CTBC: Contact-Triggered Blind Climbing for Wheeled Bipedal Robots with Instruction Learning and Reinforcement Learning

Rankun Li[*,1,2], Hao Wang[*,2], Qi Li[*,2], Zhuo Han[1,2], Yifei Chu[2], Linqi Ye[†,1], Wende Xie[†,2], and Wenlong Liao[2]

***Abstract*— In recent years, wheeled bipedal robots have gained increasing attention due to their advantages in mobility, such as high-speed locomotion on flat terrain. However, their performance on complex environments (e.g., staircases) remains inferior to that of traditional legged robots. To overcome this limitation, we propose a general contact-triggered blind climbing (CTBC) framework for wheeled bipedal robots. Upon detecting wheel-obstacle contact, the robot triggers a leg-lifting motion to overcome the obstacle. By leveraging a strongly-guided feedforward trajectory, our method enables the robot to rapidly acquire agile leg-lifting skills, significantly enhancing its capability to traverse unstructured terrains. The approach has been experimentally validated and successfully deployed on LimX Dynamics' wheeled bipedal robot, Tron1. Real-world tests demonstrate that Tron1 can reliably climb obstacles well beyond its wheel radius using only proprioceptive feedback. Project page: `https://ctbc-for-wheeled-bipedal-robots.github.io/`**

I. INTRODUCTION

Traditional legged robots have demonstrated remarkable agility and adaptability on complex terrain, yet their locomotion efficiency and speed remain comparatively low [1], making it difficult to satisfy the demands of rapid mobility. Wheeled-legged robots, benefiting from their high energy efficiency over long distances and superior travel speed [2], have been extensively studied and deployed across various domains. Nevertheless, when confronted with challenging environments such as staircases or uneven surfaces, the inherent limitations of wheeled robots become evident, as they lack the flexibility required to surmount obstacles effectively.

For wheeled-legged robots, tire dimensions exert a decisive influence on the feasibility of stair-climbing. Larger wheels confer a clear advantage; for instance, Simon Chamorro et al. [3] demonstrated that the Ascento robot can ascend 15 cm stairs with wheels of 25 cm radius. In contrast, the wheels on our Tron1 wheeled-biped robot have a radius of only 12.7 cm, substantially increasing the difficulty of stair traversal—especially when tackling taller steps.

To overcome the pain points of wheeled-legged robots in complex terrain, we draw inspiration from the contact-triggered reflexes observed in human gait and propose a contact-triggered control framework that synergizes feedforward instruction learning with reinforcement learning.

Fig. 1. We have developed a contact-triggered blind climbing control policy that works for wheeled-legged robots of various tire sizes, enabling them to conquer a variety of challenging terrains.

By leveraging privileged information within an asymmetric actor–critic architecture, we develop a biomimetic gait reflex that enables wheeled-biped robots to ascend stairs with ease. The key contributions of this work are summarized as follows:

1) **Contact-Triggered RL Task Formula.** A contact-triggered reinforcement learning task formula is proposed, which is applicable to wheeled-legged robots of various tire sizes, including the small-tired robot Tron1, to achieve stair-climbing strategies.
2) **Feedforward Trajectory Instruction Learning.** By combining instruction learning with reinforcement learning, the feedforward trajectory is used to teach the robot when to lift its legs appropriately. This approach avoids unnecessary exploration and significantly improves learning efficiency.
3) **Efficient Traversal of Complex Terrain.** This method enables the Tron1 robot to continuously climb 20cm stairs(Fig. 1), which is far beyond the tire radius. The strategy supports both fast movement on flat ground and motion on complex terrain, enhancing the practical adaptability of the robot.

II. RELATED WORK

A. RL-Based Legged Locomotion

In the realm of legged locomotion, model-based optimization techniques such as Model Predictive Control (MPC) and

* Indicates Equal Contribution.
† Indicates Corresponding Author.
[1] The School of Future Technology, Shanghai University, 200444 Shanghai, China. rankunli@shu.edu.cn, yelinqi@shu.edu.cn
[2] COWAROBOT, China.

