IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

A Practical Multi-Body Model Enabling a Flexible-Wheeled Robot to Learn Blind Stair Climbing

Chan-Young Yoon and Baek-Kyu Cho, *Member, IEEE*

***Abstract*—Controlling a flexible wheeled robot for complex tasks such as stair climbing is highly challenging. The nonlinearity inherent in soft materials hinders accurate modeling, creating a trade-off in Reinforcement Learning (RL) between simulation fidelity and learning speed. We propose an RL-friendly, multi-body model that approximates the deformation of the flexible wheel as a Mass-Spring-Damper (MSD) system composed of rigid links and joints. This model enables end-to-end RL within a fast rigid-body simulator, facilitating a blind control policy that relies solely on proprioceptive feedback. To reduce the reality gap and enhance policy robustness, we randomize the main parameters of the MSD system. In real-world experiments, a robot successfully climbed an 18 cm step, corresponding to approximately 51% of the wheel radius—a feat impossible for a rigid-wheeled equivalent. To our knowledge, this is the first successful application of RL-based blind control for stair climbing with a flexible wheeled robot. However, structural limitations in our model and challenges in parameter identification hinder sim-to-real transfer, and improving robustness remains a key issue for future work.**

***Index Terms*—Reinforcement learning, wheeled robots, flexible robotics, modeling, control, and learning for soft robots.**

I. Introduction

SOFT robots use compliance and adaptability to solve problems intractable for rigid robots. This ability comes from absorbing impacts in unpredictable environments or handling fragile objects [1], [2], [3]. Its potential is recognized in healthcare, logistics, exploration, and underwater robotics [4].

The flexible wheel, a key application of soft robotics in locomotion, has gained significant attention [5], [6]. This design endows wheeled robots, known for their energy efficiency on flat terrain, with obstacle negotiation capabilities. Stair climbing is a critical benchmark in ground mobile robotics because it demands precise control over robot posture, balance, and propulsion. Therefore, enabling a flexible wheeled robot to climb stairs is a compelling research goal.

Paradoxically, deformation—the main strength of soft robots—is the most significant challenge from a dynamic control perspective. The complex and nonlinear behavior of soft materials increases system uncertainty and complicates modeling. Despite various proposed mathematical approaches [7], this limitation is especially pronounced for a flexible wheel, which endures repetitive, high-speed impacts during locomotion.

Reinforcement Learning (RL) is a powerful tool to overcome the challenges of traditional model-based control [8]. RL learns an optimal policy through environmental interaction, removing the need for a perfect system model. However, RL relies on fast simulators, creating a dilemma for soft robotics. The Finite Element Method (FEM), the most accurate simulation for deformable bodies, is too computationally expensive for RL training. This creates a trade-off between high-fidelity simulation and RL learning speed [9].

This paper proposes a practical solution. While compliant wheels and Mass-Spring-Damper (MSD) models are established concepts, we introduce a novel integration method to construct an RL-friendly multi-body model for blind stair climbing. Specifically, we approximate the continuous deformation of the flexible wheel with a multi-body model of rigid links and joints governed by the MSD system. This model is based on physically realizable behaviors, allowing natural simulation of key deformations and contact forces during RL. With relatively low computational cost, this wheel structure integrates effectively with a high-speed rigid-body simulator like NVIDIA Isaac Gym [10] for subsequent transfer to reality (Fig. 1).

Furthermore, a core contribution of this work is validating a blind stair-climbing policy in the real-world without relying on visual sensors. The policy is trained to infer ground contact and ascend stairs using only proprioceptive feedback obtained from an IMU and motor encoders, rather than exteroceptive sensors like a camera or LiDAR. We bridged the reality gap by applying Dynamics Randomization (DR) [11] to the stiffness and damping coefficients of the MSD system governing wheel deformation.

Received 30 September 2025; accepted 3 January 2026. Date of publication 9 February 2026; date of current version 20 February 2026. This article was recommended for publication by Associate Editor E. Milana and Editor C. Della Santina upon evaluation of the reviewers' comments. This work was supported in part by the BK21 Four Program under Grant 2120240815267 of the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Korea, and in part by the Institute of Information & Communications Technology Planning & Evaluation (IITP) Grant funded through the Korea government (MSIT) under Grant RS-2025-02219317, AI Star Fellowship (Kookmin University). *(Corresponding author: Baek-Kyu Cho.)*

Chan-Young Yoon is with the Department of Mechanical System Engineering, Kookmin University, Seoul 02707, South Korea (e-mail: keoungml0@gmail.com).

Baek-Kyu Cho is with the Robotics and Control Laboratory, School of Mechanical Engineering, Kookmin University, Seoul 02707, South Korea (e-mail: swan0421@gmail.com).

This article has supplementary downloadable material available at https://doi.org/10.1109/LRA.2026.3662534, provided by the authors.

Digital Object Identifier 10.1109/LRA.2026.3662534

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

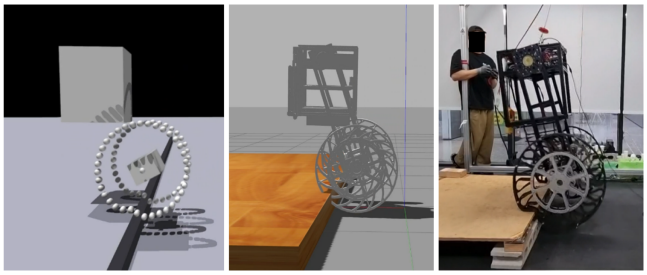

Fig. 1. Sim-to-real transfer of the blind control policy using the proposed multi-body model.

The contributions of this paper are as follows:

1) A practical multi-body model is proposed for the physically plausible and efficient simulation of deformation and contact forces in the flexible wheel.
2) To the best of our knowledge, this is the first work to implement end-to-end RL-based control for the blind stair-climbing task of a wheeled robot by integrating the proposed model with a rigid-body simulator.
3) The control policy, learned in simulation, was transferred to a real flexible wheeled robot, demonstrating the capability to ascend an 18 cm high step, which is not possible with a conventional rigid wheel model.

