Learning Humanoid Standing-up Control across Diverse Postures

Tao Huang[2,1] Junli Ren[1,3] Huayi Wang[1,2] Zirui Wang[1,4] Qingwei Ben[1,5] Muning Wen[1,2]
Xiao Chen[1,5] Jianan Li[5] Jiangmiao Pang[1]
[1]Shanghai AI Laboratory [2]Shanghai Jiao Tong University [3]The University of Hong Kong
[4]Zhejiang University [5]The Chinese University of Hong Kong

Website: humanoid-standingup.github.io Code: https://github.com/OpenRobotLab/HoST

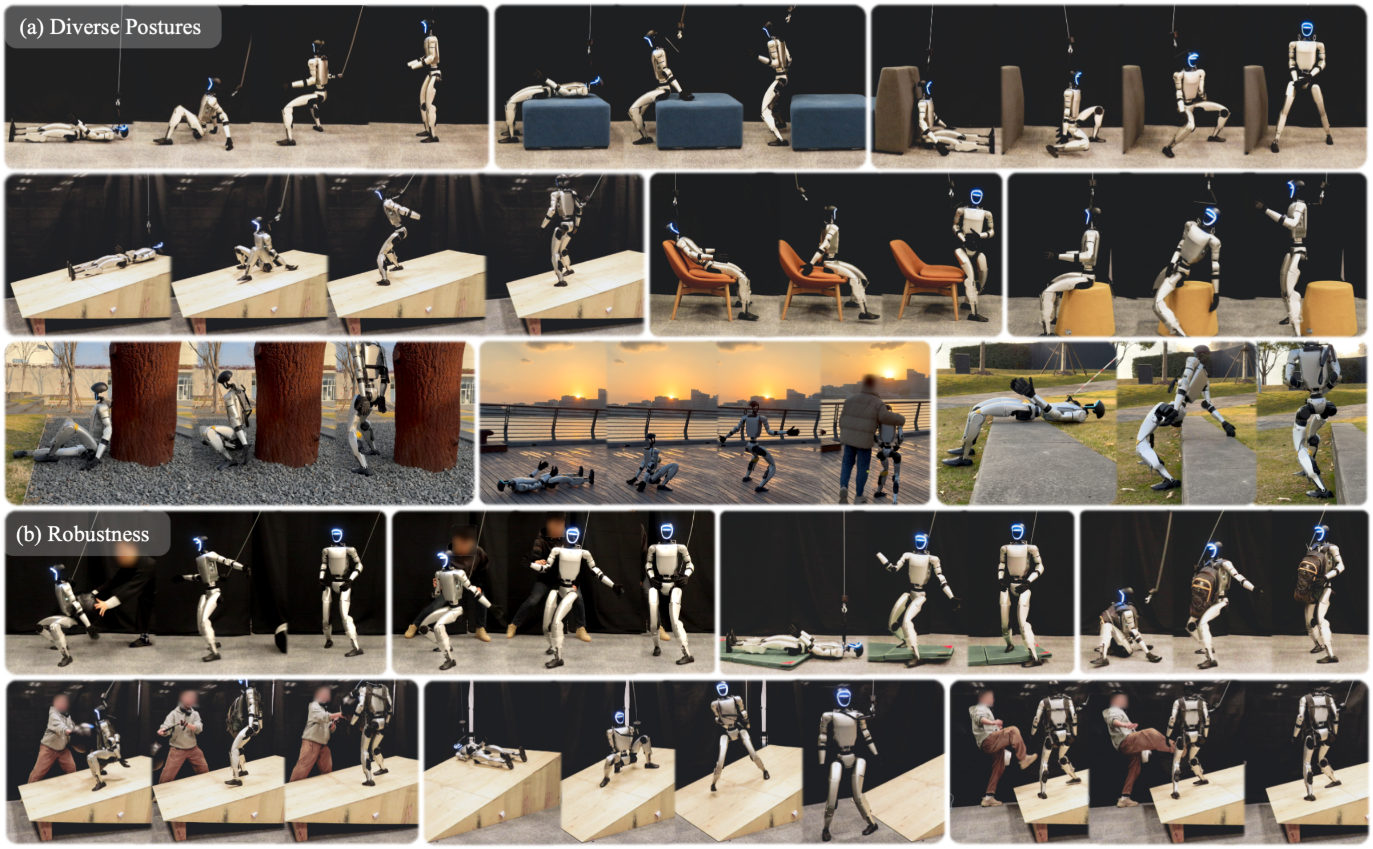


Fig. 1: Overview. (a) Our proposed framework HoST enables the humanoid robot to learn standing-up control via reinforcement learning without prior data, where the robot can successfully stand up across diverse postures in both laboratory and outdoor environments. (b) HoST also demonstrates strong robustness to many environmental disturbances, including external forces, stumbling blocks, 12kg payload, and challenging initial postures.

***Abstract*—Standing-up control is crucial for humanoid robots, with the potential for integration into current locomotion and loco-manipulation systems, such as fall recovery. Existing approaches are either limited to simulations that overlook hardware constraints or rely on predefined ground-specific motion trajectories, failing to enable standing up across postures in real-world scenes. To bridge this gap, we present HoST (Humanoid Standing-up Control), a reinforcement learning framework that learns standing-up control from scratch, enabling robust sim-to-real transfer across diverse postures. HoST effectively learns posture-adaptive motions by leveraging a multi-critic architecture and curriculum-based training on diverse simulated terrains. To ensure successful real-world deployment, we constrain the motion with smoothness regularization and implicit motion speed bound to alleviate oscillatory and violent motions on physical hardware, respectively. After simulation-based training, the learned control policies are directly deployed on the Unitree G1 humanoid robot. Our experimental results demonstrate that the controllers achieve smooth, stable, and robust standing-up motions across a wide range of laboratory and outdoor environments (Fig. 1). Videos and code are available on our project page.**

I. Introduction

Can humanoid robots stand up from a sofa, walk to a table, and pick up coffee, seamlessly like humans? Fortunately, recent advancements in humanoid robot hardware and control have enabled significant progress in bipedal locomotion [38, 26, 28, 55] and bimanual manipulation [5, 24, 9, 16], allowing robots to navigate environment and interact with objects effectively. However, the fundamental capability—standing-up control [43, 17]—remains underexplored. Most existing systems assume the robots start from a pre-standing posture, limiting their applicability to many scenes, such as transitioning from a seated position or recovering after a loss of balance. We envision that unlocking this standing-up capability would broaden the real-world applications of humanoid robots. To this end, we investigate how humanoid robots can learn to stand up across diverse postures in real environments.

A classical approach for this control task involves tracking handcrafted motion trajectories through model-based motion planning or trajectory optimization [17, 18, 22, 43]. Although effective in generating motions, these methods require extensive tuning of analytical models and often perform suboptimally in real-world settings with external disturbances [29, 23] or inaccurate actuator modeling [15]. Besides, real-time optimization on the robot makes these methods computationally intensive, prompting workarounds such as reduced optimization precision or offload computations to external machines [34, 8], though both are with practical limitations.

Reinforcement learning (RL) offers an alternative effective framework for humanoid locomotion and whole-body control [36, 13, 4, 54], benefiting from minimal modeling assumptions. However, compared to these tasks that partially decouple upper- and lower-body dynamics, RL-based standing-up control involves a highly dynamic and synergistic maneuver on both halves of the body. This complex maneuver features time-varying contact points [17], multi-stage motor skills [29], and precise angular momentum control [11], making RL exploration challenging. Although predefined motion trajectories can guide RL exploration, they are typically limited to ground-specific postures [35, 36, 52, 12], leaving the scalability to other postures unclear. Conversely, training RL agents from scratch with wide explorative strategies on the ground can lead to violent and abrupt motions that hinder real-world deployment [46], particularly for robots with many actuators and wide joint limits. In summary, learning posture-adaptive, real-world deployable standing-up control with RL remains an open problem (see Appendix B).

In this work, we address this problem by proposing HoST, an RL-based framework that learns humanoid standing-up control across diverse postures from scratch. To enable posture-adaptive motion beyond the ground, we introduce multiple terrains for training and a vertical pull force during the initial stages to facilitate exploration. Given the multiple stages of the task, we adopt multi-critic RL [33] to optimize distinct reward groups independently for a better reward balance. To ensure real-world deployment, we apply smoothness regularization and motion speed constraints to mitigate oscillatory and violent motions. Our control policies, trained in simulation [31] with domain randomization [48], can be directly deployed on the Unitree G1 humanoid robot. The resulting motions, tested in both laboratory and outdoor environments, demonstrate high smoothness, stability, and robustness to external disturbances, including forces, stumbling blocks, and heavy payloads.

We overview the real-world performance of our controllers in Fig. 1 and summarize our core contributions as follows:

TABLE I: Comparison with existing methods on standing-up control.

Method	Real Robot	w/o Prior Trajectory	Beyond Ground	High DoF	1-stage Training
Peng et al. [36]	✗	✗	✗	✓	✗
Yang et al. [52]	✗	✗	✗	✓	✓
Tao et al. [46]	✗	✓	✗	✓	✗
Haarnoja et al. [12]	✓	✗	✗	✓	✓
Gaspard et al. [10]	✓	✓	✗	✗	✓
HoST (ours)	✓	✓	✓	✓	✓

- **Real-world posture-adaptive motions** are well achieved through our proposed RL-based method, without relying on predefined trajectories or sim-to-real adaptation techniques.
- **Smoothness, stability, and robustness** are consistently demonstrated by our learned control policies, even under challenging external disturbances.
- **Evaluation protocols** are elaborately designed to analyze standing-up control comprehensively, aiming to guide future research and development in this control task.

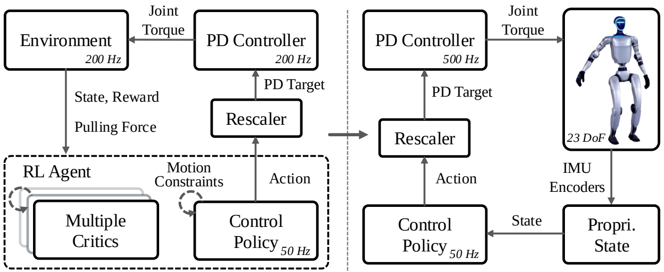
II. Related Work

A. Learning Humanoid Standing-up Control

Classical approaches to standing-up control rely on tracking handcrafted motion trajectories through model-based optimization [17, 18, 22, 43]. While effective, these methods are computationally intensive, sensitive to disturbances [29, 23], and require precise actuator modeling [15], limiting their real-world applicability. In contrast, RL-based methods learn control policies with minimal modeling assumptions, either by leveraging predefined motion trajectories to guide exploration [35, 36, 52, 12] or employing exploratory strategies to learn from scratch [46]. However, none of these methods have demonstrated real-world standing-up motion across diverse postures. Our proposed RL framework addresses these limitations by achieving posture adaptivity and real-world deployability without predefined motions, enabling smooth, stable, and robust standing-up across a wide range of laboratory and outdoor environments.

