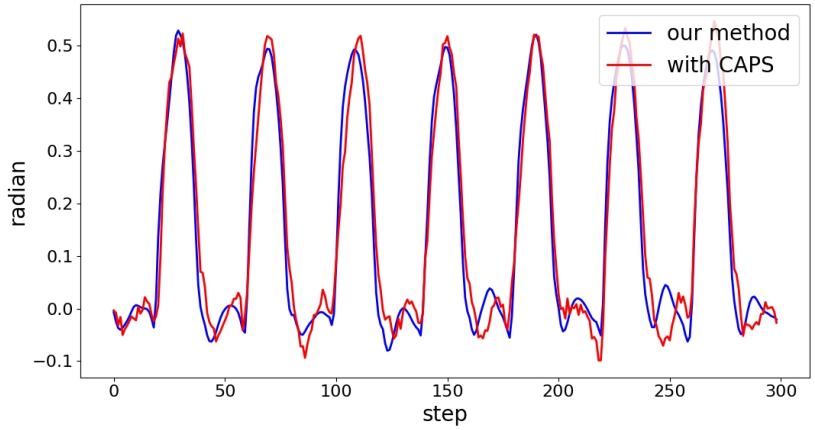


Figure 10. Joint position variation diagrams of two methods. The hip joint position changes in the robot controlled by our method and the CAPS method are compared while the robot is stepping in place. Our method keeps the robot's joint position smooth at all times. The CAPS method has some oscillations in the joint position.

5.2. Robustness Test

We perform external force disturbance experiments on the robot. The robot is struck by a water-filled bottle, as depicted in Figure 11. Despite the external impact, the robot maintains walking, with the roll variation in Euler angles illustrated in Figure 11. L04 is impacted at STEP 170 and regains stability at STEP 190. During the simulated training, no external interference is applied. Therefore, the external interference introduces a new condition to the robot. After the external disturbance, L04 takes 0.5s to regain stability. This proves that the PFRL method can cope with conditions not encountered in training.

5.3. Energy Consumption Comparison

We test the energy consumption of the L04 robot in forward walking. The speed commands are the three-speed commands in the omnidirectional walking test, and the robot walks for the 40s each time. The experiment uses the Cost of Transport (COT) to measure the energy efficiency of the robot's walking [21]. COT is a dimensionless quantity commonly used in the field of legged robots because it allows for easy comparisons of the energy efficiency of different robots or controllers. The COT for forward walking is 0.463, for sideways walking is 0.435, and for steering walking is 0.492. The ideal COT for

human walking is approximately 0.02 [22]; however, when factors such as joint friction and air resistance are considered, the COT rises to 0.05. In contrast, robots utilizing trajectory tracking control tend to exhibit COTs that are tens of times higher than that observed in human walking. For example, ASIMO has a COT of approximately 1.6, thereby demonstrating that our structure is capable of omnidirectional walking with minimal energy consumption. The COT of the baseline method is 0.533, 0.579, and 0.512 for the three-speed commands. This shows that the introduction of a priori knowledge of the cycle reduces the energy consumption of the robot's traveling.