II. Related Work

This section reviews control strategies for robots with deformable bodies. We focus on RL-based dynamic motion control, distinct from mathematical model-based [7], [12] or data-driven approaches [2].

A. Deformable Body Simulation Approaches for RL

Research combining the dynamics of deformable bodies with RL must balance the conflicting goals of simulation accuracy and learning speed. The Finite Element Method (FEM) offers the highest physical accuracy. Frameworks like Sofa [13] can accurately simulate complex deformations, while the Incremental Potential Contact (IPC) model ensures stability by preventing interpenetration [14], [15]. However, such high-fidelity simulations are inherently associated with high computational costs.

Efforts to improve FEM efficiency include Model Order Reduction (MOR), which accelerates computation by compressing high-dimensional models [16], [17]. Intermediate simulators that strike a compromise between the accuracy of FEM and the speed of rigid-body simulation have also been proposed. Elastica [18] efficiently computes the complex dynamics of soft, slender structures using the Cosserat Rod model. SoMo [19] presents a practical approach by approximating a continuum robot as a system of multiple rigid links and springs in the PyBullet engine [20]. Toolkits developed to facilitate the use of RL for soft robot control, such as SofaGym [21] and SoMoGym [22], provide various benchmark environments. Furthermore, simulators like SoftGym [23], ReForm [24], and DefGraspSim [25] enable the grasping and manipulation of deformable objects like cloth, ropes, and fruit.

However, these methods focus on fully soft robots, arms, or grippers. The hybrid robot has a rigid chassis with flexible wheels, where local wheel deformation is the key aspect. Applying these approaches is inefficient and risks excessively modeling parts irrelevant to the control objective. Moreover, they do not address the problem of locomotion involving dynamic interaction with discontinuous terrain, such as stairs.

Approximating robots with simplified models for RL has proven successful. Jitosho et al. [26] modeled a soft arm as a rigid-link system, using NVIDIA Isaac Gym [10] and DR for agile dynamic maneuvers. This work showed that a simplified model with RL can solve complex dynamic control problems without high-fidelity FEM. Accordingly, we adopt this practical approach, using an RL-friendly multi-body model in a high-speed simulator to capture the key behaviors of the flexible wheel.

B. RL-Based Blind Stair Climbing and Sim-to-Real

Blind stair climbing is a key challenge in robot control that demands a high level of robustness. Impressive achievements using RL-based controllers have been reported [27]. Siekmann et al. [28] enabled the bipedal robot Cassie to climb real stairs using only proprioceptive feedback via terrain randomization. Chamorro et al. [29] induced reflexive behaviors in legged and wheeled-legged robots, enabling them to climb stairs through the use of privileged terrain information and a position-based formulation. However, these successes have been largely confined to multi-legged robots. Although hybrid wheeled-legged robots have emerged, they remain complex systems that require the precise, coordinated control of numerous actuators.

Structurally simpler wheeled robots struggle to achieve similar mobility with RL. Prior work demonstrated that a flexible wheeled robot uses wheel deformation for terrain adaptability, enabling it to ascend stairs [5]. Unlike the distinct point contacts of legged robots, a flexible wheel creates surface contact with continuous deformation. This fundamentally complicates the inference and control of contact states. This uncertainty highlights the limitations of model-based approaches and illustrates why model-free RL control is a promising solution. Nonetheless, research on efficiently simulating the complex deformation and contact of flexible wheels to train blind control policies remains scarce.

Policies from simulation inevitably face a reality gap, stemming from parameter mismatches, sensor noise, and actuator characteristics. This gap is exacerbated in dynamic tasks with frequent, unpredictable contacts, and bridging it through randomization has become a consistent trend [30], [11]. In this paper, the uncertain wheel deformation is another critical factor. We address this by randomizing the stiffness and damping coefficients of the MSD system that govern this deformation. Varying these parameters widely in simulation produces a robust controller, even without precise knowledge of the true physical values.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

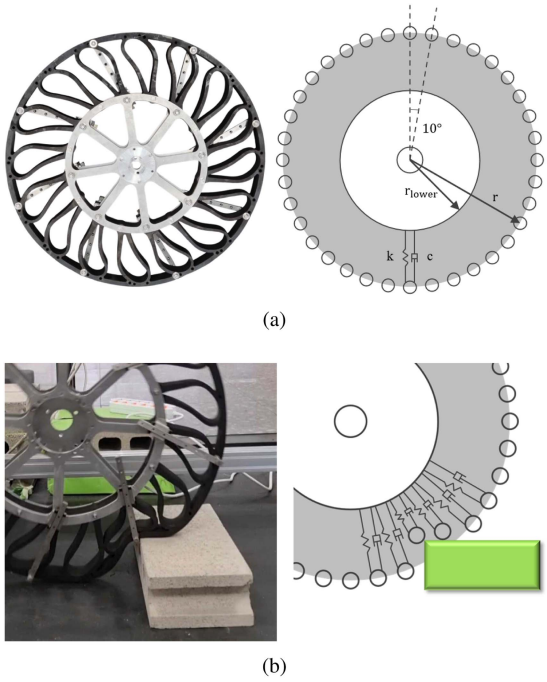

Fig. 2. Proposed multi-body model for simulating the flexible wheel. (a) Structure of the wheel model composed of spherical rigid links and the MSD system. (b) An illustration of the wheel deformation during stair climbing, as approximated by the MSD system.