B. Reinforcement Learning for Humanoid Control

Reinforcement learning (RL) has been effectively applied to various humanoid control tasks, showcasing its versatility and effectiveness. For example, RL has enabled humanoid robots to achieve robust locomotion on diverse terrains [38, 26, 55, 28], whole-body control for expressive human-like motions [35, 36, 13, 14, 4], versatile jumping [54], and loco-manipulation [7, 27, 50]. Building on these advances, we address humanoid standing-up control, a parallel problem presenting unique challenges due to its dynamic nature and the need for precise coordination of multi-stage motor skills and time-varying contact points [17, 29]. We propose a novel approach that integrates a multi-critic framework, motion constraints, and a training curriculum to facilitate real-world deployment, setting it apart from prior methods.



(a) Training HoST in simulation

(b) Real-world deployment

Fig. 2: Framework overview. (a) We train policies in simulation from scratch with multiple critics and motion constraints operationalized by rewards, smoothness regularization, and action bound (rescaler). (b) The trained polices can be directly deployed in the real robot to produce standing-up motions.

C. Learning Quadrupedal Robot Standing-up Control

Standing-up control for quadrupedal robots shares similarities with humanoid robots but faces distinct challenges due to morphological differences, such as quadrupedal designs. Classical approaches for quadrupedal robots often rely on model-based optimization and predefined motion primitives [3, 40], which work well in controlled environments but struggle with adaptability to diverse postures and real-world uncertainties. Recent RL-based methods have enabled quadrupedal robots to recover from falls and transition between poses [23, 30, 52], using exploratory learning to manage complex dynamics and environmental interactions. Our work draws inspiration from these advances, extending them to humanoid robots by addressing the unique challenges of bipedal standing-up control. By incorporating posture adaptivity, motion constraints, and a structured training curriculum, our framework bridges the gap between quadrupedal and humanoid robot control, enabling robust standing-up motions across diverse environments.

III. Problem Formulation

We formulate the problem of humanoid standing up as a Markov decision process (MDP; [37]) with finite horizon, which is defined by the tuple $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, \mathcal{T}, \mathcal{R}, \gamma \rangle$. At each timestep t, the agent (*i.e.*, the robot) perceives the state $s_t \in \mathcal{S}$ from the environment and executes an action $a_t \in \mathcal{A}$ produced by its policy $\pi_\theta(\cdot|s_t)$. The agent then observes a successor state $s_{t+1} \sim \mathcal{T}(\cdot|s_t, a_t)$ following the environment transition function $\mathcal{T}$ and receives a reward signal $r_t \in \mathcal{R}$. To solve the MDP, we employ reinforcement learning (RL; [45]), whose goal learn an optimal policy π_θ that maximizes the expected cumulative reward (return) $\mathbb{E}_{\pi_\theta}[\sum_{t=0}^{T-1} \gamma^t r_t]$ the agent receives during the whole T-length episode, where $\gamma \in [0, 1]$ is the discount factor. The expected return is estimated by a value function (critic) V_ϕ. In this paper, we adopt Proximal Policy Optimization (PPO; [42]) as our RL algorithm because of its stability and efficiency in large-scale parallel training.

1) Observation Space: We hypothesize that the proprioceptive states of robots provide sufficient information for standing-up control in our target environments. We thus include the proprioceptive information read from robot's Inertial Measurement Unit (IMU) and joint encoders into the state $s_t = [\omega_t, r_t, q_t, p_t, \dot{p}_t, a_{t-1}, \beta]$, where ω_t is the angular velocity of robot base, r_t and q_t are the roll and pitch, p_t and $\dot{p}_t$ are positions and velocities of the joints, a_{t-1} is the last action, and $\beta \in (0, 1]$ is a scalar that scale the output action. Given the contact-rich nature of the standing-up task, we implicitly enhance contact detection by feeding the policy with the previous five states [15].

2) Action Space: We employ a PD controller for torque-based robot actuation. The action a_t represents the difference between the current and next-step joint positions, with the PD target computed as $p_t^d = p_t + \beta a_t$, where each dimension of a_t is constrained to $[-1, 1]$. The action rescalar β restricts the action bounds to regulate the motion speed implicitly. This is essential to constrain the standing-up motion and will be discussed in later sections. The torque at timestep t is computed as:

$$\tau_t = K_p \cdot (p_t^d - p_t) - K_d \cdot \dot{p}_t, \tag{1}$$

where K_p and K_d represent the stiffness and damping coefficients of the PD controller. The dimension of action space $|A|$ corresponds to the number of robot actuators.

IV. Method

This section introduces HoST (Humanoid Standing-up Control), a reinforcement learning (RL)-based framework for learning humanoid robots to stand up across diverse postures, as summarized in Fig. 2. This control task is highly dynamic, multi-stage, and contact-rich, posing challenges for conventional RL approaches. We first outline the key challenges addressed in this work in Section IV-A, then describe the core components of the framework in the following sections.

A. Key Challenges & Overview

1) Reward Design & Optimization (Section IV-B): The standing-up task involves multiple motor skills: righting the body, kneeling, and rising. Learning a control policy for these stages is challenging without explicit stage separation [25, 19]. We address this by dividing the task into three stages and activating corresponding reward functions at each stage. The complexity of these skills requires multiple reward functions, which can complicate policy optimization. To mitigate this, we employ multi-critic RL [33], grouping reward functions to balance objectives effectively.

2) Exploration Challenges (Section IV-C): Despite multi-critic RL, exploration remains difficult due to the robot's high degrees of freedom and wide joint limits. Drawing inspiration from human infant skill development [6, 49], we facilitate exploration by applying a curriculum-based vertical pulling force on the base link of the robot.

3) Motion Constraints (Section IV-D): With only reward functions, the agent tends to learn violent and jerky motions, driven by high torque limits and numerous actuators. Such behaviors are impractical for real-world deployment. To address this, we introduce an action rescaler β to gradually tighten action output bounds, implicitly limiting joint torques and motion speed. Additionally, we incorporate smoothness regularization [20] to mitigate motion oscillation.

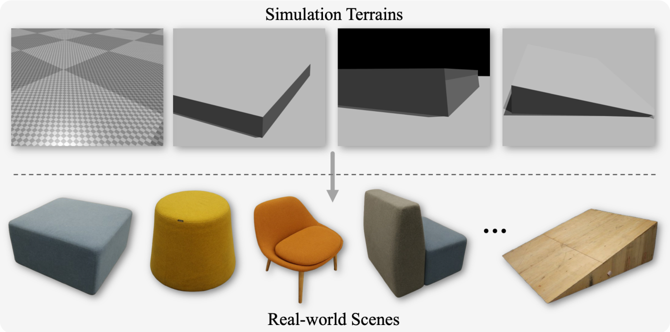


Fig. 3: Simulation terrains and real-world scenes. We design four terrains in simulation: ground, platform, wall, and slope to create initial robot postures that are likely to be met in real-world environments. Examples of these real-world environments are shown at the bottom of the figure.

TABLE II: Domain randomization settings for standing-up control.

Term	Value
Trunk Mass	$\mathcal{U}(-2, 5)$kg
Base CoM offset	$\mathcal{U}(-d, d)$m, $d = 0.12(XY), 0.08(Z)$
Link mass	$\mathcal{U}(-0.8, 1.2)\times$ default kg
Fiction	$\mathcal{U}(0.1, 1)$
Restitution	$\mathcal{U}(0, 1)$
P Gain	$\mathcal{U}(0.85, 1.15)$
D Gain	$\mathcal{U}(0.85, 1.15)$
Torque RFI [2]	$\mathcal{U}(-0.05, 0.05)\times$ torque limit N·m
Motor Strength	$\mathcal{U}(0.9, 1.1)$
Control delay	$\mathcal{U}(0, 100)$ms
Initial joint angle offset	$\mathcal{U}(-0.1, 0.1)$rad
Initial joint angle scale	$\mathcal{U}(0.9, 1.1)\times$ default joint angle rad

4) Sim-to-Real Gap (Section IV-E): A significant challenge is the sim-to-real gap. We address this through two strategies: (1) designing diverse terrains to better simulate real-world starting postures, and (2) applying domain randomization [48] to reduce the influence of physical discrepancies between simulation and the real world.

B. Reward Functions & Multiple Critics

Considering the multi-stage nature of the task, we divide the task into three stages: righting the body $h_{\text{base}} < H_{\text{stage1}}$, rising the body $H_{\text{stage1}} < h_{\text{base}} < H_{\text{stage2}}$, and standing $h_{\text{base}} > H_{\text{stage2}}$, indicated by the height of the robot base h_{base}. Corresponding reward functions are activated at each stage. We then classify reward functions into four groups: (1) **task reward** r^{task} that specifies the high-level task objectives, (2) **style reward** r^{style} that shapes the style of standing-up motion, (3) **regularization reward** r^{regu} that further regularizes the motionw, and (4) **post-task reward** r^{post} that specify the desired behaviors after successful standing up, *i.e.*, stay standing. The overall reward function is expressed as follows:

$$r_t = w^{\text{task}} \cdot r_t^{\text{task}} + w^{\text{style}} \cdot r_t^{\text{style}} + w^{\text{regu}} \cdot r_t^{\text{regu}} + w^{\text{post}} \cdot r_t^{\text{post}},$$

where w with superscript represents the corresponding reward weight. Each reward group contains multiple reward functions. A comprehensive list of all reward functions and groups is provided in Table VI.

However, we observe that using a single value function (critic) presents significant challenges in learning effective standing-up motions. Besides, the large number of reward functions makes hyperparameter tuning computationally intensive and difficult to balance. To address these challenges, we implement multiple critics (MuC; [33, 51, 53]) to estimate returns for each reward group independently, where each reward group is regarded as a separate task with its own assigned critic V_{ϕ_i}. These multiple critics are then integrated into the PPO framework for optimization as follows:

$$\mathcal{L}(\phi_i) = \mathbb{E}\left[\|r_t^i + \gamma V_{\phi_i}(s_t) - \bar{V}_{\phi_i}(s_{t+1})\|^2\right], \tag{2}$$

where r_t^i is the total reward and $\bar{V}_{\phi_i}$ is the target value function of reward group i. Each critic independently computes its advantage function A_{ϕ_i} estimated through GAE [41]. These individual advantages are then aggregated into an overall weighted advantage: $A = \sum_i w^i \cdot \frac{A_{\phi_i} - \mu_{A_{\phi_i}}}{\sigma_{A_{\phi_i}}}$, where $\mu_{A_{\phi_i}}$ and $\sigma_{A_{\phi_i}}$ are the batch mean and standard deviation of each advantage. The critics are updated simultaneously with the policy network π_θ according to:

$$\mathcal{L}(\theta) = \mathbb{E}\left[\min\left(\alpha_t(\theta)A_t, \text{clip}(\alpha_t(\theta), 1-\epsilon, 1+\epsilon)A_t\right)\right], \tag{3}$$

where $\alpha_t(\theta)$ and ϵ are the probability ratio and the clipping hyperparameter, respectively.

