

FALCON: Learning Force-Adaptive Humanoid Loco-Manipulation

Yuanhang Zhang¹, Yifu Yuan¹, Prajwal Gurunath¹, Tairan He¹, Shayegan Omidshafiei², Ali-akbar Agha-mohammadi², Marcell Vazquez-Chanlatte³, Liam Pedersen³, Guanya Shi¹

¹Carnegie Mellon University ²Field AI ³Nissan USA

Page: <https://lecar-lab.github.io/falcon-humanoid>

Code: <https://github.com/LeCAR-Lab/FALCON>

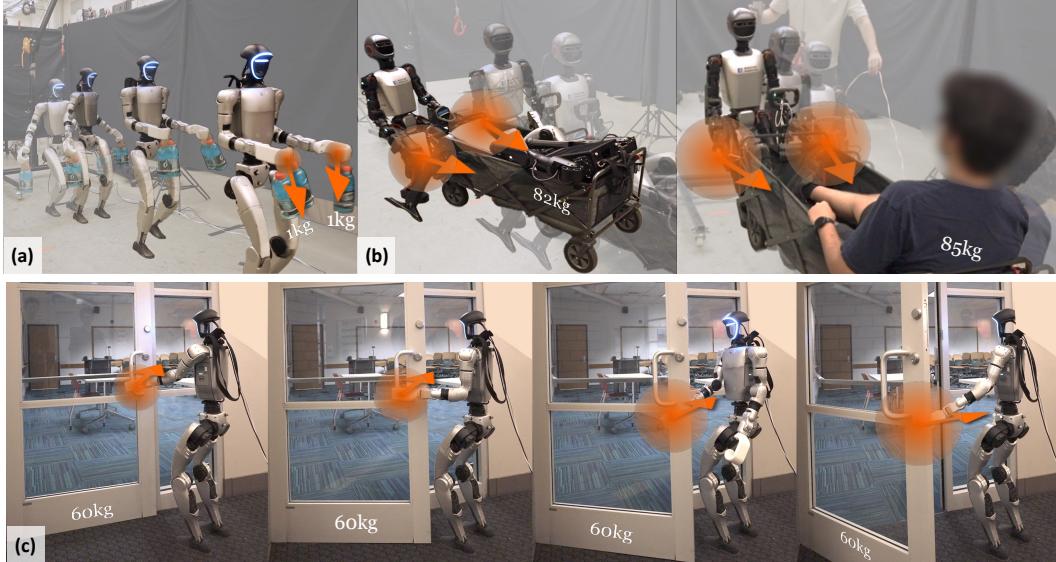


Figure 1: FALCON enables versatile forceful loco-manipulation tasks for humanoids: (a) **Transporting Payloads**: walk, squat, twist torso with payloads; (b) **Cart-Pulling** with significant longitudinal forces; (c) **Door-Opening** using both arms with multi-directional forces. Videos: <https://lecar-lab.github.io/falcon-humanoid>

Abstract: Humanoid loco-manipulation holds transformative potential for daily service and industrial tasks, yet achieving precise, robust whole-body control with 3D end-effector force interaction remains a major challenge. Prior approaches are often limited to lightweight tasks or quadrupedal/wheeled platforms. To overcome these limitations, we propose FALCON, a dual-agent reinforcement-learning-based framework for robust force-adaptive humanoid loco-manipulation. FALCON decomposes whole-body control into two specialized agents: (1) a lower-body agent ensuring stable locomotion under external force disturbances, and (2) an upper-body agent precisely tracking end-effector positions with implicit adaptive force compensation. These two agents are jointly trained in simulation with a force curriculum that progressively escalates the magnitude of external force exerted on the end effector while respecting torque limits. Experiments demonstrate that, compared to the baselines, FALCON achieves 2× more accurate upper-body joint tracking, while maintaining robust locomotion under force disturbances and achieving faster training convergence. Moreover, FALCON enables policy training without embodiment-specific reward or curriculum tuning. Using the same training setup, we obtain policies that are deployed across multiple humanoids, enabling forceful loco-manipulation tasks such as transporting payloads (0-20N force), cart-pulling (0-100N), and door-opening (0-40N) in the real world.