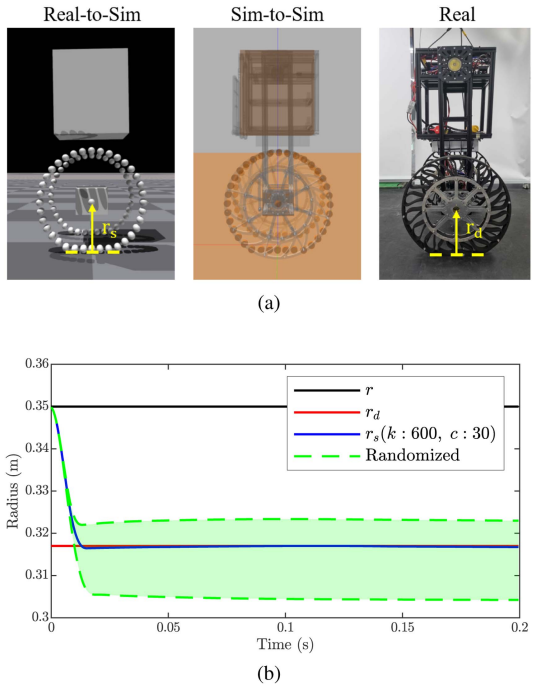

Fig. 3. The MSD system of wheel parameter identification and randomization. (a) Real-to-Sim (Left), Sim-to-Sim (Middle), and Real deformed radius r_{d} under self-weight (Right). (b) Identification of stiffness k, damping c and simulated deformation range after randomization.

III. Method

A. Real-to-Sim

The robot uses the flexible wheels detailed in [5], composed of waterjet-cut Styrene Butadiene Rubber (SBR) wheels with a width of 40 mm. We emulate this using the multi-body model in Fig. 2(a), where spherical rigid links are arranged at 10° intervals. However, this discrete structure inherently neglects the circumferential coupling of the continuous rim. A prismatic joint connects each link from its radius before deformation r to its radius after maximum deformation r_{lower}. According to [5], $r = 0.35$ m and $r_{\mathrm{lower}} = 0.25$ m. The joint motion, which functions as the MSD system, is determined by stiffness k and damping c . These coefficients are implemented via native joint PD position control in Isaac Gym to set the radius after deformation [10]. Fig. 2(b) shows this MSD-based interaction causing wheel deformation on a step. Since the SBR (Young's modulus: 4 MPa) exhibits rate-dependent viscoelasticity, this MSD model cannot capture the complex dynamic contact patches. To compensate for these unmodeled dynamics including the structural discrepancy, we employ a wide range of parameter randomization.

Static deformation is physically distinct from dynamic deformation arising from high-speed impacts, such as collisions with step edges. Deformation velocity affects the damping force, increasing nonlinearity. Since accurate parameter identification in dynamic environments is another complex research topic in itself, we focus on learning a robust policy by applying a wide range of DR. This paper assumes that the distribution of k and c, identified under static load conditions, sufficiently covers uncertainties in dynamic tasks. As shown on the right in Fig. 3(a), the radius after deformation due to the self-weight of the robot on even terrain in reality is identified as $r_{\mathrm{d}} = 0.317$ m. In simulation, we adjust k and c to reproduce this deformation, yielding r_{s}. The corresponding values, $k = 600$ and $c = 30$, were then randomized using a uniform distribution over ranges of $\pm 40\%$ for k and $\pm 20\%$ for c. Fig. 3(b) is a graph showing the radius in reality and simulation on even terrain, and it illustrates the randomized range of deformation. Since k and c govern key dynamic behaviors (deformation, restoration, shock absorption), a policy trained on their distribution is expected to handle unexpected contacts effectively. This process exposes the policy to diverse deformation scenarios, enhancing controller robustness.

B. Training Setup

Finding the optimal policy for the proposed multi-body model requires large-scale training data. We used NVIDIA Isaac Gym [10] with 512 parallel environments for efficient data collection. We adopted the Proximal Policy Optimization (PPO) algorithm [31] and introduced an Asymmetric Actor-Critic architecture to enhance learning efficiency. Fig. 4 illustrates the overall learning framework that incorporates these components. The actor network uses only observations available from the physical hardware, including IMU and motor encoder data. In contrast, the critic network uses additional privileged, simulation-only information for a more accurate value function estimate. This accurate estimate from the critic helps effectively update the actor policy, which relies on limited observations.

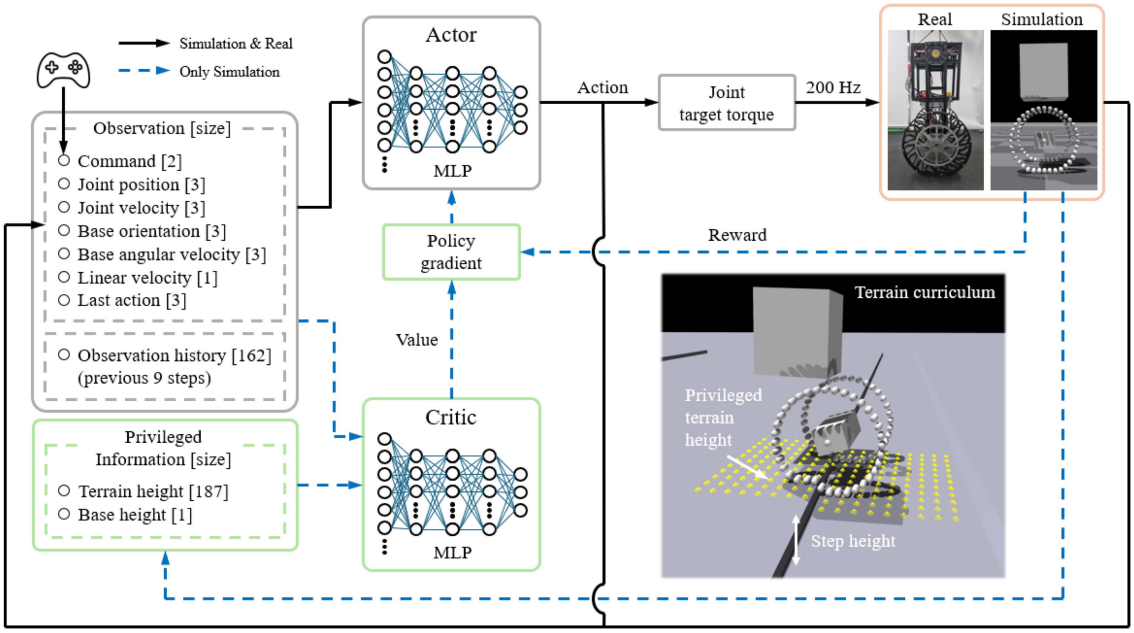

Fig. 4. Reinforcement learning framework for blind stair climbing.