C. Force Curriculum as Exploration Strategy

The primary exploration challenges emerge during the transition from falling to stable kneeling, a stage that proves difficult to explore effectively through random action noise alone. While human infants are likely to learn motor skills with external supports [6, 49], it inspires us to design environmental assistance to accelerate the exploration. Specifically, we apply an upward force $\mathcal{F}$ on the robot base, which is largely set at the start of training. This force takes effect only when the robot's trunk achieves a near-vertical orientation, indicating a successful ground-sitting posture. The force magnitude decreases progressively as the robot can maintain a target height at the end of the episode. See more details in Appendix B.

D. Motion Constraints

1) Action Bound (Rescaler): Humanoid robots often feature many DoFs, each equipped with wide position limits and high-power actuators. This configuration often results in violent motions after RL training, characterized by violent ground hitting and rapid bouncing movements. While setting low action bounds could mitigate this behavior, it might prevent the robot from exploring effective standing-up motions. To this end, we introduce an action rescaler β to scale the action output, implicitly controlling the bound of the maximal torques on each actuator. This scale coefficient gradually decreases like vertical force reduction. See more details in Appendix B.

2) Smoothness Regularization: To prevent motion oscillation, we adopt the smoothness regularization method L2C2 [20] into our multi-critic formulation. This method applies regularization to both the actor-network π_θ and critics

TABLE III: Main simulation results. We present a performance comparison between HoST and baselines for the proposed metrics. The means and standard variation are reported across 5 evaluations, each with 250 testing episodes. '/' indicates that the method completely failed on a certain task.

Method	Ground				Platform				Wall				Slope			
	E_{succ} ↑	E_{feet} ↓	E_{smth} ↓	E_{engy} ↓	E_{succ} ↑	E_{feet} ↑	E_{smth} ↓	E_{engy} ↓	E_{succ} ↑	E_{feet} ↑	$E_{\text{smth}}.$ ↓	$E_{\text{engy}}.$ ↓	E_{succ} ↑	E_{smth} ↑	E_{smth} ↓	E_{engy} ↓
(a) Ablation on Number of Critics																
HoST-w/o-MuC	0.0 (±0.0)	/	/	/	0.0 (±0.0)	/	/	/	0.0 (±0.0)	/	/	/	0.0 (±0.0)	/	/	/
HoST	**99.5** (±0.4)	**1.52** (±.10)	**2.90** (±.21)	**1.35** (±.02)	**99.8** (±0.2)	**1.16** (±.04)	**3.39** (±.39)	**0.58** (±.01)	**94.2** (±1.2)	**1.14** (±.08)	**4.66** (±.69)	**1.08** (±.02)	**98.5** (±0.4)	**5.71** (±.24)	**5.31** (±.45)	**0.83** (±.01)
(b) Ablation on Exploration Strategy																
HoST-w/o-Force	0.0 (±0.0)	/	/	/	6.8 (±2.0)	0.12 (±.02)	3.39 (±.40)	1.98 (±.02)	0.0 (±0.0)	/	/	/	0.0 (±0.0)	/	/	/
HoST-w/o-Force-RND	19.8 (±1.2)	**0.87** (±.11)	3.13 (±.18)	2.55 (±.03)	99.5 (±0.4)	1.66 (±.11)	3.55 (±.37)	0.78 (±.01)	0.0 (±0.0)	/	/	/	0.0 (±0.0)	/	/	/
HoST	**99.5** (±0.4)	1.52 (±0.10)	**2.90** (±.21)	**1.35** (±.02)	**99.8** (±0.2)	**1.16** (±.04)	**3.39** (±.39)	**0.58** (±.01)	**94.2** (±1.2)	**1.14** (±.08)	**4.66** (±.69)	**1.08** (±.02)	**98.1** (±0.4)	**5.71** (±.24)	**5.44** (±.45)	**0.89** (±.01)
(c) Ablation on Motion Constraints																
HoST-w/o-Bound	98.8 (±0.6)	7.27 (±.42)	9.52 (±.25)	3.59 (±.02)	99.4 (±0.8)	6.23 (±.34)	11.65 (±.34)	1.76 (±.03)	**99.6** (±0.5)	5.48 (±.70)	8.80 (±.74)	1.73 (±.02)	82.4 (±4.4)	32.22 (±2.5)	16.44 (±.86)	2.62 (±.07)
HoST-Bound0.25	**99.8** (±0.4)	**1.16** (±.08)	**2.75** (±.19)	1.56 (±.01)	**99.8** (±0.1)	**0.68** (±.05)	**3.17** (±.41)	0.79 (±.02)	84.6 (±2.5)	**0.42** (±.02)	**4.23** (±.71)	1.44 (±.04)	98.0 (±1.4)	**2.74** (±.16)	**4.67** (±.42)	0.90 (±.02)
HoST-w/o-L2C2	92.3 (±0.7)	2.29 (±.06)	4.05 (±.21)	1.43 (±.01)	**99.8** (±0.0)	1.93 (±.07)	4.47 (±.42)	0.92 (±.02)	97.8 (±1.6)	1.43 (±.16)	5.29 (±.70)	1.55 (±.02)	98.8 (±0.8)	3.93 (±.24)	6.32 (±.46)	1.12 (±.02)
HoST-w/o-r^{style}	99.2 (±0.5)	1.36 (±.07)	2.83 (±.21)	1.67 (±.03)	82.2 (±3.5)	1.18 (±.08)	3.56 (±.40)	0.67 (±.03)	0.0 (±0.0)	/	/	/	21.4 (±3.2)	8.61 (±.12)	6.49 (±.54)	1.69 (±.05)
HoST	99.5 (±0.4)	1.52 (±.10)	2.90 (±.21)	**1.35** (±.02)	**99.8** (±0.2)	1.16 (±.04)	3.39 (±.39)	**0.58** (±.01)	94.2 (±1.2)	1.14 (±.08)	4.66 (±.69)	**1.08** (±.02)	**98.5** (±0.4)	5.71 (±.24)	5.31 (±.45)	**0.83** (±.01)
(d) Ablation on Historical States																
HoST-History0	98.1 (±1.4)	2.11 (±.14)	2.72 (±.22)	1.27 (±.02)	99.5 (±0.5)	1.53 (±.13)	3.29 (±.40)	**0.47** (±.01)	64.5 (±1.2)	1.66 (±.04)	4.74 (±.72)	1.66 (±.03)	97.4 (±2.0)	5.20 (±.24)	**4.97** (±.48)	**0.66** (±.02)
HoST-History2	99.3 (±0.3)	2.25 (±.13)	**2.56** (±.19)	**1.16** (±.01)	99.4 (±0.5)	**0.77** (±.39)	**3.27** (±.39)	0.60 (±.01)	93.7 (±1.4)	1.79 (±.08)	4.81 (±.71)	1.22 (±.01)	98.6 (±0.6)	5.06 (±.24)	5.35 (±.44)	0.77 (±.01)
HoST-History5 (ours)	**99.5** (±0.4)	**1.52** (±.10)	2.90 (±.21)	1.35 (±.02)	**99.8** (±0.2)	1.16 (±.04)	3.39 (±.39)	0.58 (±.01)	**94.2** (±1.2)	**1.14** (±.08)	4.66 (±.69)	**1.08** (±.02)	**98.6** (±0.4)	5.71 (±.24)	5.31 (±.45)	0.83 (±.01)
HoST-History10	98.8 (±0.8)	1.62 (±.08)	3.02 (±.20)	1.60 (±.02)	99.2 (±0.8)	0.78 (±.05)	3.55 (±.40)	0.71 (±.01)	88.2 (±2.6)	1.24 (±.06)	**4.61** (±.72)	1.46 (±.05)	**98.6** (±0.8)	**3.93** (±.26)	5.41 (±.49)	0.91 (±.01)

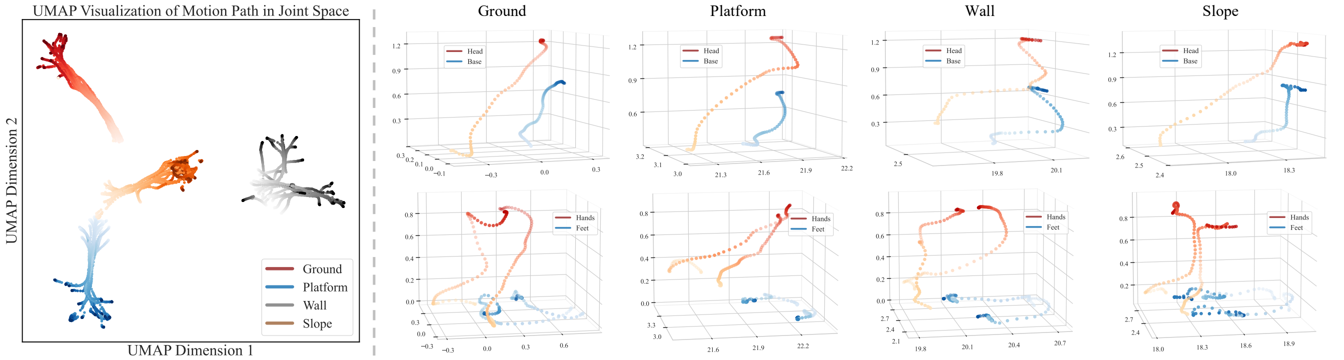


Fig. 4: Motion analysis in simulation. (Left) UMAP visualization of joint-space trajectories demonstrates similar but distinct motion patterns on the terrains except for the wall. Besides, the trajectories of each terrain are overall consistent, with variation to handle the difference of starting postures. (Right) 3D trajectory visualizations reveal stable, coordinated hand-foot motion and dynamic posture adaptability, demonstrating effective whole-body coordination and validating the proposed framework. Point color in the plot indicates motion progression, with lighter shades for earlier and darker for later times.