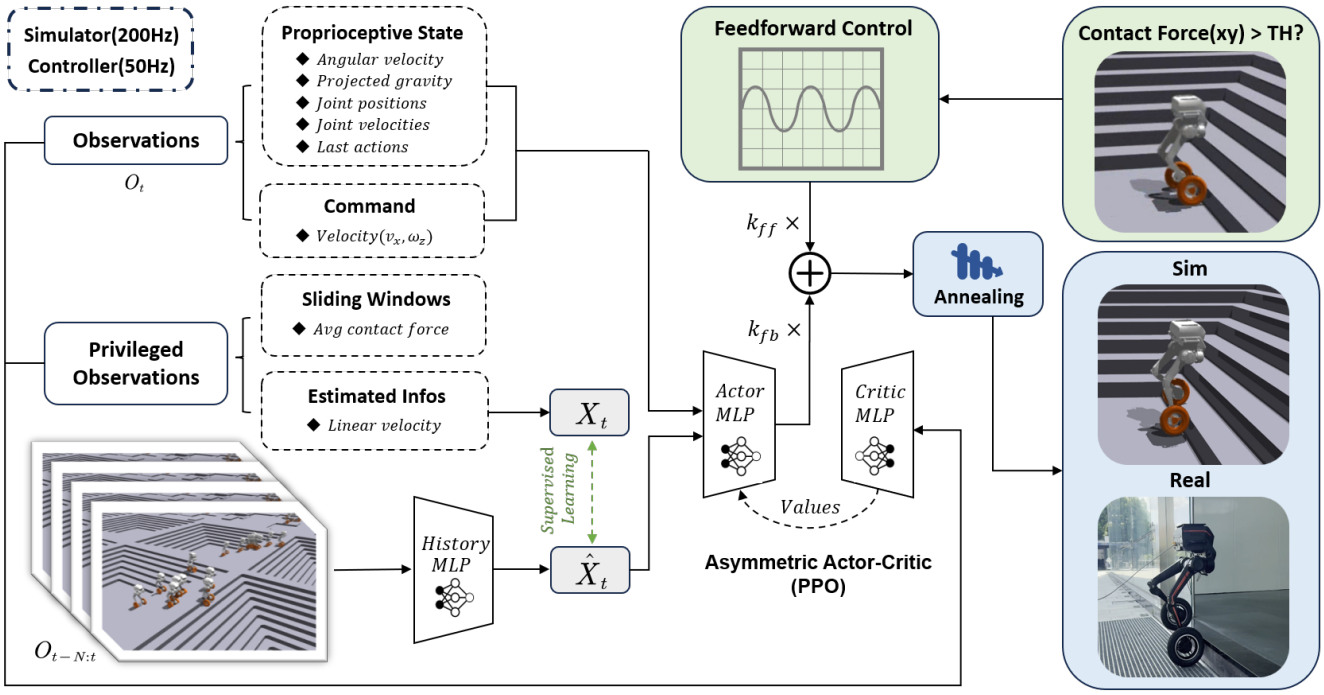

Fig. 2. Overview of our universal contact-triggered blind stair-climbing framework. The overall framework is mainly composed of a state estimator and an asymmetric actor-critic network. For elastic tires, we introduce a contact force sliding window to simulate more realistic contact. When the contact force in the xy direction of the foot exceeds a threshold, it triggers the designed feedforward reference trajectory to guide the robot to lift its leg. After annealing the feedforward trajectory, the method can be zero-shot transferred to the physical robot.

trajectory optimization have long been widely adopted [4]. Nevertheless, the reliance of these methods on accurate and intricate dynamic models often constrains their robustness and generalization. Recently, model-free reinforcement learning has risen to prominence, offering an end-to-end learning paradigm that simultaneously strengthens robustness and markedly enhances generalization across diverse motion-control tasks, positioning itself as a powerful alternative to traditional approaches.

In 2019, Hwangbo et al. [5] introduced reinforcement learning to legged robot control by proposing a policy conditioned on desired velocity that outputs joint-position targets, augmented with an actuator network to enhance motor modeling accuracy. Building directly on this seminal framework, Lee et al. [6] developed a teacher–student architecture that enables robust traversal of hills, steps, and other challenging terrains using only on-board proprioception. Siekmann et al. [7] further demonstrated that an RL policy trained solely on proprioception can drive the bipedal robot Cassie to blindly ascend real-world stairs. Leveraging massively GPU-parallel simulation in Isaac Gym, Rudin et al. [8] trained a locomotion policy with Proximal Policy Optimization (PPO) in just 20 minutes that transfers zero-shot to real hardware. Extending this promising work, Rudin et al. [9] introduced a task formulation based on positional goals: within a strict time limit, the robot must reach a target location, autonomously planning both path and motion to overcome obstacles and complete navigation without additional motion priors.

B. Wheeled-Legged Locomotion on Rough Terrain

In complex-terrain locomotion, conventional model-based methods often hinge on either simple heuristics that dictate when to walk or when to drive [10], or on fixed, pre-defined gait sequences [11]. Most policies for legged robots still embed hand-engineered gait patterns [12], [13] or biologically inspired motion primitives [6], [14].

Bjelonic et al. [15] introduced an online gait generator driven by leg availability: when the availability score of any leg drops below a threshold, the leg is automatically switched to swing while the others continue to drive or support; a single MPC parameter set suffices for all gaits, eliminating manual cost-weight tuning. Klemm et al. [16] leveraged non-smooth trajectory optimization to co-solve global motion planning and contact switching for stairs, steps and jumps in one pass, creating a closed perception–control loop and demonstrating continuous stair climbing on the Ascento wheeled-leg platform. Lee et al. [17] trained a quadrupedal wheeled robot with RL to switch on-the-fly between high-speed wheel driving and legged obstacle clearance in response to commands and terrain, enabling robust obstacle traversal. Lee et al. [18] further proposed a fully-integrated end-to-end framework that fuses model-free RL, privileged learning and hierarchical control, allowing seamless transitions between walking and driving for tasks such as table jumping and stair climbing. Chamorro et al. [3] demonstrated that a blind RL policy, operating without vision or localization, enables Ascento to reliably climb 15 cm stairs by relying solely on a positional objective, binary terrain

flags, and an asymmetric actor-critic architecture.

Although prior work has made significant advances in the locomotion of wheeled-legged robots over complex terrain, to the best of our knowledge, no universal obstacle-traversal framework yet exists for bipedal wheeled robots with arbitrary wheel sizes. In particular, when the robot is required to surmount high steps without any additional exteroceptive sensing, existing methods often struggle and are rarely effective.

III. METHODOLOGY

As illustrated in Fig. 2, our universal contact-triggered blind climbing framework is depicted. The following sections systematically detail the training environment, reinforcement-learning task formulation, training pipeline, design of the contact-triggered mechanism, integration of feedforward instruction learning, and the sim-to-real transfer strategy with concrete deployment specifics.

A. Learning Environment

1) Simulator: We select Isaac Gym [19] as our training platform because it is specifically designed for reinforcement-learning applications and is equipped with a GPU-accelerated architecture that dramatically increases agent-training speed thanks to its high degree of parallelism. In addition, Isaac Gym supports domain randomization techniques [20], which improve the robustness of reinforcement-learning agents by introducing environmental variations during training and thus facilitate the transfer of policies to the real world. To validate our sim-to-real pipeline, we also leverage MuJoCo [21] for cross-validation of the trained policies [22]. MuJoCo is renowned for its high-fidelity simulation and is widely used to verify reinforcement-learning policies; a policy that successfully deploys in MuJoCo is generally expected to transfer seamlessly to the physical environment.