TABLE I
HYPERPARAMETERS FOR THE LEARNING

Category	Parameter	Value
Actor Network	Neural network	MLP
	Hidden layer units	[512, 256, 128]
	Activation	ELU
	Learning rate	5e-4
Critic Network	Neural network	MLP
	Hidden layer units	[512, 256, 128]
	Activation	ELU
	Learning rate	3e-4
PPO Algorithm	Discount factor	0.99
	GAE Discount factor	0.95
	Clip range	0.2
	Entropy coefficient	0.001
	Horizon length	48
	Mini batch size	4096

The detailed hyperparameters are summarized in Table I. All training was performed on a desktop computer with an Intel Core i5-9400F CPU and an NVIDIA GeForce RTX 3060 Ti GPU.

C. Training Details

This paper adopts several ideas from Chamorro et al. [29] for composing privileged information and designing the terrain curriculum [27], [32]. Unlike their position-based navigation task [33], we formulate our task as standard velocity-based tracking. Consequently, the reward function is designed differently to match our control objectives.

1) Observation Space: The observation space $\boldsymbol{o}_t$ and the privileged information $\boldsymbol{o}_t^p$ at time step t are defined as follows:

$$\boldsymbol{o}_t = [\boldsymbol{v}_{\text{cmd}}, \boldsymbol{q}, \dot{\boldsymbol{q}}, \boldsymbol{\Theta}, \boldsymbol{\omega}, v_{b,x}, \boldsymbol{a}_{t-1}, \boldsymbol{o}_h]. \tag{1}$$

$$\boldsymbol{o}_t^p = [\boldsymbol{h}_{\text{terrain}}, h_{\text{base}}]\,. \tag{2}$$

Here, $\boldsymbol{v}_{\text{cmd}}$ includes the linear velocity command for the base in the x-direction $v_{\text{cmd},x}$, and the angular velocity command for the base in the z-direction $\omega_{\text{cmd},z}$. The terms $\boldsymbol{q} = [\theta_{\text{head}}, \theta_{\text{left}}, \theta_{\text{right}}]$ and $\dot{\boldsymbol{q}} = [\dot{\theta}_{\text{head}}, \dot{\theta}_{\text{left}}, \dot{\theta}_{\text{right}}]$ are joint states measured by motor encoders while $\boldsymbol{\Theta} = [\phi_{\text{base}}, \theta_{\text{base}}, \psi_{\text{base}}]$ and $\boldsymbol{\omega} = [\omega_{b,x}, \omega_{b,y}, \omega_{b,z}]$ represent base states obtained from an IMU utilizing Attitude and Heading Reference System (AHRS). The linear velocity of the base in the x-direction $v_{b,x}$ is calculated from the radius and velocity of the wheels, and the last actions $\boldsymbol{a}_{t-1}$ are included in the observation space. The observation history $\boldsymbol{o}_h$ consists of the observations from the previous 9 time steps. The privileged information consists of height data, which is difficult to acquire from a real robot without external sensors. $\boldsymbol{h}_{\text{terrain}}$ comprises height measurements sampled at uniform 0.1 m intervals over a 1.6×1.0 m grid around the robot, and h_{base} is the height of the base.

2) Action Space: The action space defines the policy output $\boldsymbol{a}_t$ which is mapped to the target joint torques:

$$\boldsymbol{a}_t = [a_{\text{head}}, a_{\text{left}}, a_{\text{right}}]. \tag{3}$$

$$\boldsymbol{\tau}_t = \tau_{\max} \cdot \boldsymbol{a}_t = \left[\tau_{\text{head}}^{\text{des}}, \tau_{\text{left}}^{\text{des}}, \tau_{\text{right}}^{\text{des}}\right]. \tag{4}$$

The policy learns to translate full observation $\boldsymbol{o}_t$ directly into the target joint torques $\boldsymbol{\tau}_t$ at 200 Hz.

3) Reward: The reward function encourages accurate tracking of command velocities. Primarily, tracking is enhanced by minimizing error. A critical design feature, determined heuristically, is that orientation control is handled separately. We apply a severe penalty on absolute yaw deviation, distinct from the angular velocity reward. This ensures the policy actively corrects its heading against wheel slip, prioritizing stable direction over temporary speed. Secondary terms penalize excessive angular velocities and head position. Abrupt actions are also penalized

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

TABLE II
REWARD FUNCTIONS

Reward	Mathematical Expression	Weight
Linear velocity tracking	$\exp(-4.0 \cdot \lvert v_{\text{cmd},x} - v_{b,x} \rvert)$	1.2
Angular velocity tracking	$\exp(-2.0 \cdot \lvert \omega_{\text{cmd},z} - \omega_{b,z} \rvert)$	0.3
Base yaw posture	ψ^2_{base}	-10.0
Head pitch posture	θ^2_{head}	-2.0
Base pitch stability	$\lvert \dot{\theta}_{\text{base}} \rvert$	-0.4
Head pitch stability	$\lvert \dot{\theta}_{\text{head}} \rvert$	-0.4
Action rate	$\sum(\boldsymbol{a}_{t-1} - \boldsymbol{a}_t)^2$	-0.02

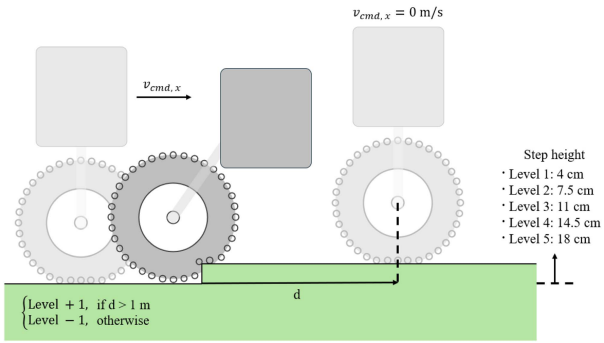

Fig. 5. Illustration of the terrain curriculum training protocol.