Keywords: Humanoid Loco-Manipulation, Force Adaptation, RL

1 Introduction

Humanoid robots have demonstrated remarkable progress in locomotion and manipulation [1–6]. However, extending these capabilities to forceful loco-manipulation remains fundamentally challenging. Tasks such as door opening, highlighted in the 2015 DARPA Challenge [7], require not only precise manipulation under dynamic, multi-directional forces but also maintaining lower-body stability throughout the interaction. Meeting these demands calls for humanoid systems that can flexibly adapt to varying payloads and contact forces without compromising overall precision and robustness in loco-manipulation.

Reinforcement Learning (RL) has achieved impressive results for humanoid whole-body control [8–27], yet existing RL approaches succeed mostly on lightweight tasks but do not consider significant interaction force during loco-manipulation tasks. Currently, there are two main paradigms: (1) *Lower-RL-Upper-IK*, which applies RL to lower-body locomotion while using kinematic solvers for upper-body control [8, 9], lacks whole-body dynamics modeling for forceful interaction and has limited whole-body coordination; (2) *Monolithic-Whole-body-RL*, which directly learns to control all degrees of freedom [10, 11], suffers from inefficient exploration as a single policy must simultaneously learn weakly correlated locomotion and manipulation skills. Although some advances have been made in force adaptation for quadrupeds [12–14, 28, 29], humanoids pose extra challenges like instability, higher complexity, and stricter torque limits, especially in certain joint configurations.

In this work, we aim to develop an RL framework that enables humanoid robots to perform a diverse set of force-adaptive loco-manipulation tasks. To this end, we introduce FALCON, a dual-agent RL architecture trained with a carefully designed 3D force curriculum respecting joint torque limits. Our key innovations include: (1) A *dual-agent learning decomposition* that separates lower-body and upper-body policy training with tailored rewards while sharing the same whole-body proprioception and commands; (2) A *3D force curriculum* with joint torque feasibility that progressively scales applied 3D forces on both end-effectors while enforcing joint torque constraints through inverse dynamics. FALCON enables efficient joint training of both stable locomotion and accurate EE tracking in forceful loco-manipulation tasks. We validate FALCON on Unitree G1 and Booster T1 humanoids, demonstrating its generalization across different platforms through: (1) Transporting Payloads, (2) Cart-Pulling, and (3) Door-Opening (Figure. 1), which require real-time adaptation to significant unknown 3D interaction force. In summary, our main contributions are:

- We introduce FALCON, a dual-agent reinforcement learning framework that enables humanoids to perform forceful loco-manipulation while adapting to substantial, unknown end-effector forces (0–100N, up to 30% of body weight). FALCON improves the upper-body joint tracking accuracy over prior methods by 100% while maintaining robust locomotion performance.
- To facilitate the efficient RL training, we design a 3D force curriculum with progressive force application while ensuring joint torque feasibility and maximizing its force-adaptive capability.
- We validate FALCON on two different humanoid platforms (Unitree G1, Booster T1), achieving strong cross-platform generalization with minimal tuning overhead.