2) Learning Algorithm: We adopt PPO with an asymmetric actor–critic architecture [23]. This variant of the standard actor–critic framework employs separate networks for the actor and the critic, permitting independent updates [24], [25]. Building upon this, we further introduce a Multi-Layer Perceptron (MLP) encoder for state estimation [26]. All training and experiments are conducted on a single NVIDIA GeForce RTX 4090 GPU with 24 GB of VRAM.

3) Terrains: Our environment is structured as an 8 m × 8 m terrain divided into 10 columns: one column of smooth slope, one of rough slope, six columns of stairs, and two columns of discrete obstacles, as shown in Fig. 3. To progressively increase the curriculum difficulty, the terrain is further split into 10 rows. Generally, the greater the diversity of obstacles encountered during training, the more robust the resulting policy becomes.

B. Task Formulation

1) State: We adopt an asymmetric actor–critic architecture, so we partition the state into two parts: (i) Observations, which are accessible to both the Actor and the Critic, and (ii) Privileged Information, which is revealed to the Critic only during training. As summarized in Table I, we explicitly list the observations available to the actor during both training and deployment, as well as the privileged information reserved for the critic at training time. In particular, the term last actions denotes a composite action vector obtained by a weighted fusion of the raw actions directly output by the network and the actions derived from feedforward trajectory.

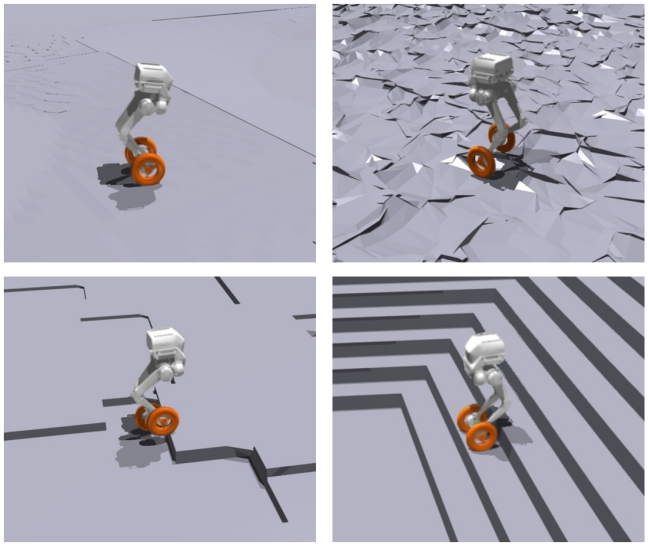

Fig. 3. Terrain type. From left to right and top to bottom: smooth slope, rough slope, discrete obstacles, and stairs.

TABLE I
OBSERVATION & PRIVILEGED INFORMATION

Symbol	Description	Units	Coeff.	Size	Noise (%)
Observation (Actor / Critic)					
$\vec{\theta}$	Angular velocity	rad/s	1.0	3	±2
$\vec{\gamma}$	Projected gravity	–	1.0	3	±5
$\vec{q}$	Joint positions	rad	1.0	6	±1
$\vec{\dot{q}}$	Joint velocities	rad/s	0.05	8	±50
$a_{\vec{last}}$	Last actions	rad & rad/s	1.0	8	0
Privileged Information (Critic only)					
v_x	Linear velocity	m/s	2.0	3	–
μ_{contact}	Avg contact forces	N	1.0	6	–

2) Actions: In this study, the robot's action space has a dimension of 8. These actions correspond to the robot's individual joints, including both leg and wheel joints. For the leg joints, the action commands are directly used as target positions for low-level proportional–derivative (PD) controllers, i.e., the controllers strive to drive the joints to these preset positions. For the wheel joints, the action vector represents target angular velocities; in other words, the joint motors operate in velocity-control mode, aiming to reach the specified angular velocities.

3) Rewards: Our task-specific rewards are summarized in Table II. The reward function is composed of three main components:

1) **Task rewards**: including velocity-tracking and foot-

TABLE II

REWARD TERMS AND CLASSIFICATIONS

Reward	Formula	Coeff.
Task Rewards		
Lin. vel tracking x	$\exp\big(-20\,(v_{\text{cmd},x}-v_{\text{base},x})^2\big)$	1.2
Lin. vel tracking y	$\exp\big(-20\,(v_{\text{cmd},y}-v_{\text{base},y})^2\big)$	1.0
Lin. vel tracking x pb	$\frac{\Delta\phi_x}{\Delta t}$	1.0
Lin. vel tracking y pb	$\frac{\Delta\phi_y}{\Delta t}$	0.8
Ang. vel tracking	$\exp\big(-20\,\lvert\omega_{\text{cmd}}-\omega_{\text{base}}\rvert\big)$	1.0
Ang. vel tracking pb	$\frac{\Delta\phi_\omega}{\Delta t}$	0.5
Tracking target pos	$\exp\big(-2\lVert q-q_{\text{target}}\rVert\big)-0.2\lVert q-q_{\text{target}}\rVert$	0.8
Feet air time	$\sum_i \min(t_{\text{air},i},0.5)\,\mathbb{I}_{\text{first contact},i}$	2.0
Feet contact number	$\sum_i\big[\mathbb{I}_{\text{contact}_i=\text{stance}_i}-1.3\,\mathbb{I}_{\text{contact}_i\neq\text{stance}_i}\big]$	2.0
Feet clearance	$\sum_i \mathbf{1}_{\text{swing},i}\cdot\mathbf{1}_{h_{\min}<h_i<h_{\max}}$	2.0
Style Rewards		
Nominal foot position	$\frac{1}{N}\sum_i\exp\Big[-\big(\frac{(z_i-z_{\text{nom}})^2}{\sigma_z^2}+\frac{\lVert\mathbf{v}_{\text{cmd}}\rVert^2}{\sigma_v^2}\big)\Big]$	1.0
Default pose	$\sum_j\lvert q_j-q_{j,\text{default}}\rvert$	-1.0
Feet distance	$\max(0,d_{\min}-d)+\max(0,d-d_{\max})$	-10.0
Wheel zero velocity	$\exp\big(-\sum_{j\in\{3,7\}}\mathbf{1}_{\text{swing},j}\,\dot\theta_j^2\big)$	0.5
Same foot x position	$\lvert x_0-x_1\rvert$	-2.0
Base height	$\lvert h_{\text{base}}-h_{\text{target}}\rvert$	-20.0
Orientation	$\bar g_x^2+\bar g_y^2$	-12.0
Regularization Rewards		
Wheel spin	$\sum_j\max(0,0.8\,\lvert r\dot\theta_j\rvert-\lVert\mathbf{v}_{\text{foot},j}\rVert-0.1)$	-5.0
Opposite base vel	$\max\big(0,-\text{sgn}(v_{\text{cmd}})\,v_x\big)$	-40.0
Opposite wheel vel	$\sum_{j\in\{L,R\}}\max\big(0,-\text{sgn}(v_{\text{cmd}})\,\dot\theta_j\big)$	-2.0
Lin vel z	v_z^2	-0.3
Ang vel xy	$\omega_x^2+\omega_y^2$	-0.01
Torques	$\sum_j\tau_j^2$	-1×10^{-5}
Dof acc	$\sum_j\ddot q_j^2$	-2.5×10^{-7}
Dof vel	$\sum_j\dot q_j^2$	-1×10^{-5}
Action rate	$\sum_j(a_j-a_j^{\text{prev}})^2$	-0.01
Action smooth	$\sum_j(a_j-2a_j^{\text{prev}}+a_j^{\text{prev2}})^2$	-0.005
Collision	$\sum_{i\in\mathscr{F}_{\text{penalised}}}\mathbf{1}_{\lVert\mathbf{F}_i\rVert>1}$	-50.0
Feet contact forces	$\max\big(0,\bar F_z-F_{\max}\big)$	-5.0
Dof pos limits	$-\sum_j\max\big(0,\lvert q_j-q_j^{\text{limit}}\rvert\big)$	-2.0

lifting terms, which ensure the robot moves at the desired speed and follows the prescribed gait pattern.

2) **Style rewards**: comprising foot-pose and body-pose terms, which encourage the robot to maintain a natural and stable gait.
3) **Regularization rewards**: used to optimize motion smoothness and prevent superfluous joint movements.

C. Contact-Triggered Mechanism

The methodology is inspired primarily by [17], with the core objective of updating the robot's state and determining whether each foot should be in the stance or swing phase based on the contact forces measured at the feet in the horizontal (xy) plane. We extend this framework with three key enhancements:

1) **Threshold-based Trigger:** Once the contact force on either wheel exceeds the preset threshold, the feedforward trajectory is instantly triggered, causing that leg to lift first; the contralateral leg then synchronously follows, producing a coordinated alternating ascent.

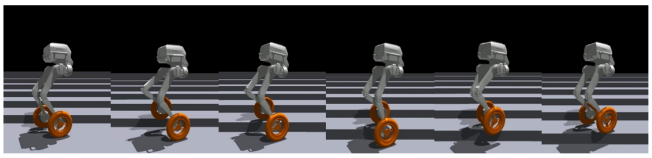

Fig. 4. When either wheel makes contact, only the contacting leg lifts.

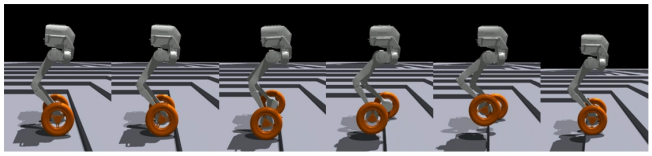

Fig. 5. If both wheels are in contact, the leg with stable contact or the larger contact force is chosen to lift.

2) **Sliding-Window Filter:** Because Isaac Gym is a rigid-body simulator, wheel–ground contacts can flicker. Following [3], we apply a three-frame sliding window to aggregate historical contact states, suppressing noise while preserving genuine triggers.
3) **Wheel-Leg Integration:** The trigger mechanism is coupled with the rolling wheel model, enabling seamless transitions between rolling and stepping for energy-efficient, high-performance obstacle traversal.

The triggering mechanism determines the lifting sequence by continuously monitoring the contact forces on both feet in real time. For each foot, the system stores the latest three frames of force data and designates the contact as stable contact only if all three frames exceed the threshold.

1) If contact is detected with only one foot, lift solely that foot (Fig. 4).
2) When both feet are in contact (Fig. 5):
   i) If one foot has stable contact while the other does not, the foot with stable contact is lifted first.
   ii) If both feet exhibit stable contact, the foot with the larger contact force is lifted first.

This mechanism ensures the robot can flexibly adapt its locomotion strategy to actual contact conditions, enabling more natural and stable motion over complex terrain.