TABLE III
RANDOMIZATION CONFIGURATION

Parameter	Range	Distribution	Operation
Initial Pitch	[-0.25, 0.25] rad	Uniform	Reset
Initial Yaw	[-0.1, 0.1] rad	Uniform	Reset
Initial Head Joint Position	[-0.15, 0.15] rad	Uniform	Reset
Initial Wheel Joint Position	[-0.3, 0.3] rad	Uniform	Reset
Initial Head Joint Velocity	[-0.2, 0.2] rad/s	Uniform	Reset
Initial Wheel Joint Velocity	[-0.5, 0.5] rad/s	Uniform	Reset
Link Mass	±30%	Uniform	Scaling
Gravity	[0, 0.1]	Gaussian	Additive
Observation Noise	[0, 0.008]	Gaussian	Additive
Ground Friction	[0.5, 1.2]	Uniform	Scaling
Rolling Friction	[0.001, 0.02]	Uniform	Additive
Torsion Friction	[0.001, 0.01]	Uniform	Additive
Ground Restitution	[0, 0.2]	Gaussian	Additive
Response Coefficient	[0.2, 1.0]	Uniform	Sampling
Joint Damping	±20%	Uniform	Scaling
Joint Friction	±20%	Uniform	Scaling
Joint Armature	±20%	Uniform	Scaling
Maximum Joint Velocity	±20%	Uniform	Scaling
Maximum Joint Torque	±20%	Uniform	Scaling

for smooth motion. Table II details the mathematical formulation and weight for each reward term.

4) Task: This paper investigates whether our proposed model and RL policy can overcome the dynamic uncertainties and model discrepancies of a real robot. We define the core task as stair climbing, the most challenging scenario. The 18 cm target height corresponds to approximately 51% of the wheel radius and matches standard architectural specifications. The curriculum has five levels with step heights from 4 cm to 18 cm. Training begins at the easiest level. The agent advances to the next level after climbing a step and moving forward over 1 m. Upon failure, it reverts to the last mastered level. This difficulty adjustment promotes progressive learning through sequential mastery of each level.

The training protocol (Fig. 5) is designed for climbing behavior to emerge from the terrain curriculum and velocity tracking, without an explicit reward for ascending the step. Each episode lasts for 10 seconds. During the initial 9 seconds, a forward velocity command within the range of [0.5, 0.6] m/s is issued. For the final second, a stop command is given. This sequence teaches the robot stable stopping in conjunction with climbing. The forward command range ensures the minimum momentum and wheel deformation needed to climb the step, which requires a certain speed. If the robot gets stuck at the step, the large velocity error results in a low reward. This implicitly drives the agent to learn step-climbing strategies as part of the velocity tracking task. In 10% of environments, only a stop command is given for the full episode to teach stationary stability.

D. Randomization Strategy

To facilitate successful sim-to-real transfer, we train the policy over a distribution of many plausible environments to ensure robust operation. In addition to randomizing the flexible wheel, we apply a multi-layered randomization to standard dynamics parameters, including initial states, external disturbances, and ground friction.

Diverse initial states account for potential variations in a real robot. At the start of each episode, we randomize joint positions, velocities, and the robot body pose to enhance policy generalization. To build resilience against external forces, random impulses of $[-0.5, 0.5]$ m/s and $[-0.5, 0.5]$ rad/s are applied to $v_{b,x}$ and $\omega_{b,z}$ every 3 seconds. This teaches the policy to rapidly restore posture and resume velocity tracking after a loss of balance.

Dynamics parameters are also randomized to compensate for simulation physics model inaccuracies. To reflect the surface contact of the flexible wheel, we include rolling and torsion friction, which impede wheel rotation. Physical properties like actuator torque and velocity limits are randomized once at environment creation. This reflects that actuator characteristics do not change abruptly during operation. A policy trained over this distribution of uncertainties acquires robustness against unpredictability. The complete set of randomization parameters and their detailed configurations are summarized in Table III.

IV. EXPERIMENTAL VALIDATION

This section validates the robustness and real-world transferability of the control policy learned in simulation. First, we assess policy generalization by transferring it to a simulator with a different physics engine. We then validate sim-to-real transfer to a real robot climbing an 18 cm step and analyze the resulting reflexive behaviors.

A. Sim-to-Sim Transfer

Successful sim-to-real transfer requires a policy that generalizes beyond its training environment and does not overfit. While