V_{ϕ_i} by introducing a bounded sampling distance between consecutive states s_t and s_{t+1}:

$$\mathcal{L}_{\text{L2C2}} = \lambda_\pi D(\pi_\theta(s_t), \pi_\theta(\bar{s}_t)) + \lambda_V \sum D(V_{\phi_i}(s_t), V_{\phi_i}(\bar{s}_t)),$$

where D is a distance metric, λ_π and λ_V are weight coefficient, $\bar{s}_t = s_t + (s_{t+1} - s_t) \cdot u$ is the interpolated state given a uniform noise $u \sim \mathcal{U}(\cdot)$. We combine this objective function with ordinary PPO objectives to train our control policies.

E. Training in Simulation & Sim-to-Real Transfer

We use Isaac Gym [31] simulator with 4096 parallel environments and the 23-DoF Unitree G1 robot to train standing-up control policies with the PPO [42] algorithm.

1) Terrain Design: To model the diverse starting postures in the real world, we design 4 terrains to diversify the starting postures: (1) **ground** that is flat, (2) **platform** that supports the trunk of robot, (3) **wall** that supports the trunk of the robot, and (4) **slope** with a benign inclination that supports the whole robot. We visualize these terrains and examples of their corresponding scenes in the real world in Fig. 3.

2) Domain Randomization: To enhance real-world deployment, we employ domain randomization [48] to bridge the physical gap between simulation and reality. The randomization parameters, detailed in Table II, include body mass, base center of mass (CoM) offset, PD gains, torque offset, and initial pose, following [2, 28]. Notably, the CoM offset is critical, as it enhances controller robustness against real-world CoM position noise, which may arise from insufficient torques or discrepancies between simulated and real robot models.

F. Implementation Details

Our implementation of PPO is based on [39]. The actor and critic networks are structured as 3-layer and 2-layer MLPs, respectively. Each episode has a rollout length of 500 steps. For smoothness regularization, the weight coefficients λ_π and λ_V are set to 1 and 0.1, respectively. The PD controller operates at 200 Hz in simulation and 500 Hz on the real robot to ensure accurate tracking of the PD targets, while the control policies run at 50 Hz. Additional implementation details and hardware setup are provided in Appendix A.

V. Simulation Experiments

A. Experimenrt Setup

1) Evaluation Metrics.: While the design of evaluation metrics for humanoid standing-up control remains an open question [44], we aim to make a step forward by proposing the following metrics:

- **Success rate** E_{succ}: The episode is considered successful if the robot's base height, h_{base}, exceeds a target height h_{targ} and is maintained for the remainder of the episode, indicating stable standing.

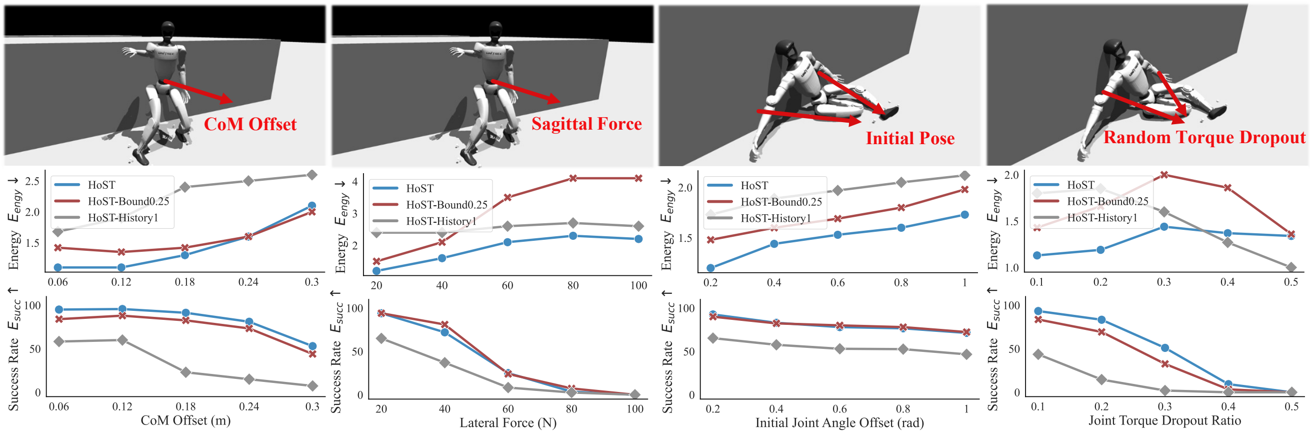


Fig. 5: Robustness analysis in simulation. Evaluation of control policies under four environmental disturbances demonstrates the robustness of our controllers. The poor performance of HoST-History1 indicates the importance of historical information for robustness, while HoST-Bound0.25's high energy consumption reveals limitations in motion quality under disturbance, demonstrating the effect of curriculum setup of action bound.

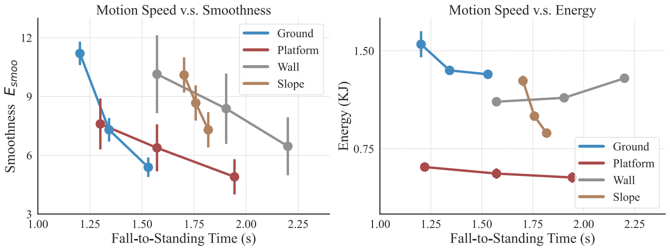


Fig. 6: Trade-off analysis in simulation. Trade-offs between motion speed, smoothness, and energy across terrains. Results show the inverse speed-smoothness relationship, indicating the importance of constrained motion speed achieved by our method for real-world deployment.

- **Feet movement** E_{feet}: The distance traveled by the robot's feet after reaching the target height h_{targ}, indicating stability in the standing pose.
- **Motion smoothness** E_{smth}: We aggregate the movement of all joint angles of consecutive control steps to measure the smoothness of the motion. It indicates that the robot should keep a smooth motion during the whole episode.
- **Energy** E_{engy}: The energy consumed before reaching h_{targ}, indicating the avoidance of violent standing-up motion.

2) Baselines: To evaluate the effectiveness of the key design choices in HoST, we compare it against the following ablated versions:

- **Single critic**: A baseline using a single critic RL to assess the impact of multiple critics on motor skill learning.
- **Exploration strategy**: Baselines with random noise and curiosity-based rewards (e.g., RND [1]) to evaluate the effectiveness of the force curriculum.
- **Motion constraints**: Ablation of action bounds β and smoothness regularization L2C2 to test their influence on motion smoothness.
- **Historical states**: Ablation of the number of historical states to assess their effect on standing-up motion.

B. Main Results

HoST demonstrates good efficacy in learning standing-up control across all terrains, as shown in Table III. The effect of key design choices is summarized as follows:

Multiple critics are crucial for learning motor skills Using the same reward functions, the performance of the single critic version of HoST deteriorates significantly across all terrains, achieving zero success rates. This highlights the importance of multiple critics in learning and integrating motor skills while also reducing the hyperparameter tuning burden.

Force curriculum enhances exploration efficiency. Without the proposed force curriculum, the robot fails to stand up on all terrains except the platform, as the other terrains require exploration from a fully fallen state to stable kneeling. While curiosity-based exploration partially alleviates this challenge, performance remains unsatisfactory. In contrast, the force curriculum greatly improves exploration efficiency.

Action bound prevents abrupt motions. While the robot can learn to stand up without action bounds (HoST-w/o-Bound), its movements are excessively violent, as indicated by three performance metrics. With action bounds, HoST demonstrates smoother motions and higher success rates. Although HoST-Bound0.25 performs well, its motions are less natural due to restricted exploration during training.

Smoothness regularization prevents motion oscillation. Adding smoothness constraints significantly reduces motion oscillation and increases energy efficiency, validating the effectiveness of smooth regularization. Further discussion is presented in Section VI.

Medium history length yields great performance. HoST with short history length underperforms in contact-rich scenarios, such as the Wall terrain. In contrast, a longer history length improves performance, though it slightly reduces motion smoothness and increases energy consumption compared to the default setting.

C. More Analyses

Trajectory analysis (Fig. 4). Following [12], we apply Uniform Manifold Approximation and Projection (UMAP; [32]) to project joint-space motion trajectories into 2D, providing a visualization of the humanoid robot's motion across diverse

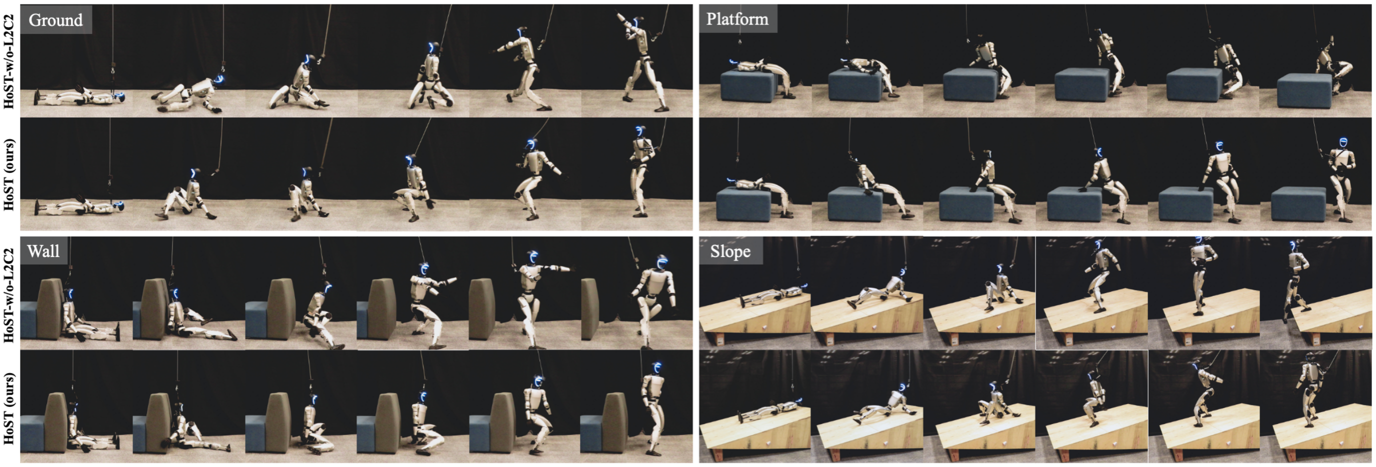


Fig. 7: Snapshot of real robot motion. We directly transfer our policies from simulation to four real-world scenes that correspond to four simulation terrains. We conclude that (1) our policies can produce smooth and successful standing-up motion in all tested scenes and (2) smooth regularization of L2C2 is important to avoid oscillation and improve stability.