D. Feedforward Instruction Learning

The idea of feedforward instruction learning originates primarily from [27]. This approach leverages a baseline gait motion as a feedforward signal, providing the robot with a clear starting point for locomotion. By integrating the reward mechanism of reinforcement learning, the robot can rapidly learn and master diverse locomotion gaits. This method significantly reduces the exploration required when starting from a random policy, thereby optimizing the robot's learning process for complex action sequences. We adapt this idea to wheeled-legged robots by injecting feedforward trajectories solely into the hip-pitch and knee-pitch joints, with the knee trajectory amplitude set to twice that of the hip. The composite desired joint position is computed as

TABLE III
DOMAIN RANDOMIZATION

Parameter	Range	Unit
Payload mass	$[-0.5,\ 2]$	kg
Center of mass shift	$[-3,\ 3]\times[-2,\ 2]\times[-3,\ 3]$	cm
Kp Factor	$[0.8,\ 1.2]$	N/rad
Ka Factor	$[0.8,\ 1.2]$	N·s/rad
Friction	$[0.2,\ 1.6]$	–
Restitution	$[0.0,\ 1.0]$	–
Inertia	$[0.8,\ 1.2]$	–
Motor torque	$[0.8,\ 1.2]$	N
IMU offset	$[-1.2,\ 1.2]$	–
Default dof pos	$[-0.05,\ 0.05]$	N
Step delay	$[0,\ 20]$	ms
Push interval	8	s
Push vel (xy)	1.0	m/s

follows:

$$a = k_{\text{fb}} a_{\text{policy}} + k_{\text{ff}} a_{\text{feedforward}}, \tag{1}$$

$$a_{\text{feedforward}}(t) = 0.5\left(1 - \cos\left(\tfrac{2\pi}{T} t\right)\right), \quad T = 0.6 \text{ s}. \tag{2}$$

where a comprises the desired joint actions for robots fed to the PD controller, and the positive scalars k_{fb}, k_{ff} are tunable weights balancing the policy output a_{policy} and the feedforward trajectory $a_{\text{feedforward}}$.

E. Sim-to-real Transfer

1) Domain Randomization: To achieve zero-shot sim-to-real transfer, we introduce a broad set of randomization factors in simulation to model real-world uncertainties and enhance the policy's generalization ability, as detailed in Table III.

2) Annealing: During simulation training, the torques are computed from a composite action. For sim-to-real deployment, however, the robot relies solely on the neural-network policy. We therefore anneal the feedforward component: as training progresses and the policy converges, the coefficient k_{ff} is gradually decreased. This annealing continues until k_{ff} reaches zero, at which point the robot's motion is completely controlled by the network policy.

IV. EXPERIMENTS

A. Simulation Experiments

To quantify the contribution of each key component in the proposed CTBC method, we conducted four controlled ablation experiments under an identical simulation environment and training setup. All policies started from the same random seed and were evaluated after 80000 iterations to ensure fairness. The compared methods are:

- **CTBC** (ours): employs both (1) contact-triggered leg-lifting mechanism and (2) feedforward trajectory.
- **CTBC w/o feedforward**: retains the contact-triggered mechanism but removes the feedforward trajectory.
- **CTBC w/o contact-trigger**: retains the feedforward trajectory but removes the contact-triggered mechanism.
- **CTBC w/o both**: removes both components and serves as the baseline.

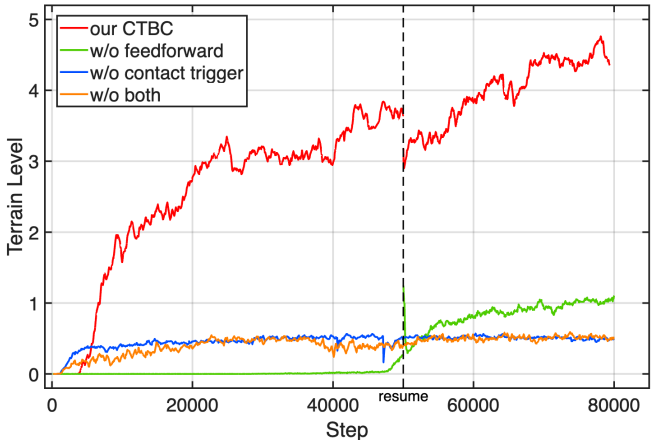

Fig. 6. Terrain-level versus training iterations for all ablation variants. Step height increases from 8 cm to 20 cm while width decreases from 50 cm to 28 cm.

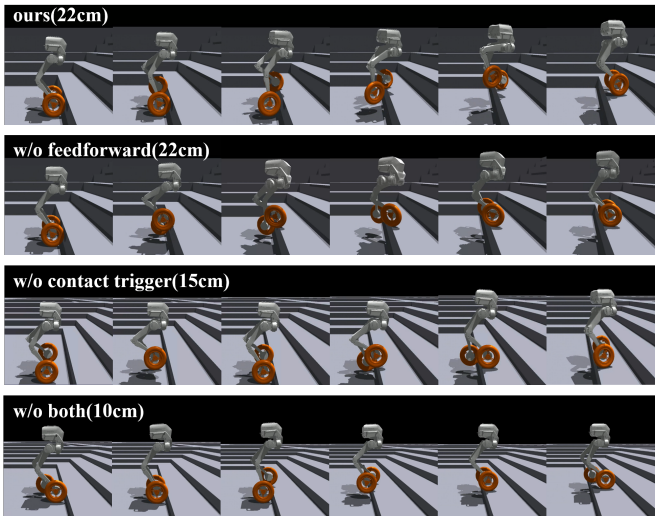

Fig. 7. Ablation experiments in simulation

For the training of both CTBC and CTBC w/o feedforward, we observed that a fore–aft leg motion makes it easier for the robot to learn the lifting action. Consequently, we adopted a two-stage training: in stage one, no constraints are placed on the lateral foot positions; in stage two, we add a "*same foot x position*" reward on top of the policy from stage one to correct the fore–aft leg motion.