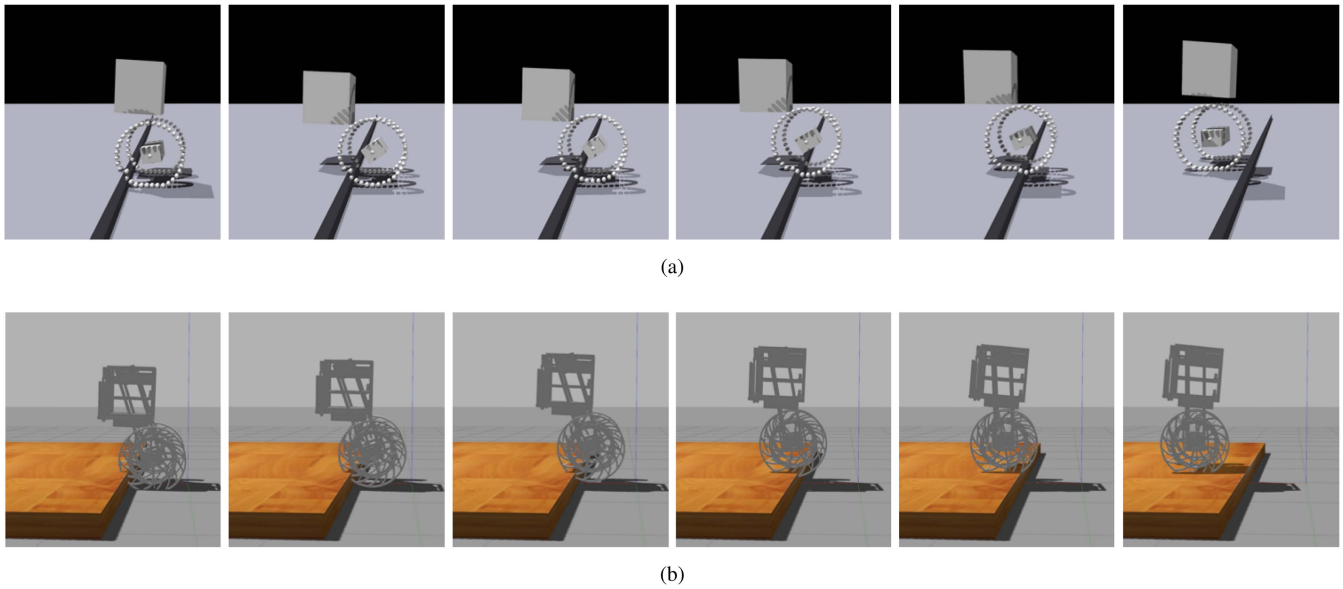

Fig. 6. Experimental results of control policy in simulation. (a) Training in Isaac Gym. (b) Sim-to-Sim transfer: from Isaac Gym to Gazebo.

TABLE IV
COMPARISON OF SIMULATOR PERFORMANCE

	Isaac Gym (PhysX)	Gazebo (ODE)
Success Rate (%)	89	83
Peak Base Pitch (rad)	0.56 $\sim$ 0.74	0.45 $\sim$ 0.58
Peak Power (W)	582 $\sim$ 943	653 $\sim$ 975

our training simulator is fast, it is optimized for the NVIDIA PhysX engine [32]. To assess policy robustness, we conducted cross-validation in a different simulation environment. We chose Gazebo as our validation platform, a widely used simulator with strong ROS integration for physical robot experiments.

Although PhysX and the Open Dynamics Engine (ODE) used by Gazebo employ similar numerical integration methods for handling contact and friction dynamics, they have a fundamental difference: PhysX intentionally neglects Coriolis forces to prioritize speed [34], [35]. In the proposed multi-body model, numerous rigid links undergo discontinuous and repetitive impacts with step edges. Such complex contact interactions can yield vastly different outcomes depending on the implementation details of the physics engine. A policy that exploited PhysX-specific loopholes would likely fail in the ODE-based Gazebo environment. Thus, success in Gazebo without retraining indicates the policy learned general contact dynamics, not engine-specific quirks.

Although trained for velocity tracking, the policy was evaluated on its stair-climbing success rate. This is because instantaneous velocity error is an unreliable performance metric during highly dynamic tasks like stair climbing. A successful climb is a complex maneuver that may require temporarily sacrificing velocity tracking for stability and force. Therefore, while velocity tracking induces the behavior, the climbing success rate is the key performance metric for task completion.

Table IV compares performance on the 18 cm stair climbing task. Over 100 trials, the training (Fig. 6(a)) and validation (Fig. 6(b)) environments showed high success rates of 89% and 83% , respectively, demonstrating generalization. Notably, Gazebo showed a lower peak base pitch and higher peak power, despite an identical wheel model implementation. We interpret this as the policy adapting to more challenging physics by using a stable, high-power strategy over a dynamic, Center of Gravity (CoG) maneuver. This indicates the policy is robust enough to adapt to altered dynamics.

B. Sim-to-Real Transfer

To validate the necessity of the proposed multi-body model, we set a rigid wheel model baseline. This baseline failed to learn a meaningful climbing strategy, as it was unable to move forward after colliding with the step. This clearly shows the task is fundamentally impossible for a rigid wheel model.

Building on simulation validation, we transferred the multi-body policy to hardware for final performance evaluation. Experiments with the robot and an 18 cm step recorded a 10% success rate, representing 1 of 10 trials. Fig. 7(a) shows the one successful attempt. This success occurred when learned dynamic strategies, like wheel deformation and CoG shifts, were ideally reproduced. The measured profiles in Fig. 7(b) included a peak base pitch of 0.37 rd and a peak power of 771 W. A comparison with the simulation results in Table IV reveals a clear trend. As the physical gap from the training environment widened, the policy consistently adopted a stability-focused strategy characterized by a lower peak pitch and higher peak power.

However, the high failure rate shows a reality gap where the structural limitations exacerbate unmodeled dynamics, specifically the lack of circumferential coupling and complex material dynamics, such as rate-dependent effects. Fig. 8 details the