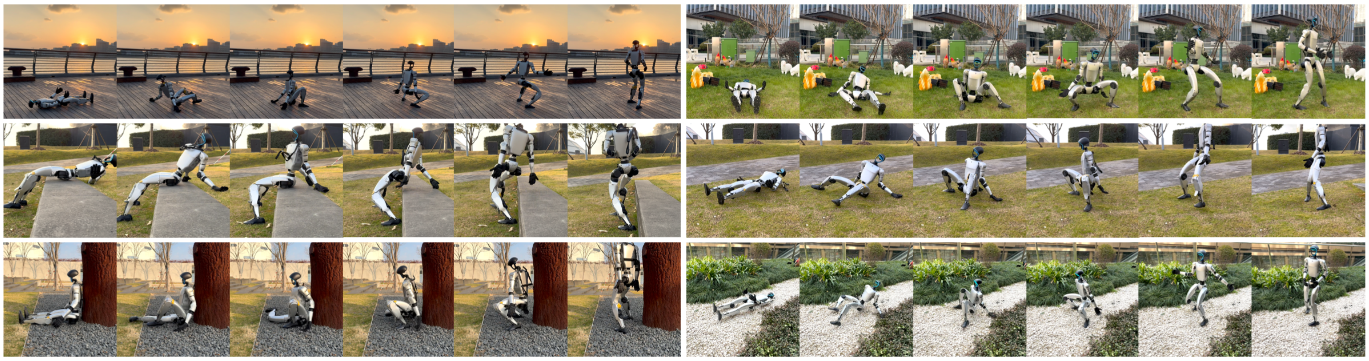


Fig. 8: Snapshot of outdoor experiments. We test our controllers in diverse outdoor environments, demonstrating smooth motion on unseen terrains such as grassland, wooden platforms, and stone roads, as well as successful performance on stone platforms and tree-leaning postures.

TABLE IV: Main results for real robot experiments. We report the success rate and motion smoothness to quantitatively compare our methods with the baseline. The results demonstrate the superiority of our method and the importance of adding smooth regularization into our method.

Method	Ground		Platform		Wall		Slope		Overall	
	$E_{\text{succ}} \uparrow$	$E_{\text{smth}} \downarrow$	$E_{\text{succ}} \uparrow$	$E_{\text{smth}} \downarrow$	$E_{\text{succ}} \uparrow$	$E_{\text{smth}} \downarrow$	$E_{\text{succ}} \uparrow$	$E_{\text{smth}} \downarrow$	$E_{\text{succ}} \uparrow$	$E_{\text{smth}} \downarrow$
HoST-w/o-L2C2	5/5	2.09	2/5	7.85	4/5	13.36	0/5	2.89	11/20	6.54
HoST (ours)	5/5	1.83	5/5	5.06	5/5	7.22	5/5	1.94	20/20	4.01

terrains. The resulting UMAP figure demonstrates distinct motion patterns: smooth, controlled movement on flat ground, while more complex, yet consistent, trajectories emerge on challenging terrains such as Wall. Additionally, in the 3D trajectory plots, the coordinated motion of the robot's hands and feet reveals significant posture adaptability, as the robot adjusts its stance dynamically for balance and stability. These observations highlight the harmonious whole-body coordination achieved by our controllers and validate the effectiveness of our proposed framework.

Robustness analysis (Fig. 5). We comprehensively evaluate the robustness of our learned control policies by simulating various environmental disturbances. Specifically, we test four types of external perturbations: CoM position offset in the sagittal direction, consistent sagittal force, initial joint angle offset, and random torque dropout ratio. Our results demonstrate that the policies exhibit remarkable robustness across all disturbances, achieving high success rates and efficient motion energy utilization. Notably, the poor performance of HoST-History1 underscores the critical role of historical information, which implicitly encodes contact dynamics, in maintaining robustness. Furthermore, while HoST-Bound0.25 achieves a high success rate, its elevated energy consumption highlights its limited ability to maintain motion smoothness under disturbance. These findings validate the robustness of our policies while indicating the importance of historical context and curriculum of action bound for robust standing-up.

Trade-off analysis (Fig. 6). We examine trade-offs between motion speed, smoothness, and energy consumption across terrains. On the left, motion speed and smoothness exhibit an inverse relationship: longer fall-to-standing times enhance smoothness but reduce speed, a trend consistent across all terrains. On the right, energy consumption increases with fall-to-standing time, with terrain-specific variations. For exam-

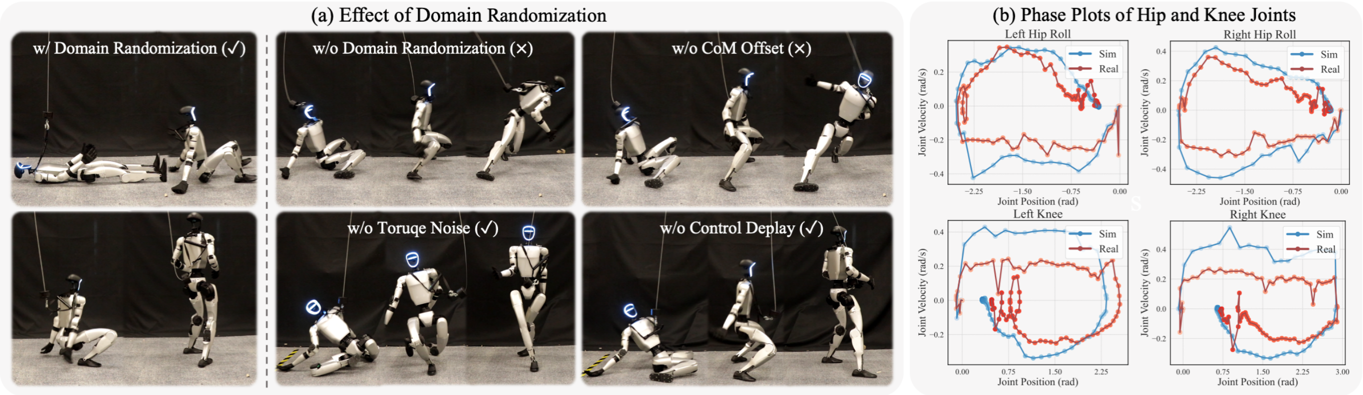


Fig. 9: Sim-to-real analysis. (a) We analyze the effect of each domain randomization term, showing that our randomization terms effectively mitigate the sim-to-real gap, with the CoM position being particularly influential. (b) To further investigate the sim-to-real gap, we compare the phases of knee and hip joints that are crucial for standing-up control. The results reveal significant discrepancies in joint velocities, suggesting a sim-to-real gap in joint torques.

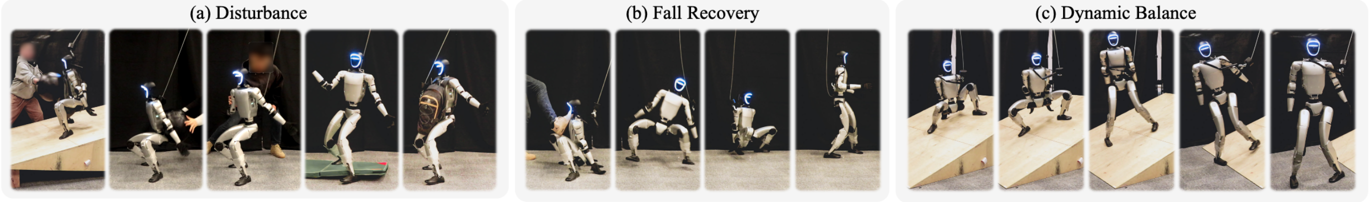


Fig. 10: Emergent properties in real robot experiments. (a) our controllers show great robustness to the external force (3kg ball), blocking objects on the ground, and payload mass up to 12kg (2x mass of trunk. (b) Our controllers also exhibit a surprising ability to recover from very large external forces without fully falling down. (c) Our policies also exhibit the ability of dynamic balancing over a 15° slippery slope without falling down.

TABLE V: Robustness to payload and random torque dropout.

Metric	Payload Mass					Torque Dropout Ratio			
	4kg	6kg	8kg	10kg	12kg	0.05	0.1	0.15	0.2
$E_{\mathrm{smth}} \downarrow$	1.75	1.92	1.86	1.82	1.85	2.00	2.16	2.61	/
$E_{\mathrm{succ}} \uparrow$	3/3	3/3	3/3	3/3	2/3	3/3	3/3	3/3	0/3

ple, the Slope terrain requires higher energy for balancing. Interestingly, the Wall terrain shows a distinct trend: energy consumption rises sharply at longer fall-to-standing times despite low motion speed, suggesting greater energy intensity. This is likely due to the need for increased force or modified body mechanics to push against a vertical surface, making the motion in Wall less energy-efficient than other terrains. Overall, the results reveal a clear inverse relationship between motion speed and smoothness, indicating the importance of constrained motion speed for real-world deployment and validating the necessity of our approach to achieve such motions.

VI. Real Robot Experiments

A. Main Results

We evaluate our method in both laboratory and outdoor environments corresponding to simulation terrains, using HoST-w/o-L2C2 as the baseline to examine the effect of smoothness regularization during deployment.

Smooth regularization improves motions (Fig. 7). Motion oscillations are observed in all scenes without smoothness regularization, often leading to standing-up failures. In contrast, our method produces smooth and stable motions, especially on 10.5° slope. Quantitative results in Table IV strengthen this conclusion, with our approach achieving a 100% success rate and high motion smoothness across all scenes.[1]

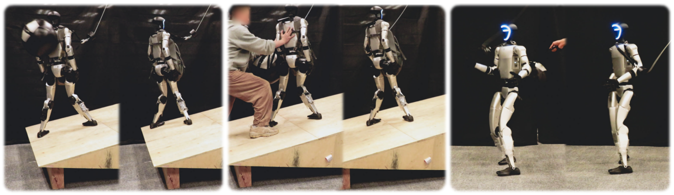


Fig. 11: Standing stability. Our control policies demonstrate great stability against external disturbances after successful standing up.

Generalization to outdoor environments (Fig. 8). We evaluate our learned controllers in a variety of outdoor environments, testing their ability to generalize to terrains not encountered during training. On flat ground, the controllers produce stable, smooth motions across grassland, wooden platforms, and stone roads. Notably, these terrains were not included in the training simulations. Additionally, our controllers successfully handle more complex scenarios, including stone platforms and tree-leaning postures, demonstrating their adaptability to diverse real-world conditions.

B. Sim-to-real Analysis

In this analysis, we investigate the effect of various domain randomization terms on the sim-to-real gap, as shown in Fig. 9. Our results demonstrate that the introduction of these randomization terms significantly reduces the sim-to-real gap, particularly with respect to the Center of Mass (CoM) position.

[1] We select the successful episode to compute smoothness to reflect the effect of L2C2 regularization better. Due to the unavailability of the height, we compute the smoothness E_{smth} within two seconds after starting up.