We adopt terrain level as the unified metric to measure the robot's ability to ascend progressively more challenging stairs. Difficulty is increased by simultaneously adjusting two geometric parameters: the step height rises linearly from 8 cm to 20 cm, while the step width decreases linearly from 50 cm to 28 cm, both scaling with terrain level. To validate our method, we employ a 10 cm feedforward leg-lift trajectory while using a reward function to constrain the lift height between 10 cm and 20 cm during simulation training. The results are presented in Fig. 6.

The ablation results in Fig. 7 show that, without the feedforward trajectory (CTBC w/o feedforward), the robot's

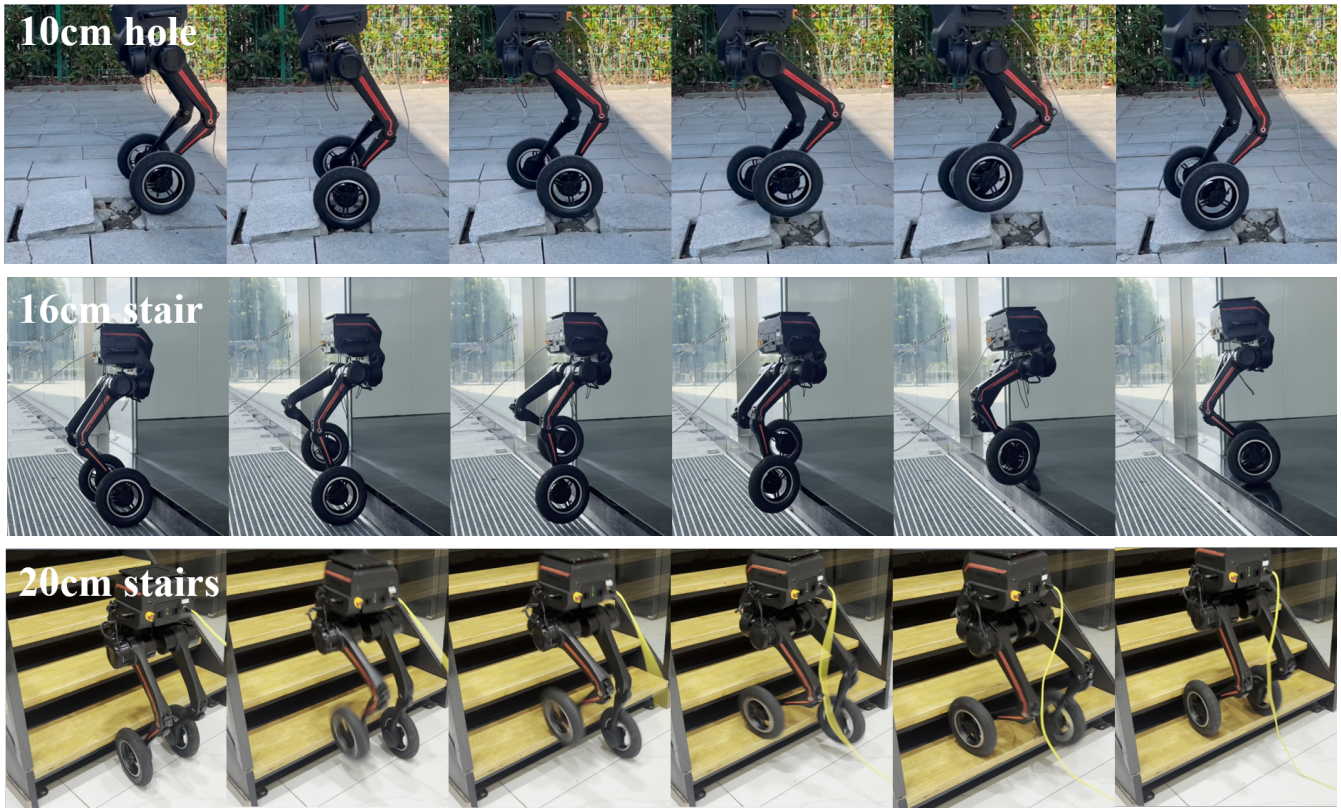

Fig. 8. Hole-escape and stair-climbing experiments

TABLE IV
SUCCESS RATE (%) ON STAIRS OF INCREASING HEIGHT

Ablation Experiments	Step height (cm)						
	8	10	12	15	18	20	22
CTBC (our method)	**100**	**100**	**100**	**98**	**96**	**86**	**70**
CTBC w/o feedforward	96	96	96	92	80	58	38
CTBC w/o contact-trigger	62	60	56	46	18	2	0
CTBC w/o both	46	34	28	8	4	0	0

leg-lift height when attempting a 22 cm step is markedly reduced. Removing the contact-triggered mechanism (CTBC w/o contact-triggered) causes rapid, inefficient stepping motions that waste energy. When both components are ablated (CTBC w/o both), the robot is unable to acquire a viable stair-ascending gait and is limited to clearing 10 cm steps with markedly low success. This further underscores the necessity of combining feedforward trajectories and contact-triggered mechanism for energy-efficient, high-success obstacle traversal.

To intuitively compare how each method affects the robot's obstacle-crossing capability, we evaluated 100 robots under identical domain-randomization settings: stair width was fixed at 40 cm while height was progressively increased; success rates are reported in Table IV. The results highlight two key findings:

1) Contact-triggered mechanism functions as a gating controller for leg lift initiation. Its removal causes success rates to fall sharply reaching zero under feedforward-only control.
2) Feedforward trajectories enhance lift height and dynamic stability. When combined with contact triggering they significantly shorten learning time and raise terrain-level scores improving 20 cm step success from 58% to 86% and maintaining 70% at 22 cm.