2 Related Works

2.1 Humanoid Loco-Manipulation

Humanoid loco-manipulation remains a challenging control problem in robotics. While traditional model-based methods (e.g., simplified dynamics models and MPC) [20, 13, 21–25, 30] offer real-time planning, their reliance on manual design limits flexibility and generalizability. In contrast, learning-based methods—particularly sim-to-real RL—have demonstrated promising results in versatile loco-manipulation tasks [31, 17, 11, 8, 9, 32, 33]. For humanoids, two primary paradigms have emerged: *Lower-RL-Upper-IK* and *Monolithic-Whole-body-RL*. For *Lower-RL-Upper-IK*, Lu et al. [8] introduce PMP, which uses inverse kinematics (IK) and PD control for upper body control while

locomotion is trained and conditioned on a Conditional Variational Autoencoder (CVAE) representing upper-body motions. Then, Ben et al. [9] propose HOMIE that follows the same decoupling framework but introduces an exoskeleton-based cockpit for more intuitive human teleoperation. For *Monolithic-Whole-body-RL*, Dao et al. [32] adopted a unified RL approach for box pick-and-place tasks, training distinct skills (e.g., lifting, walking, stance) and orchestrating them via a finite state machine. He et al. [16, 17] and Ji et al. [10] employ a teacher-student training framework to mimic human motions for loco-manipulation tasks.

Despite these advances, few RL methods address significant unknown force disturbances on the EEs for humanoid loco-manipulation, and both paradigms exhibit critical shortcomings accordingly. *Lower-RL-Upper-IK* approaches suffer from delayed force compensation for upper-body control. *Monolithic-Whole-body-RL* methods face sample inefficiency from coarsely related task objectives between upper-body manipulation and lower-body locomotion, often leading to overfitting and the behavioral dominance of either upper or lower body. In this study, inspired by [34–36], we propose FALCON, a dual-agent RL framework employing task-specific reward formulations for upper-lower body decomposition. Unlike separately trained architectures, the two agents in FALCON are jointly trained with shared proprioception and commands, allowing mutual awareness of each other’s behaviors. This joint training prevents the agents from adapting in isolation and enables coordinated responses to external forces that affect the full-body dynamics.

2.2 Forceful Interaction in Legged Robots

Forceful interaction has been extensively studied for quadrupedal robots with mounted arms, through model-based approaches—particularly MPC combined with force planning and control for robust and adaptive locomotion and manipulation [12, 13, 37]. Recent advances in RL have further enhanced adaptability, enabling quadrupeds to learn adaptive and agile force interactions including impedance control [38] and aggressive force adaptation [14]. For humanoids, forceful interaction presents significantly greater challenges due to their more complex dynamics and stringent joint limits. Unlike quadrupeds with centralized mass distributions, humanoids exhibit coupled dynamics between their upper and lower bodies, making force adaptation particularly difficult. Recent model-based approaches have demonstrated force control for heavy-duty tasks [39, 22], but these require prior knowledge of manipulated objects’ mass, center of mass (CoM), or pre-defined force trajectories, limiting their applicability to unknown disturbances. While some works have attempted explicit force estimation for humanoids [40], they are restricted to quasi-static scenarios and cannot handle force adaptation in dynamic loco-manipulation scenarios.

In this paper, FALCON learns to implicitly adapt to unknown external forces on the different EEs with a novel 3D EE force curriculum that considers humanoid joint torque limits. In this way, we can maximize the force adaptability of the learned loco-manipulation policy while ensuring the joint torque limits for robust and safe real-world deployment.

3 FALCON: Force-Adaptive Humanoid Loco-Manipulation

Humanoid loco-manipulation under external EE forces requires coordinated control of both the lower and upper body. We first formulate the problem as a unified *dual goal-conditioned* policy learning problem. Let the degrees of freedom (DoFs) of the humanoid be partitioned into lower-body joints and upper-body joints, with n^l denoting the number of lower-body DoFs, n^u the number of upper-body DoFs, and $n = n^l + n^u$ the total number of actuated joints.