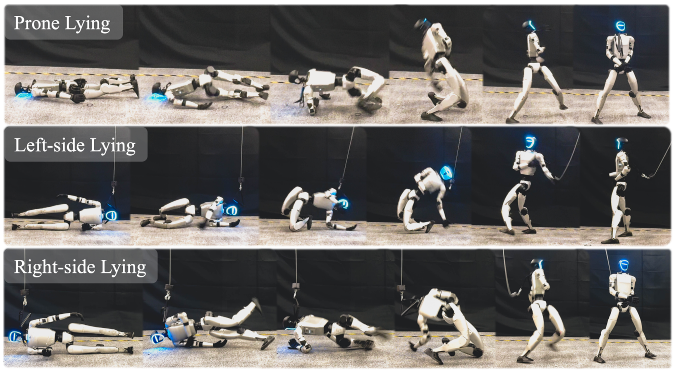


Fig. 12: More diverse postures. HoST can learn across prone postures on the ground. The learn policies can also handle side-lying postures.

Phase plot. To further investigate the sources of this gap, we examine the phase plots of the knee and hip roll joints. These joints are considered most important for standing-up motions. We observe a notable discrepancy between simulated and real-world joint velocities, suggesting a gap in joint torques. This highlights the need for more accurate actuator modeling to bridge the sim-to-real gap in humanoid standing-up tasks, which is also suggested by previous work on quadrupedal robots [15]. Despite this, our controllers remain effective in handling these discrepancies, exhibiting joint paths consistent with the simulated ones.

C. Emergent Properties

Robustness to external disturbance (Fig. 10a). The robustness of our control policies was tested through experiments involving external disturbances, such as a 3 kg ball impact and obstructive objects. The controllers maintained stability even under significant disturbances, like objects disrupting the robot's center of gravity. Additionally, the controllers managed payloads up to 12kg, twice the mass of the humanoid robot's trunk. We also quantitatively verify the great robustness of payload and torque dropout ratio in Table V.

Fall recovery (Fig. 10b). Our controllers also exhibited strong resilience in recovering from large external forces without fully falling down. This capability is vital for humanoid robots navigating unpredictable real-world scenarios with sudden impacts or balance shifts. Testing showed that, even under abrupt perturbations, the robots regained their upright posture, demonstrating the effectiveness of our control strategies in maintaining dynamic stability.

Dynamic balance (Fig. 10c). We further tested our controllers on a 15° slippery slope, simulating challenging real-world conditions such as unstable surfaces. The controllers not only maintained stability on the incline but also adjusted posture and center of mass in real time to counteract the slippery conditions. These results highlight the adaptability and stability of our controllers, ensuring humanoid robots can operate safely on diverse and unpredictable terrains.

Standing stability (Fig. 11). Our controllers demonstrate strong standing stability, effectively resisting external disturbances after successful standing up. This stability is beneficial for integrating our controllers into existing control systems.

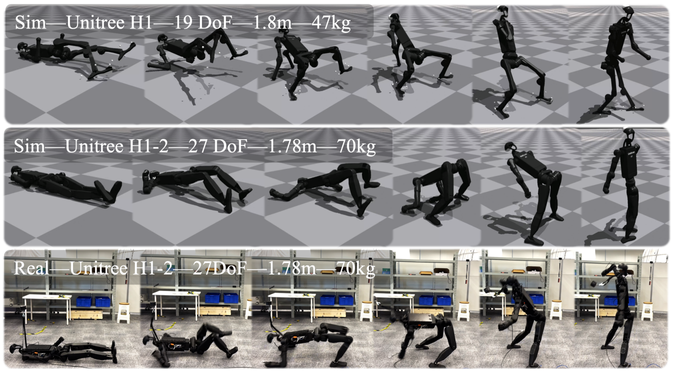

Sim—Unitree H1—19 DoF—1.8m—47kg
Sim—Unitree H1-2—27 DoF—1.78m—70kg
Real—Unitree H1-2—27DoF—1.78m—70kg

Fig. 13: Extension to large-size robots. HoST can be easily extended to Unitree H1 and H1-2 humanoid robots with minor hyperparameter tuning.

D. Prone and Side-lying Postures on the Ground

We demonstrate that HoST is capable of learning across prone postures on the ground, as visualized in Fig. 12. Besides, the learned policies can also handle side-lying postures without any tuning. However, there are significant differences in motion patterns between prone and supine postures. This somehow limits our method: when training from posture postures, harder constraints on hip joints are necessary to prevent violent motions, making the feasibility of joint training from prone and supine postures unclear currently.

E. Extend HoST to Large-size Humanoid Robots

We believe that standing-up control is more challenging in larger humanoid robots than in G1 due to their increased weight and the limited actuators. As an initial test, we extend HoST to Unitree H1 and H1-2. Simulation and real-world motions are visualized in Fig. 13. Compared to G1, we observe greater reliance on (i) upper-body contact with the ground and (ii) high hip actuation. While successful, two sim-to-real gaps emerge: (i) the need for high-stiffness joints to compensate for insufficient torques and, which is consistent with the observation in Section VI-B; (ii) noticeable deviations in upper-body posture. It remains unclear whether these gaps originate from our framework or hardware limitations. Identifying the source of these gaps is valuable in the future.

VII. Conclusion

Our proposed framework, HoST, advances humanoid standing-up control by addressing the limitations of existing methods, which either neglect hardware constraints or rely on predefined motion trajectories. By leveraging reinforcement learning from scratch, HoST enables the learning of posture-adaptive standing-up motions across diverse terrains, ensuring effective sim-to-real transfer. The multi-critic architecture, along with smoothness regularization and implicit speed constraints, optimizes the controllers for real-world deployment. Experimental results with the Unitree G1 humanoid robot demonstrate smooth, stable, and robust standing-up motions in a variety of real-world scenarios. Looking forward, this work paves the way for integrating standing-up control into existing humanoid systems, with the potential of expanding their real-world applicability.

VIII. Limitations and Future Directions

While our method demonstrates strong real-world performance, we acknowledge several key limitations that should be addressed in the near future.

Perception of the environment. Although proprioception alone is sufficient for many postures, some failures were observed during outdoor tests, such as standing from a seated position and colliding with surroundings. Integrating perceptual capabilities will help address this issue.

More diverse postures. We observe that training with both supine and prone postures has negatively impacted performance due to interference between sampled rollouts. Addressing this issue could further enhance capabilities like fall recovery and improve overall system generalization.

Integration with existing humanoid systems. Although this paper does not demonstrate integration with existing humanoid systems, we envision that standing-up control can be effectively incorporated into current humanoid frameworks to extend their real-world applications.

Acknowledgments

This work is funded in part by the National Key R&D Program of China (2022ZD0160201), and Shanghai Artificial Intelligence Laboratory.

References

[1] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. In *International Conference on Learning Representations (ICLR)*, 2019.

[2] Luigi Campanaro, Siddhant Gangapurwala, Wolfgang Merkt, and Ioannis Havoutis. Learning and deploying robust locomotion policies with minimal dynamics randomization. In *6th Annual Learning for Dynamics & Control Conference (L4DC)*, 2024.

[3] Juan Alejandro Castano, Chengxu Zhou, and Nikos Tsagarakis. Design a fall recovery strategy for a wheel-legged quadruped robot using stability feature space. In *International Conference on Robotics and Biomimetics (ROBIO)*, 2019.

[4] Xuxin Cheng, Yandong Ji, Junming Chen, Ruihan Yang, Ge Yang, and Xiaolong Wang. Expressive whole-body control for humanoid robots. In *Robotics Science and Systems (RSS)*, 2024.

[5] Xuxin Cheng, Jialong Li, Shiqi Yang, Ge Yang, and Xiaolong Wang. Open-television: Teleoperation with immersive active visual feedback. *arXiv preprint arXiv:2407.01512*, 2024.

[6] Laura J Claxton, Dawn K Melzer, Joong Hyun Ryu, and Jeffrey M Haddad. The control of posture in newly standing infants is task dependent. *Journal of Experimental Child Psychology*, 2012.

[7] Jeremy Dao, Helei Duan, and Alan Fern. Sim-to-real learning for humanoid box loco-manipulation. In *International Conference on Robotics and Automation (ICRA)*, 2024.

[8] Farbod Farshidian, Michael Neunert, Alexander W Winkler, Gonzalo Rey, and Jonas Buchli. An efficient optimal planning and control framework for quadrupedal locomotion. In *International Conference on Robotics and Automation (ICRA)*, 2017.

[9] Zipeng Fu, Qingqing Zhao, Qi Wu, Gordon Wetzstein, and Chelsea Finn. Humanplus: Humanoid shadowing and imitation from humans. In *Conference on Robot Learning (CoRL)*, 2024.

[10] Clément Gaspard, Marc Duclusaud, Grégoire Passault, Mélodie Daniel, and Olivier Ly. Frasa: An end-to-end reinforcement learning agent for fall recovery and stand up of humanoid robots. *arXiv preprint arXiv:2410.08655*, 2024.

[11] Ambarish Goswami and Vinutha Kallem. Rate of change of angular momentum and balance maintenance of biped robots. In *International Conference on Robotics and Automation (ICRA)*, 2004.

[12] Tuomas Haarnoja, Ben Moran, Guy Lever, Sandy H Huang, Dhruva Tirumala, Jan Humplik, Markus Wulfmeier, Saran Tunyasuvunakool, Noah Y Siegel, Roland Hafner, et al. Learning agile soccer skills for a bipedal robot with deep reinforcement learning. *Science Robotics*, 2024.

[13] Tairan He, Zhengyi Luo, Xialin He, Wenli Xiao, Chong Zhang, Weinan Zhang, Kris Kitani, Changliu Liu, and Guanya Shi. Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. In *Conference on Robot Learning (CoRL)*, 2024.

[14] Tairan He, Wenli Xiao, Toru Lin, Zhengyi Luo, Zhenjia Xu, Zhenyu Jiang, Jan Kautz, Changliu Liu, Guanya Shi, Xiaolong Wang, et al. Hover: Versatile neural whole-body controller for humanoid robots. In *International Conference on Robotics and Automation (ICRA)*, 2025.

[15] Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 2019.

[16] Zhenyu Jiang, Yuqi Xie, Jinhan Li, Ye Yuan, Yifeng Zhu, and Yuke Zhu. Harmon: Whole-body motion generation of humanoid robots from language descriptions. In *Conference on Robot Learning (CoRL)*, 2024.