B. Real-World Experiments

We deployed the CTBC policy on the 8-DoF wheeled-legged robot *LimX Dynamics Tron1* without any exteroceptive sensing. To test its robustness and transferability, we designed two extreme scenarios: hole escape and stair climbing, as shown in Fig. 8

Hole escape: When one wheel drops into a 10 cm-deep hole, the contact force on the corresponding leg first exceeds the trigger threshold. The policy immediately executes a leg-lifting motion, allowing the robot to free itself smoothly.

Stair climbing: Confronted with 16 cm and 20 cm-high steps, the policy chooses which leg to lift first based on the relative contact forces when both wheels nearly touch the step, enabling rapid ascent while maintaining balance.

As Fig. 9 shows, using our CTBC method the biped-wheeled robot can continuously ascend 20 cm open-gap stairs, an extremely challenging task. When the contact force on the left wheel reaches the threshold, the left leg is triggered to execute a motion close to the feedforward

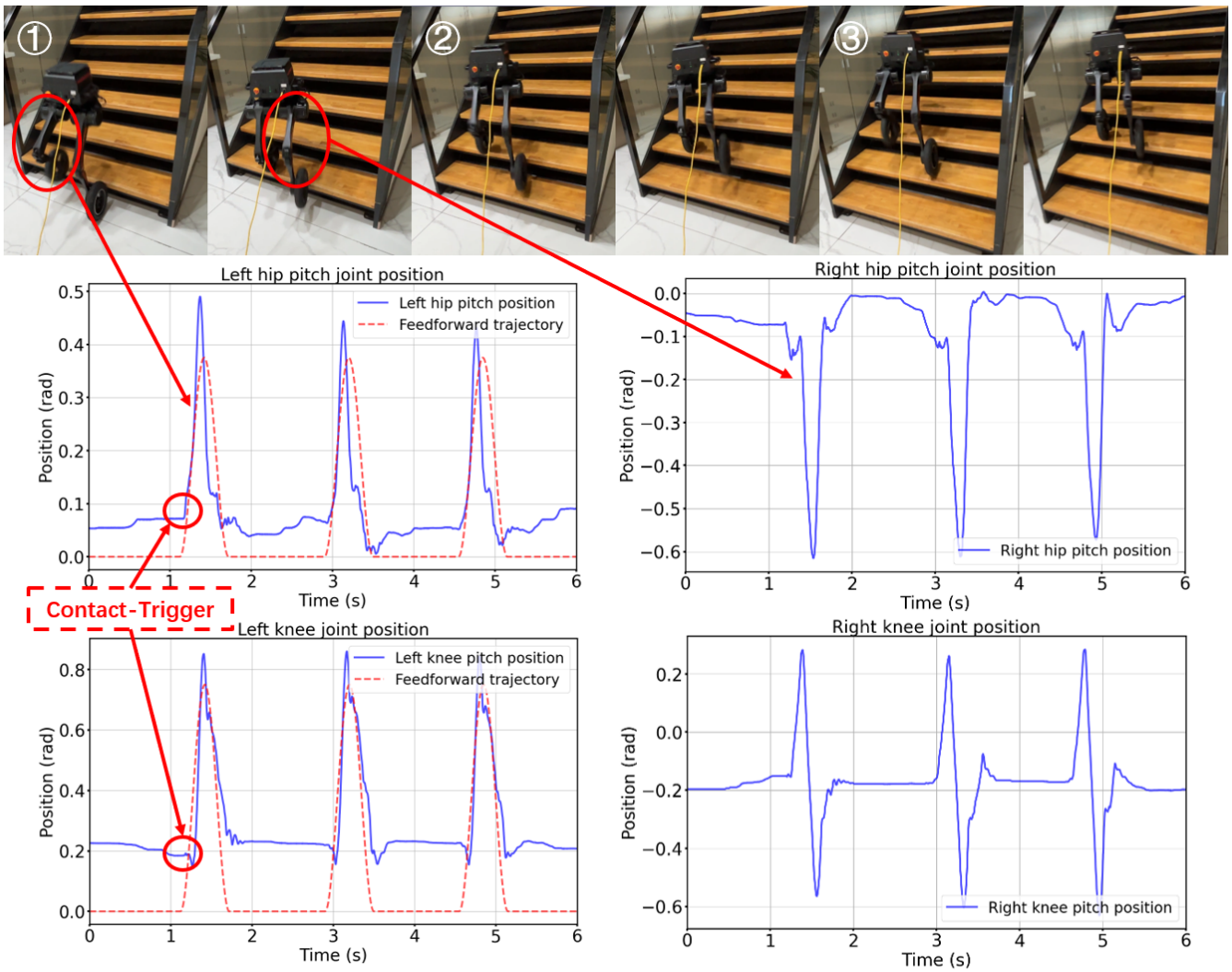

Fig. 9. Climb continuously up three open-gap stairs, each 20 cm high.

trajectory while reinforcement learning adaptively adjusts the leg's path. This confirms that the feedforward trajectory provides effective guidance.

Remarkably, even when the annealing stage is skipped and the feedforward trajectory is abruptly removed, the robot can still ascend 20 cm steps stably. This demonstrates that the network has internalized the leg-lifting policy and exhibits exceptional generalization.

It is worth noting that the deployed policy uses only a 10 cm-lift feedforward trajectory to surmount obstacles up to 20 cm. To overcome higher obstacles, one only needs to increase the feedforward trajectory height and expand the leg-lift range in the reward function, then re-train the policy.