The robot proprioception $s_t^p \in \mathcal{S}_t$ is defined as $s_t^p \triangleq [q_{t-4:t}, \dot{q}_{t-4:t}, \omega_{t-4:t}^{\text{root}}, g_{t-4:t}, a_{t-5:t-1}]$, which contains five-step histories of joint positions $q_t \in \mathbb{R}^n$, joint velocities $\dot{q}_t \in \mathbb{R}^n$, root angular velocity $\omega_t^{\text{root}} \in \mathbb{R}^3$, projected gravity $g_t \in \mathbb{R}^3$, and previous actions $a_{t-1} \in \mathbb{R}^n$. The goal space \mathcal{G}_t consists of locomotion goals $\mathcal{G}_t^l \triangleq [\mathbf{v}_t^{\text{lin,ang}}, \phi_t^{\text{stance}}, h_t^{\text{root}}, w_t^{\text{yaw}}]$, specifying desired root linear and angular velocities, stance indicators, root heights, and waist yaw angles, and manipulation goals $\mathcal{G}_t^u \triangleq [\mathbf{q}_t^{\text{upper}*}]$, specifying target joint configurations for the upper body where $\mathbf{q}_t^{\text{upper}*} \in \mathbb{R}^{n^u}$.

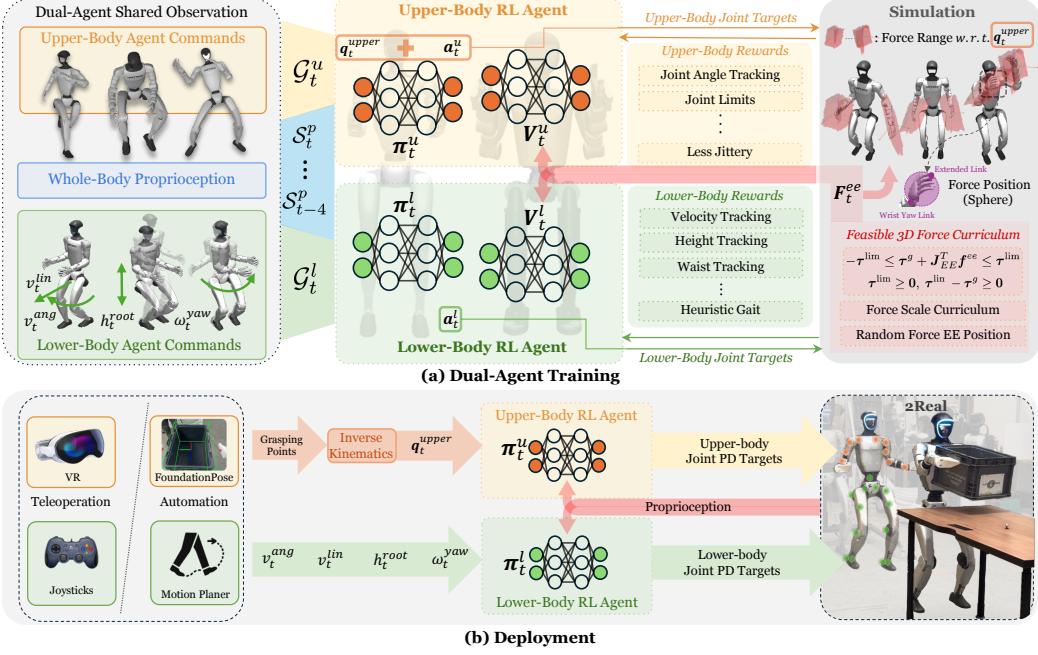


Figure 2: Overview of FALCON. (a) Two agents with different sub-tasks are jointly trained with shared whole-body proprioception. During training, we apply 3D external forces bounded by upper-body joint torque limits on the end-effectors; (b) FALCON is deployed with either teleoperation or an autonomy pipeline including FoundationPose [41] for pose estimation and motion planning.