[17] Fumio Kanehiro, Kenji Kaneko, Kiyoshi Fujiwara, Kensuke Harada, Shuuji Kajita, Kazuhito Yokoi, Hirohisa Hirukawa, Kazuhiko Akachi, and Takakatsu Isozumi. The first humanoid robot that has the same size as a human and that can lie down and get up. In *International Conference on Robotics and Automation (ICRA)*, 2003.

[18] Fumio Kanehiro, Kiyoshi Fujiwara, Hirohisa Hirukawa, Shin'ichiro Nakaoka, and Mitsuharu Morisawa. Getting up motion planning using mahalanobis distance. In *International Conference on Robotics and Automation (ICRA)*, 2007.

[19] Dohyeong Kim, Hyeokjin Kwon, Junseok Kim, Gunmin Lee, and Songhwai Oh. Stage-wise reward shaping for acrobatic robots: A constrained multi-

objective reinforcement learning approach. *arXiv preprint arXiv:2409.15755*, 2024.

[20] Taisuke Kobayashi. L2c2: Locally lipschitz continuous constraint towards stable and smooth reinforcement learning. In *International Conference on Intelligent Robots and Systems (IROS)*, 2022.

[21] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. In *Robotics: Science and Systems (RSS)*, 2021.

[22] Yasuo Kuniyoshi, Yoshiyuki Ohmura, Koji Terada, and Akihiko Nagakubo. Dynamic roll-and-rise motion by an adult-size humanoid robot. *International Journal of Humanoid Robotics*, 2004.

[23] Joonho Lee, Jemin Hwangbo, and Marco Hutter. Robust recovery controller for a quadrupedal robot using deep reinforcement learning. *arXiv preprint arXiv:1901.07517*, 2019.

[24] Jinhan Li, Yifeng Zhu, Yuqi Xie, Zhenyu Jiang, Mingyo Seo, Georgios Pavlakos, and Yuke Zhu. Okami: Teaching humanoid robots manipulation skills through single video imitation. In *Conference on Robot Learning (CoRL)*, 2024.

[25] Zhongyu Li, Xue Bin Peng, Pieter Abbeel, Sergey Levine, Glen Berseth, and Koushil Sreenath. Robust and versatile bipedal jumping control through reinforcement learning. In *Robotics Science and Systems (RSS)*, 2023.

[26] Zhongyu Li, Xue Bin Peng, Pieter Abbeel, Sergey Levine, Glen Berseth, and Koushil Sreenath. Reinforcement learning for versatile, dynamic, and robust bipedal locomotion control. *The International Journal of Robotics Research (IJRR)*, 2024.

[27] Fukang Liu, Zhaoyuan Gu, Yilin Cai, Ziyi Zhou, Shijie Zhao, Hyunyoung Jung, Sehoon Ha, Yue Chen, Danfei Xu, and Ye Zhao. Opt2skill: Imitating dynamically-feasible whole-body trajectories for versatile humanoid loco-manipulation. *arXiv preprint arXiv:2409.20514*, 2024.

[28] Junfeng Long, Junli Ren, Moji Shi, Zirui Wang, Tao Huang, Ping Luo, and Jiangmiao Pang. Learning humanoid locomotion with perceptive internal model. *arXiv preprint arXiv:2411.14386*, 2024.

[29] Dingsheng Luo, Yaoxiang Ding, Zidong Cao, and Xihong Wu. A multi-stage approach for efficiently learning humanoid robot stand-up behavior. In *International Conference on Mechatronics and Automation*, 2014.

[30] Yuntao Ma, Farbod Farshidian, and Marco Hutter. Learning arm-assisted fall damage reduction and recovery for legged mobile manipulators. In *International Conference on Robotics and Automation (ICRA)*, 2023.

[31] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu-based physics simulation for robot learning. *arXiv preprint arXiv:2108.10470*, 2021.

[32] Leland McInnes, John Healy, and James Melville. Umap: Uniform manifold approximation and projection for dimension reduction. *arXiv preprint arXiv:1802.03426*, 2018.

[33] Siddharth Mysore, George Cheng, Yunqi Zhao, Kate Saenko, and Meng Wu. Multi-critic actor learning: Teaching rl policies to act with style. In *International Conference on Learning Representations (ICLR)*, 2022.

[34] Michael Neunert, Farbod Farshidian, Alexander W Winkler, and Jonas Buchli. Trajectory optimization through contacts and automatic gait discovery for quadrupeds. *Robotics and Automation Letters (RA-L)*, 2017.

[35] Xue Bin Peng, Pieter Abbeel, Sergey Levine, and Michiel Van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *Transactions On Graphics (TOG)*, 2018.

[36] Xue Bin Peng, Yunrong Guo, Lina Halper, Sergey Levine, and Sanja Fidler. Ase. *Transactions on Graphics (TOG)*, 2022.

[37] Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[38] Ilija Radosavovic, Tete Xiao, Bike Zhang, Trevor Darrell, Jitendra Malik, and Koushil Sreenath. Real-world humanoid locomotion with reinforcement learning. *Science Robotics*, 2024.

[39] N. Rudin, David Hoeller, Philipp Reist, and Marco Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2024.

[40] Uluc Saranli, Alfred A Rizzi, and Daniel E Koditschek. Model-based dynamic self-righting maneuvers for a hexapedal robot. *The International Journal of Robotics Research (IJRR)*, 2004.

[41] John Schulman, Philipp Moritz, Sergey Levine, Michael I. Jordan, and P. Abbeel. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438*, 2015.

[42] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[43] Jörg Stückler, Johannes Schwenk, and Sven Behnke. Getting back on two feet: Reliable standing-up routines for a humanoid robot. In *IAS*, 2006.

[44] Rajesh Subburaman, Dimitrios Kanoulas, Nikos Tsagarakis, and Jinoh Lee. A survey on control of humanoid fall over. *Robotics and Autonomous Systems*, 166:104443, 2023.

[45] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

[46] Tianxin Tao, Matthew Wilson, Ruiyu Gou, and Michiel Van De Panne. Learning to get up. In *SIGGRAPH Conference Proceedings*, 2022.

[47] Yuval Tassa, Yotam Doron, Alistair Muldal, Tom Erez, Yazhe Li, Diego de Las Casas, David Budden, Abbas Abdolmaleki, Josh Merel, Andrew Lefrancq, et al. Deepmind control suite. *arXiv preprint arXiv:1801.00690*,

2018.

[48] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter 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)*, 2017.

[49] Claes Von Hofsten. Eye–hand coordination in the newborn. *Developmental psychology*, 1982.

[50] Jin Wang, Rui Dai, Weijie Wang, Luca Rossini, Francesco Ruscelli, and Nikos Tsagarakis. Hypermotion: Learning hybrid behavior planning for autonomous loco-manipulation. In *Conference on Robot Learning (CoRL)*, 2024.

[51] Pei Xu, Xiumin Shang, Victor Zordan, and Ioannis Karamouzas. Composite motion learning with task control. *Transactions on Graphics (TOG)*, 2023.

[52] Chuanyu Yang, Can Pu, Guiyang Xin, Jie Zhang, and Zhibin Li. Learning complex motor skills for legged robot fall recovery. *Robotics and Automation Letters (RA-L)*, 2023.

[53] Fatemeh Zargarbashi, Jin Cheng, Dongho Kang, Robert Sumner, and Stelian Coros. Robotkeyframing: Learning locomotion with high-level objectives via mixture of dense and sparse rewards. In *Conference on Robot Learning (CoRL)*, 2024.

[54] Chong Zhang, Wenli Xiao, Tairan He, and Guanya Shi. Wococo: Learning whole-body humanoid control with sequential contacts. In *Conference on Robot Learning (CoRL)*, 2024.

[55] Ziwen Zhuang, Shenzhe Yao, and Hang Zhao. Humanoid parkour learning. In *Conference on Robot Learning (CoRL)*, 2024.

Appendix

A. More Experimental Details

Hardware Setup. We conducted our experiments using the Unitree G1 humanoid robot, which has a mass of 35 kg, a height of 1.32 m, and 23 actuated degrees of freedom (6 per leg, 5 per arm, and 1 in the waist). The robot is equipped with a Jetson Orin NX for onboard computation and uses an IMU and joint encoders to provide proprioceptive feedback.

Evaluation Protocol. Each policy is evaluated on each terrain with 5 repetitions of 250 episodes each, totaling 1250 episodes. We report the mean and standard deviation of performance. The target standing-up height is set to 0.6m for the slope terrain and 0.7m for all other terrains during evaluation.

Robustness Test. The CoM bias and sagittal force are set on the x-axis direction of the robot. The initial joint angle offset is applied to all joints of the robot. The random torque dropout is applied to each simulation step (200Hz), where the torques are set to zero if being dropout.

Expression of metrics: Smoothness E_{smth} is computed via $\sum_{t=0}^{T-2} \|p_{t+2} - 2p_{t+1} + p_t\|^2$, where p_t is the joint positions. Energy E_{engy} is computed via $\sum_{t=0}^{h_{\text{base}}<H_{\text{stage2}}} |\boldsymbol{\tau}_t| \cdot |\dot{p}_t|^T \mathrm{d}t$ approximately, where $\boldsymbol{\tau}_t$ is joint torques and $\dot{p}_t$ is joint velocities, $\mathrm{d}t$ (0.02s) is the time of a single policy step.

B. More Implementation Details

Curriculum Setup. The curriculum adjustment condition is consistent for both the vertical force and action bound: the head height h_{head} must reach a target height H_{head} by the end of each episode. Initially, the vertical force $\mathcal{F}$ is set to 200N, and the action bound β is set to 1. Upon reaching the target head height, the vertical force decreases by 20N, and the action bound decreases by 0.02. The lower bounds for the vertical force and action bound are 0N and 0.25, respectively.

Stage Division. The first stage involves righting the body, where we set H_{stage1} to 0.45m. The second stage involves rising the body, with H_{stage2} set to 0.65m.

Reward Functions. We present the complete set of reward functions and their detailed descriptions in Table VI. Several regularization reward terms are adapted from prior work [21, 28, 13]. Additionally, we incorporate a tolerance reward, $f_{\text{tol}}(i, b, m, v)$, as defined in [47, 46]. This reward is computed as a function of an input value i, which is constrained by three parameters: bounds b, margin m, and value v. The bounds b define the region where the reward is 1 if i lies within the bounds. Outside this region, the reward smoothly decreases according to a Gaussian function, reaching the value v at a distance determined by the margin m.

Ankle parallel reward is calculated as the variance of keypoints' height of the ankle, as visualized in the right figure. These keypoints are handcrafted without collision models.