V. CONCLUSIONS

In conclusion, we propose a contact-triggered, blind stair-climbing method for biped-wheeled robots that blends feed-forward instruction learning with reinforcement learning, enabling the robot to conquer stairs and holes without any external perception. Both simulation and hardware experiments demonstrate robust traversal over stairs and holes, confirming the framework's terrain-crossing capability. However, the extended front–rear gait in the first phase limits the rear leg's contact opportunities, causing the second-phase policy to consistently lift the front leg first and become locked into this early preference. Moreover, the current strategy remains purely blind and has yet to incorporate external perception, navigation, or trajectory planning. Future work will therefore focus on eliminating this early bias and adopting the blind policy as a low-level controller, augmented by a lightweight vision/LiDAR high-level policy. This hierarchical "blind–perceptive" architecture will enable real-time terrain estimation, global navigation, and efficient traversal of stairs and holes in unknown environments.

REFERENCES

[1] P. Biswal and P. K. Mohanty, "Development of quadruped walking robots: A review," *Ain Shams Engineering Journal*, vol. 12, no. 2, pp. 2017–2031, 2021.

[2] M. Bjelonic, V. Klemm, J. Lee, and M. Hutter, "A survey of wheeled-legged robots," in *Climbing and walking robots conference*. Springer, 2022, pp. 83–94.

[3] S. Chamorro, V. Klemm, M. d. L. I. Valls, C. Pal, and R. Siegwart, "Reinforcement learning for blind stair climbing with legged and wheeled-legged robots," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2024, pp. 8081–8087.

[4] P. M. Wensing, M. Posa, Y. Hu, A. Escande, N. Mansard, and A. Del Prete, "Optimization-based control for dynamic legged robots," *IEEE Transactions on Robotics*, vol. 40, pp. 43–63, 2023.

[5] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, no. 26, p. eaau5872, 2019.

[6] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, vol. 5, no. 47, p. eabc5986, 2020.

[7] J. Siekmann, K. Green, J. Warila, A. Fern, and J. Hurst, "Blind bipedal stair traversal via sim-to-real reinforcement learning," *arXiv preprint arXiv:2105.08328*, 2021.

[8] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Conference on robot learning*. PMLR, 2022, pp. 91–100.

[9] N. Rudin, D. Hoeller, M. Bjelonic, and M. Hutter, "Advanced skills by learning locomotion and local navigation end-to-end," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 2497–2503.

[10] M. Bjelonic, P. K. Sankar, C. D. Bellicoso, H. Vallery, and M. Hutter, "Rolling in the deep–hybrid locomotion for wheeled-legged robots using online trajectory optimization," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3626–3633, 2020.

[11] M. Hosseini, D. Rodriguez, and S. Behnke, "State estimation for hybrid locomotion of driving-stepping quadrupeds," in *2022 Sixth IEEE International Conference on Robotic Computing (IRC)*. IEEE, 2022, pp. 103–110.

[12] C. D. Bellicoso, F. Jenelten, C. Gehring, and M. Hutter, "Dynamic locomotion through online nonlinear motion optimization for quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2261–2268, 2018.

[13] F. Jenelten, J. Hwangbo, F. Tresoldi, C. D. Bellicoso, and M. Hutter, "Dynamic locomotion on slippery ground," *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 4170–4176, 2019.

[14] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Science robotics*, vol. 7, no. 62, p. eabk2822, 2022.

[15] M. Bjelonic, R. Grandia, O. Harley, C. Galliard, S. Zimmermann, and M. Hutter, "Whole-body mpc and online gait sequence generation for wheeled-legged robots," in *2021 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2021, pp. 8388–8395.

[16] V. Klemm, Y. de Viragh, D. Rohr, R. Siegwart, and M. Tognon, "Non-smooth trajectory optimization for wheeled balancing robots with contact switches and impacts," *IEEE Transactions on Robotics*, 2023.

[17] J. Lee, M. Bjelonic, and M. Hutter, "Control of wheeled-legged quadrupeds using deep reinforcement learning," in *Climbing and Walking Robots Conference*. Springer, 2022, pp. 119–127.

[18] J. Lee, M. Bjelonic, A. Reske, L. Wellhausen, T. Miki, and M. Hutter, "Learning robust autonomous navigation and locomotion for wheeled-legged robots," *Science Robotics*, vol. 9, no. 89, p. eadi9641, 2024.

[19] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa *et al.*, "Isaac gym: High performance gpu-based physics simulation for robot learning," *arXiv preprint arXiv:2108.10470*, 2021.

[20] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2017, pp. 23–30.

[21] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 2012, pp. 5026–5033.

[22] X. Gu, Y.-J. Wang, and J. Chen, "Humanoid-gym: Reinforcement learning for humanoid robot with zero-shot sim2real transfer," *arXiv preprint arXiv:2404.05695*, 2024.

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

[24] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Lillicrap, T. Harley, D. Silver, and K. Kavukcuoglu, "Asynchronous methods for deep reinforcement learning," in *International conference on machine learning*. PmLR, 2016, pp. 1928–1937.

[25] I. Grondman, L. Busoniu, G. A. Lopes, and R. Babuska, "A survey of actor-critic reinforcement learning: Standard and natural policy gradients," *IEEE Transactions on Systems, Man, and Cybernetics, part C (applications and reviews)*, vol. 42, no. 6, pp. 1291–1307, 2012.

[26] G. Ji, J. Mun, H. Kim, and J. Hwangbo, "Concurrent training of a control policy and a state estimator for dynamic and robust legged locomotion," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 4630–4637, 2022.

[27] L. Ye, J. Li, Y. Cheng, X. Wang, B. Liang, and Y. Peng, "From knowing to doing: learning diverse motor skills through instruction learning," *arXiv preprint arXiv:2309.09167*, 2023.