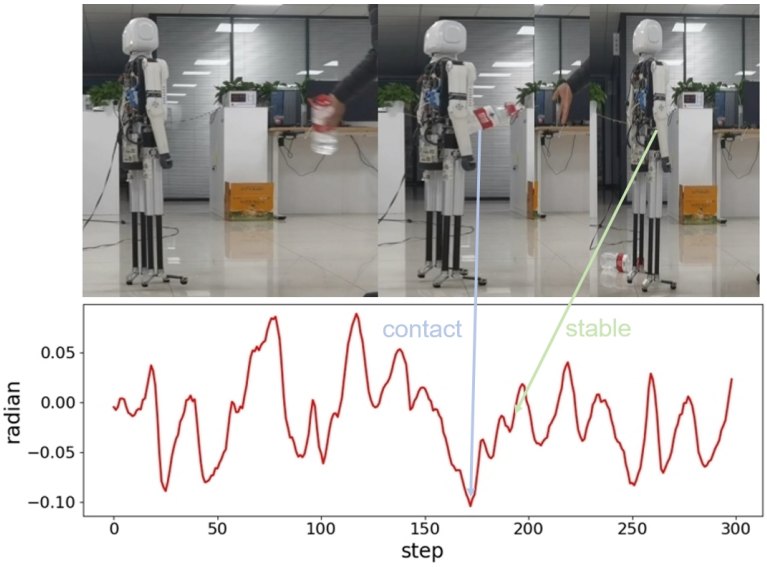


Figure 11. Diagram of the L04 robot's anti-interference performance. The bipedal robot remains stable when hit by a water bottle during walking. Below is a variation of the roll of the robot's COM. The robot only needs to adjust briefly after being smashed.

6. Conclusions

This paper proposes a novel control framework that integrates prior knowledge with RL, addressing key challenges in the motion control of parallel telescopic-legged bipedal robots by combining model-based prior knowledge with data-driven RL. The PFRL framework achieves highly robust sim2real transfer without relying on domain randomization or complex reward function design solely through the synergistic mechanism of feedforward motion compensation and policy network feedback regulation. Experimental results demonstrate that PFRL excels in omnidirectional walking tasks on flat ground, achieving a 100% success rate in forward and lateral walking and a 90% success rate in steering. Its dual-loop control architecture enables real-time dynamic balance adjustment, allowing the robot to quickly respond to external disturbances and maintain stability without adversarial training. Future work will focus on enabling walking on unstructured surfaces. We plan to introduce privileged information or teacher–student structures to recognize terrain information and body line velocity information. We will also conduct additional robustness tests using L04.

Author Contributions: Conceptualization, J.X. and J.H.; methodology, J.X.; software, H.M.; validation, Y.H.; formal analysis, Y.H.; investigation, J.X.; resources, Y.H.; data curation, Y.H.; writing—original draft preparation, J.X.; writing—review and editing, J.X. and Y.H.; visualization, J.H.; supervision, Y.H.; project administration, Y.H.; funding acquisition, Y.H. All authors have read and agreed to the published version of the manuscript.

Funding: This work was partially supported by the Natural Science Foundation of Shanghai under grant 24ZR1453100 and in part by the National Natural Science Foundation of China under grant 62403323.

Data Availability Statement: Data are contained within the article.