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

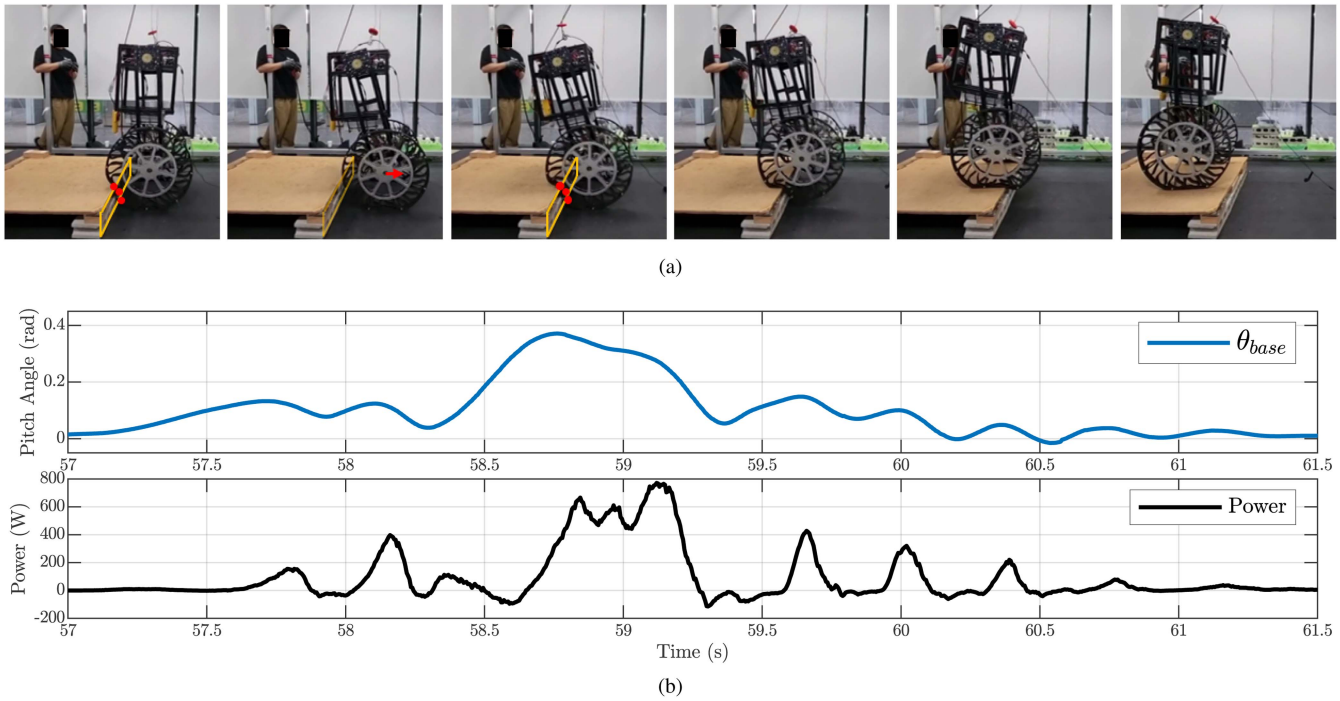

Fig. 7. Successful Sim-to-Real transfer. (a) Snapshot of the robot exhibiting a reflexive back-off and subsequent forward motion to overcome an 18 cm step. (b) Corresponding profiles of the base pitch angle and power consumption.

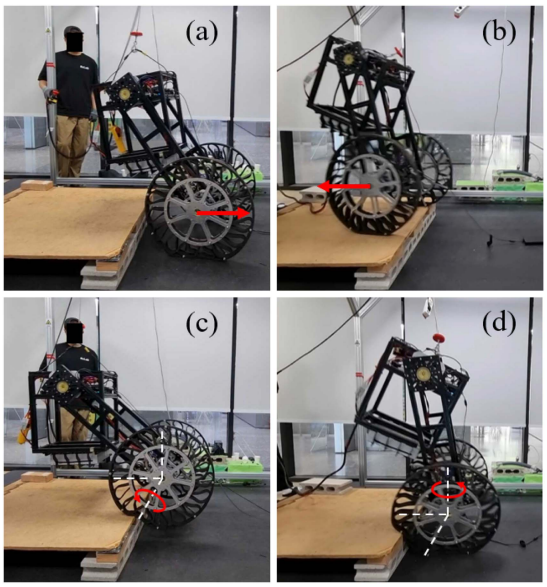

Fig. 8. Classification of observed failure modes during the Sim-to-Real transfer experiments. (a) Reflexive action overshoot (4 trials). (b) Forward somersault after climbing (2 trials). (c) Excessive base pitch (2 trials). (d) Yaw deviation due to asymmetric contact (1 trial).

primary failure modes. In simulation, the multiple rigid links move independently due to the lack of tangential constraints. When a specific link caught on the edge of the step, it could act as an anchor. This allowed the robot to gain temporary support that enabled aggressive dynamic strategies with large CoG shifts. As the real wheel is a continuum, this interlocking phenomenon was not reproducible. Consequently, simulation-optimized aggressive maneuvers resulted in catastrophic real-world failures. This demonstrates that structural discrepancies go beyond simple parameter uncertainties, leading to transfer failure.

C. Reflexive Behavior Analysis

An interesting dynamic behavior was observed during the 18 cm real-world stair climbing experiments. As shown in Fig. 7(a), after a strong impact with the step, the robot did not fail. Instead, it performed a reflexive avoidance and retry maneuver, momentarily reversing before pushing forward again to climb. Similar emergent behaviors have been reported in multi-legged robots [27], [29], [36].

This emergent behavior is not pre-programmed but is a strategy learned by the blind policy using only proprioceptive feedback. Without vision, the policy detects an impasse by sensing abrupt changes in IMU pitch or motor torque. The learned optimal action is not to force forward motion, but to briefly reverse. This maneuver stabilizes posture and readjusts wheel-step contact, allowing the robot to reapply propulsive force from a more advantageous position. This demonstrates the policy has learned a high-level strategy beyond simple velocity tracking, enabling autonomous stability during unpredictable physical interactions.

V. Conclusion

We proposed an RL-friendly multi-body model for blind stair climbing with a flexible wheeled robot. The model approximates wheel deformation as a multi-body MSD system to enable policy learning in a high-speed simulator. Our end-to-end policy, trained with a terrain curriculum, for the first time performed an

18 cm stair climb in simulation and reality, a task impossible for a rigid wheel model. This controller also exhibited reflexive behaviors in response to unpredictable contacts.

However, a reduced sim-to-real success rate stemmed from key limitations. Structural discrepancies arose from approximating a continuum wheel with discrete bodies. Static-only parameter identification may not have been sufficient for dynamic events. The blind policy also cannot predict complex terrain like continuous stairs. Future work will bridge this gap to improve policy robustness and success rate. We will also address complex navigation challenges like continuous stair climbing by integrating external sensors and a predictive control scheme.

REFERENCES

[1] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, 2015.

[2] Z. Chen et al., "Data-driven methods applied to soft robot modeling and control: A review," *IEEE Trans. Autom. Sci. Eng.*, vol. 22, pp. 2241–2256, 2025.

[3] Z. Chen et al., "A survey on soft robot adaptability: Implementations, applications, and prospects [Survey]," *IEEE Robot. & Automat. Mag.*, doi: 10.1109/MRA.2025.3584346.

[4] C. Lee et al., "Soft robot review," *Int. J. Control, Autom. Syst.*, vol. 15, no. 1, pp. 3–15, 2017.

[5] G. Kim, H. Chung, and B.-K. Cho, "MOBINN: Stair-climbing mobile robot with novel flexible wheels," *IEEE Trans. Ind. Electron.*, vol. 71, no. 8, pp. 9182–9191, Aug. 2024.