PPO Implementation. Our PPO implementation follows the framework outlined in [39]. The actor network consists of a 3-layer MLP with hidden dimensions [512, 256, 128], while each critic network is a 2-layer MLP with hidden dimensions [512, 256]. Each iteration includes 50 steps per environment, with 5 learning epochs and 4 mini-batches per epoch. The discount factor γ is set to 0.99, the clip ratio is set to 0.2, and the entropy coefficient is 0.01. The multi-critic architecture is based on previous work [33], where each advantage function is independently calculated and normalized within its corresponding reward group.

Baseline Implementations. HoST-w/o-MuC represents a baseline with a single value network, essentially a standard RL implementation. HoST-w/o-Force-RND removes the vertical force curriculum and introduces an RND reward with a coefficient of 0.2 [1]. HoST-Bound0.25 uses a fixed action bound of $\beta = 0.25$ without a curriculum. HoST-w/p-r^{style} eliminates all style-related reward functions. Lastly, HoST-History modifies the history length of states while keeping other implementations unchanged.

Terrains. The heights of the platforms range from 20cm to 92cm. The slope inclination varies from approximately 1° to 14°. The wall inclination spans from approximately 14° to 84°.

PD Controller. In simulation, the stiffness values are set as 100 for the upper body, 40 for the ankle, 150 for the hip, and

TABLE VI: Reward functions and groups used for learning standing-up control. Reward functions within the same group are independently normalized, whose associated advantaged functions are estimated via a distinct critic. The bold symbols represent vectors. The H with subscripts represents the threshold height of standing-up stages defined in Section IV-B. The f_{tol} is a gaussian-style function with a saturation bound, referring to [47, 46] for more details. 'G' denotes ground, and the letters in 'PSW' denote platform, slope, and wall, respectively.

Term	Expression	Weight	Description
(a) Task Reward	r^{task}	$w^{\text{task}} = 2.5$	It specifies the high-level task objectives.
Head height	$f_{\text{tol}}\left(h_{\text{head}}, [1, \text{inf}], 1, 0.1\right)$	1	The head of robot head h_{head} in the world frame.
Base orientation	$f_{\text{tol}}\left(-\theta^{\text{z}}_{\text{base}}, [0.99, \text{inf}], 1, 0.05\right)$	1	The orientation of the robot base represented by projected gravity vector.
(b) Style Reward	r^{style}	$w^{\text{style}} = 1$	It specifies the style of standing-up motion.
Waist yaw deviation	$\mathbb{1}(\lvert q_{\text{waist}}\rvert > 1.4)$	-10	It penalizes the large joint angle of the waist yaw.
Hip roll/yaw deviation	$\mathbb{1}(\max(\lvert \boldsymbol{q}^{\text{l,r}}_{\text{hip}}\rvert) > 1.4) \mid \mathbb{1}(\min(\lvert \boldsymbol{q}^{\text{l,r}}_{\text{hip}}\rvert) > 0.9)$	$-10/-10$	It penalizes the large joint angle of hip roll/yaw joints.
Knee deviation	$\mathbb{1}(\max(\boldsymbol{q}^{\text{l,r}}_{\text{knee}}) > 2.85) \mid \mathbb{1}(\min(\boldsymbol{q}^{\text{l,r}}_{\text{knee}}) < -0.06)$	$\frac{-0.25(G)}{-10(PSW)}$	It penalizes the large joint angle of knee joints.
Shoulder roll deviation	$\mathbb{1}(q^{l}_{\text{shoulder}} < -0.02) \mid \mathbb{1}(q^{r}_{\text{shoulder}} > 0.02)$	-2.5	It penalizes the large joint angle of shoulder roll joint.
Foot displacement	$\exp\left(-2 \times \Vert \boldsymbol{q}^{\text{xy}}_{\text{base}} - \boldsymbol{q}^{\text{xy}}_{\text{foot}}\Vert^{2}.\text{clip}(0.3, \text{inf})\right) \times \mathbb{1}(h_{\text{base}} > H_{\text{stage2}})$	2.5/2.5	It encourages robot CoM locates in support polygon, inspired by [11].
Ankle parallel	$(\text{var}(\boldsymbol{q}^{z}_{\text{left ankle}}) + \text{var}(\boldsymbol{q}^{z}_{\text{right ankle}}))/2 < 0.05$	20	It encourages the ankles to be parallel to the ground via ankle keypoints.
Foot distance	$\Vert \boldsymbol{q}^{l}_{\text{feet}} - \boldsymbol{q}^{r}_{\text{feet}}\Vert^{2} > 0.9$	-10	It penalizes a far distance between feet.
Feet stumble	$\mathbb{1}(\exists i, \lvert \mathbf{F}^{\text{xy}}_{i}\rvert > 3\lvert F^{\text{z}}_{i}\rvert)$	$\frac{0(G)}{-25(PSW)}$	It penalizes a horizontal contact force with the environment.
Shank orientation	$f_{\text{tol}}(\text{mean}(\boldsymbol{\theta}^{\text{l,r}}_{\text{shank}}[2]), [0.8, \text{inf}], 1, 0.1) \times \mathbb{1}(h_{\text{base}} > H_{\text{stage1}})$	10	It encourages the left/right shank to be perpendicular to the ground.
Base angular velocity	$\exp(-2 \times \Vert \boldsymbol{\omega}^{\text{xy}}_{\text{base}}\Vert^{2}) \times \mathbb{1}(h_{\text{base}} > H_{\text{stage1}})$	1	It encourages low angular velocity of the during rising up.
(c) Regularization Reward	r^{regu}	$w^{\text{regu}} = 0.1$	It specifies the regulariztaion on standing-up motion.
Joint acceleration	$\Vert \ddot{p}\Vert^{2}$	$-2.5e^{-7}$	It penalizes the high joint accelrations.
Action rate	$\Vert a_{t} - a_{t-1}\Vert^{2}$	$-1e^{-2}$	It penalizes the high changing speed of action.
Smoothness	$\Vert a_{t} - 2a_{t-1} + a_{t-2}\Vert^{2}$	$-1e^{-2}$	It penalizes the discrepancy between consecutive actions.
Torques	$\Vert \boldsymbol{\tau}\Vert^{2}$	$-2.5e^{-6}$	It penalizes the high joint torques.
Joint power	$\lvert \boldsymbol{\tau}\rvert \lvert \dot{p}\rvert^{T}$	$-2.5e^{-5}$	It penalizes the high joint power
Joint velocity	$\Vert \dot{p}\Vert^{2}_{2}$	$-1e^{-4}$	It penalizes the high joint velocity.
Joint tracking error	$\Vert p_{t} - p^{\text{target}}_{t}\Vert^{2}$	$-2.5e^{-1}$	It penalizes the error between PD target (Eq. (1)) and actual joint position.
Joint position limits	$\sum_{i}[(p_{i} - p^{\text{Lower}}_{i}).\text{clip}(-\text{inf}, 0) + (p_{i} - p^{\text{Higher}}_{i}).\text{clip}(0, \text{inf})]$	$-1e^{2}$	It penalizes the joint position that beyond limits.
Joint velocity limits	$\sum_{i}[(\lvert \dot{p}_{i}\rvert - \dot{p}^{\text{Limit}}_{i}).\text{clip}(0, \text{inf})]$	-1	It penalizes the joint velocity that beyond limits.
(d) Post-task Reward	r^{post}	$w^{\text{post}} = 1$	It specifies the desired behaviors after a successful standing up.
Base angular velocity	$\exp(-2 \times \Vert \boldsymbol{\omega}^{\text{xy}}_{\text{base}}\Vert^{2}) \times \mathbb{1}(h_{\text{base}} > H_{\text{stage2}})$	10	It encourages low angular velocity of robot base after standing up.
Base linear velocity	$\exp(-5 \times \Vert \boldsymbol{v}^{\text{xy}}_{\text{base}}\Vert^{2}) \times \mathbb{1}(h_{\text{base}} > H_{\text{stage2}})$	10	It encourages low linear velocity of robot base after standing up.
Base orientation	$\exp(-5 \times \Vert \boldsymbol{\theta}^{\text{xy}}_{\text{base}}\Vert^{2} \times \mathbb{1}(h_{\text{base}} > H_{\text{stage2}})$	10	It encourages the robot base to be perpendicular to the ground.
Base height	$\exp(-20 \times \Vert h_{\text{base}} - h^{\text{target}}_{\text{base}}\Vert^{2} \times \mathbb{1}(h_{\text{base}} > H_{\text{stage2}})$	10	It encourages the robot base to reach a target height.
Upper-body posture	$\exp(-0.1 \times \Vert p_{\text{upper}} - p^{\text{target}}_{\text{upper}}\Vert^{2}) \times \mathbb{1}(h_{\text{base}} > H_{\text{stage2}})$	10	It encourages the robot to track a target upper body postures.
Feet parallel	$\exp(-20 \times \lvert h^{l}_{\text{feet}} - h^{r}_{\text{feet}}\rvert.\text{clip}(0.02, \text{inf})) \times \mathbb{1}(h_{\text{base}} > H_{\text{stage2}})$	2.5	In encourages the feet to be parallel to each other.

200 for the knee. The damping values are set to 4 for the upper body, 2 for the ankle, 4 for the hip, and 6 for the knee. High stiffness values for the hip and knee are used due to the high torque demands during the standing-up process. During real-world deployment, we observe a significant torque gap between simulation and reality (see Fig. 9). Thus, the stiffness of the hip and knee are adjusted to 200 and 275, respectively.

Joint	G1		H1		H1-2	
	Kp	Kd	Kp	Kd	Kp	Kd
Hip	150	4	350	4	350	4
Knee	200	6	350	4	350	4
Ankle	40	2	120	2	120	2
Shoulder	100	4	350	4	350	4
Elbow	100	4	350	4	350	4
Waist	100	4	200	4	200	4

Observation noises are without curriculum, set as below:

Observation	Ang. Velocity	Pitch & Roll	DoF Position	DoF Velocity	Action Rescaler
Noise Scale	$\mathcal{U}(-0.2, 0.2)$	$\mathcal{U}(-0.05, 0.05)$	$\mathcal{U}(-0.01, 0.01)$	$\mathcal{U}(-1.5, 1.5)$	$\mathcal{U}(-0.025, 0.025)$

Learning across prone postures. We make the following adjustment to work the algorithm: more strict constraints on hip joint deviation rewards, weights for reward groups, and additional thigh orientation reward functions as a replacement for shank orientation rewards.

Extending HoST to Unitree H1 and H1-2. We make the following adjustment to work the algorithm: scale of pulling force, height for curriculum, height for stage division, target postures, PD controllers, observation and action spaces. In our implementations, H1 has 19 actuators and H1-2 has 27 actuators. During the hardware deployment, the stiffness of hip and knee joints are amplified to 1.5 times than the simulation ones, similar to G1. We present more instructions in our code repository. Please refer to there for more details.