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

References

1. Wang, K.; Marsh, D.; Saputra, R.P.; Chappell, D.; Jiang, Z.; Raut, A.; Kon, B.; Kormushev, P. Design and Control of SLIDER: An Ultra-Lightweight, Knee-Less, Low-Cost Bipedal Walking Robot. In Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Las Vegas, NV, USA, 25–29 October 2020; pp. 3488–3495.
2. Mansor, Z.; Irawan, A.; Abas, M.F. Evolution, Design, and Future Trajectories on Bipedal Wheel-legged Robot: A Comprehensive Review. *Int. J. Robot. Control Syst.* **2023**, *3*, 673–703. [CrossRef]
3. Agarwal, S.; Popovic, M. Study of toe joints to enhance locomotion of humanoid robots. In Proceedings of the 2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids), Beijing, China, 6–9 November 2018; pp. 1039–1044.
4. Daneshmand, E.; Khadiv, M.; Grimminger, F.; Righetti, L. Variable horizon mpc with swing foot dynamics for bipedal walking control. *IEEE Robot. Autom. Lett.* **2021**, *6*, 2349–2356. [CrossRef]
5. Wang, K.; Fei, H.; Kormushev, P. Fast online optimization for terrain-blind bipedal robot walking with a decoupled actuated slip model. *Front. Robot. AI* **2022**, *9*, 812258. [CrossRef] [PubMed]
6. Qian, Y.; Yang, P.; Liu, W.; Sun, S.; Fu, M.; Song, W. Generative Design of XingT, A Human-sized Heavy-duty Bipedal Robot. In Proceedings of the 2022 IEEE International Conference on Robotics and Biomimetics (ROBIO), Jinghong, China, 5–9 December 2022; pp. 513–518.
7. Wang, K.; Xin, G.; Xin, S.; Mistry, M.; Vijayakumar, S.; Kormushev, P. A unified model with inertia shaping for highly dynamic jumps of legged robots. *Mechatronics* **2023**, *95*, 103040. [CrossRef]
8. Mou, H.; Tang, J.; Liu, J.; Xu, W.; Hou, Y.; Zhang, J. High Dynamic Bipedal Robot with Underactuated Telescopic Straight Legs. *Mathematics* **2024**, *12*, 600. [CrossRef]
9. Wang, H.; Luo, H.; Zhang, W.; Chen, H. CTS: Concurrent Teacher-Student Reinforcement Learning for Legged Locomotion. *IEEE Robot. Autom. Lett.* **2024**, *9*, 9191–9198. [CrossRef]
10. Yan, Y.; Mascaro, E.V.; Egle, T.; Lee, D. I-CTRL: Imitation to Control Humanoid Robots Through Bounded Residual Reinforcement Learning: A New Framework. *IEEE Robot. Autom. Mag.* 2025, *early access*.
11. Peng, X.B.; Ma, Z.; Abbeel, P.; Levine, S.; Kanazawa, A. Amp: Adversarial motion priors for stylized physics-based character control. *ACM Trans. Graph. (ToG)* **2021**, *40*, 1–20. [CrossRef]
12. Siekmann, J.; Godse, Y.; Fern, A.; Hurst, J. Sim-to-real learning of all common bipedal gaits via periodic reward composition. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Yokohama, Japan, 13–17 May 2024; pp. 7309–7315.
13. Li, Z.; Peng, X.B.; Abbeel, P.; Levine, S.; Berseth, G.; Sreenath, K. Robust and Versatile Bipedal Jumping Control through Reinforcement Learning. In Proceedings of the Robotics: Science and Systems XIX, Daegu, Republic of Korea, 10–14 July 2023.
14. Zhang, J.Z.; Yang, S.; Yang, G.; Bishop, A.L.; Gurumurthy, S.; Ramanan, D.; Manchester, Z. Slomo: A general system for legged robot motion imitation from casual videos. *IEEE Robot. Autom. Lett.* **2023**, *8*, 7154–7161. [CrossRef]
15. Rodriguez, D.; Behnke, S. DeepWalk: Omnidirectional bipedal gait by deep reinforcement learning. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 3033–3039.
16. Wu, Q.; Zhang, C.; Liu, Y. Custom sine waves are enough for imitation learning of bipedal gaits with different styles. In Proceedings of the 2022 IEEE International Conference on Mechatronics and Automation (ICMA), Guilin, China, 7–10 August 2022; pp. 499–505.
17. Ye, L.; Wang, X.; Liang, B. Realizing Human-like Walking and Running with Feedforward Enhanced Reinforcement Learning. In Proceedings of the International Conference on Intelligent Robotics and Applications, Hangzhou, China, 5–7 July 2023; pp. 439–451.
18. Li, Z.; Cheng, X.; Peng, X.B.; Abbeel, P.; Levine, S.; Berseth, G.; Sreenath, K. Reinforcement learning for robust parameterized locomotion control of bipedal robots. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 2811–2817.
19. Mysore, S.; Mabsout, B.; Mancuso, R.; Saenko, K. Regularizing action policies for smooth control with reinforcement learning. In Proceedings of the 2021 IEEE International Conference on Robotics and Automation (ICRA), Xi'an, China, 30 May–5 June 2021; pp. 1810–1816.
20. Gu, X.; Wang, Y.J.; Chen, J. Humanoid-Gym: Reinforcement Learning for Humanoid Robot with Zero-Shot Sim2Real Transfer. *arXiv* **2024**, arXiv:2404.05695.

21. Nishii, J. An analytical estimation of the energy cost for legged locomotion. *J. Theor. Biol.* **2006**, *238*, 636–645. [CrossRef] [PubMed]
22. Alexander, R.M. Walking made simple. *Science* **2005**, *308*, 58–59. [CrossRef] [PubMed]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.