[6] J.-Y. Lee et al., "Variable-stiffness–morphing wheel inspired by the surface tension of a liquid droplet," *Sci. Robot.*, vol. 9, no. 93, 2024, Art. no. eadl2067.

[7] C. Armanini, F. Boyer, A. T. Mathew, C. Duriez, and F. Renda, "Soft robots modeling: A structured overview," *IEEE Trans. Robot.*, vol. 39, no. 3, pp. 1728–1748, Jun. 2023.

[8] C. Laschi, T. G. Thuruthel, F. Lida, R. Merzouki, and E. Falotico, "Learning-based control strategies for soft robots: Theory, achievements, and future challenges," *IEEE Control Syst. Mag.*, vol. 43, no. 3, pp. 100–113, Jun. 2023.

[9] J. Gao, M. Y. Michelis, A. Spielberg, and R. K. Katzschmann, "Sim-to-real of soft robots with learned residual physics," *IEEE Robot. Autom. Lett.*, vol. 9, no. 10, pp. 8523–8530, Oct. 2024.

[10] V. Makoviychuk et al., "Isaac gym: High performance gpu-based physics simulation for robot learning," 2021, *arXiv:2108.10470*.

[11] X. B. Peng, M. Andrychowicz, W. Zaremba, and P. Abbeel, "Sim-to-real transfer of robotic control with dynamics randomization," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2018, pp. 3803–3810.

[12] D. A. Haggerty et al., "Control of soft robots with inertial dynamics," *Sci. Robot.*, vol. 8, no. 81, 2023, Art. no. eadd6864.

[13] F. Faure et al., "SOFA: A multi-model framework for interactive physical simulation," in *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*, Berlin, Germany: Springer, 2012, pp. 283–321.

[14] C. M. Kim, M. Danielczuk, I. Huang, and K. Goldberg, "IPC-graspSim: Reducing the sim2real gap for parallel-jaw grasping with the incremental potential contact model," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 6180–6187.

[15] J. A. Fernández-Fernández, R. Lange, S. Laible, K. O. Arras, and J. Bender, "Stark: A unified framework for strongly coupled simulation of rigid and deformable bodies with frictional contact," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2024, pp. 16888–16894.

[16] O. Goury and C. Duriez, "Fast, generic, and reliable control and simulation of soft robots using model order reduction," *IEEE Trans. Robot.*, vol. 34, no. 6, pp. 1565–1576, Dec. 2018.

[17] S. Tonkens, J. Lorenzetti, and M. Pavone, "Soft robot optimal control via reduced order finite element models," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 12010–12016.

[18] N. Naughton, J. Sun, A. Tekinalp, T. Parthasarathy, G. Chowdhary, and M. Gazzola, "Elastica: A compliant mechanics environment for soft robotic control," *IEEE Robot. Autom. Lett.*, vol. 6, no. 2, pp. 3389–3396, Apr. 2021.

[19] M. A. Graule, C. B. Teeple, T. P. McCarthy, G. R. Kim, R. C. S. Louis, and R. J. Wood, "SOMO: Fast and accurate simulations of continuum robots in complex environments," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 3934–3941.

[20] E. Coumans and Y. Bai, and , "PyBullet, a Python module for physics simulation for games, *Robot. Mach. Learn.*, Accessed: Feb. 15, 2026. [Online]. Available: http://pybullet.org

[21] P. Schegg et al., "SofaGym: An open platform for reinforcement learning based on soft robot simulations," *Soft Robot.*, vol. 10, no. 2, pp. 410–430, 2023.

[22] M. A. Graule, T. P. McCarthy, C. B. Teeple, J. Werfel, and R. J. Wood, "SoMoGym: A toolkit for developing and evaluating controllers and reinforcement learning algorithms for soft robots," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 4071–4078, Apr. 2022.

[23] X. Lin, Y. Wang, J. Olkin, and D. Held, "SoftGym: Benchmarking deep reinforcement learning for deformable object manipulation," in *Proc. Conf. Robot Learn.*, 2021, pp. 432–448.

[24] R. Laezza, R. Gieselmann, F. T. Pokorny, and Y. Karayiannidis, "Reform: A robot learning sandbox for deformable linear object manipulation," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2021, pp. 4717–4723.

[25] I. Huang et al., "Defgraspsim: Physics-based simulation of grasp outcomes for 3D deformable objects," *IEEE Robot. Autom. Lett.*, vol. 7, no. 3, pp. 6274–6281, Jul. 2022.

[26] R. Jitosho, T. G. W. Lum, A. Okamura, and K. Liu, "Reinforcement learning enables real-time planning and control of agile maneuvers for soft robot arms," in *Proc. Conf. Robot Learn.*, 2023, pp. 1131–1153.

[27] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Sci. Robot.*, vol. 5, no. 47, 2020, Art. no. eabc5986.

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

[29] 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 *Proc. IEEE Int. Conf. Robot. Autom.*, 2024, pp. 8081–8087.

[30] 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 *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 23–30.

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

[32] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Proc. Conf. Robot Learn.*, 2022, pp. 91–100.

[33] N. Rudin, D. Hoeller, M. Bjelonic, and M. Hutter, "Advanced skills by learning locomotion and local navigation end-to-end," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2022, pp. 2497–2503.

[34] N. Koenig and A. Howard, "Design and use paradigms for Gazebo, an open-source multi-robot simulator," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 3, 2004, pp. 2149–2154.

[35] T. Erez, Y. Tassa, and E. Todorov, "Simulation tools for model-based robotics: Comparison of bullet, havok, mujoco, ode and physx," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2015, pp. 4397–4404.

[36] D. Youm, H. Jung, H. Kim, J. Hwangbo, H.-W. Park, and S. Ha, "Imitating and finetuning model predictive control for robust and symmetric quadrupedal locomotion," *IEEE Robot. Autom. Lett.*, vol. 8, no. 11, pp. 7799–7806, Nov. 2023.