Under this unified formalism, conventional methods differ mainly in how they generate the action $a_t \in \mathbb{R}^n$ that commands the robot joints:

- *Lower-RL-Upper-IK*: lower-body actions $a_t^l \in \mathcal{A}_t^l \subset \mathbb{R}^{n^l}$ are generated by a policy $\pi^l : s_t^p \times \langle \mathcal{G}_t^l, \mathcal{G}_t^u \rangle \mapsto \mathcal{A}_t^l$ conditioned on whole-body proprioception and goals, while upper-body actions $a_t^u \in \mathcal{A}_t^u \subset \mathbb{R}^{n^u}$ are computed through inverse kinematics (IK) solvers based on \mathcal{G}_t^u .
- *Monolithic-Whole-body-RL*: a single policy $\pi : s_t^p \times \mathcal{G}_t \mapsto a_t$ directly predicts the full-body action $a_t \in \mathbb{R}^n$, attempting to satisfy both locomotion and manipulation objectives simultaneously.

While *Lower-RL-Upper-IK* methods are sample-efficient, they neglect upper-body force compensation and whole-body coupling under EE force disturbances. In contrast, *Monolithic-Whole-body-RL* methods improve expressiveness but suffer from exploration inefficiency due to the large action space spanning coarsely related locomotion and manipulation objectives. To overcome these challenges, we introduce FALCON, a dual-agent RL framework that achieves training efficiency and coordination through decomposition learning with shared whole-body observation.

3.1 Dual-Agent Learning Framework

As shown in Figure 2, FALCON jointly trains two agents, each specialized for a different subtask. The lower-body locomotion agent learns a policy $\pi^l : s_t^p \times \mathcal{G}_t^l \mapsto \mathcal{A}_t^l$ with value function $V^l(\cdot)$, while the upper-body manipulation agent learns a policy $\pi^u : s_t^p \times \mathcal{G}_t^u \mapsto \mathcal{A}_t^u$ with value function $V^u(\cdot)$. Both agents observe the same proprioceptive input s_t^p but optimize independent goal-conditioned objectives:

$$r_t^l = \mathcal{R}^l(s_t^p, \mathcal{G}_t^l) \text{ (locomotion)} \quad r_t^u = \mathcal{R}^u(s_t^p, \mathcal{G}_t^u) \text{ (manipulation)} \quad (1)$$

These two policy parameters θ_l and θ_u are updated via proximal policy optimization (PPO [42]):

$$\max_{\theta_l} \mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} r_t^l \right] \text{ (Lower-body)} \quad \max_{\theta_u} \mathbb{E} \left[\sum_{t=1}^T \gamma^{t-1} r_t^u \right] \text{ (Upper-body)} \quad (2)$$

where γ is the discount factor. The upper-body target joint angles q_t^{upper} (target joints of shoulders, elbows, wrists) are randomly sampled from the AMASS dataset [43] during training, and calculated via IK during deployment. The combined action from the two agents $a_t = [a_t^l; a_t^u]$ is sent to a joint-level PD controller. As the real-world humanoid control is inherently partially observable, we adopt asymmetric actor-critic training, where critics additionally access privileged information including root linear velocities and EE forces F_t^{ee} during training but not during deployment. Detailed reward designs and domain randomization during training are provided in Appendix A.1 and Appendix A.2.

3.2 Torque-Limit-Aware 3D Force Curriculum

For humanoid robots—particularly those with relatively weak joint torque limits, such as the wrist joints on the Unitree Humanoid G1—it is crucial to explicitly account for these torque constraints when large external disturbances are applied to the end-effectors (EEs). Ignoring these limits during upper body policy training can lead to unexpected or unsafe behaviors due to torque saturation or joint limit violations in real-robot deployment. Additionally, it's important to gradually increase the external force during training, allowing the policy to progressively learn effective force adaptation strategies. To achieve these, our force application framework follows through three principles:

Torque-Aware Force Computation: Before applying forces, we first need to estimate the maximum forces that we can exert on the left or right end-effector. Given the left or right end-effector Jacobian $J_{EE} \in \mathbb{R}^{3 \times \frac{n_u}{2}}$ at its Center of Mass (CoM), their joint torque limit $\tau^{\lim} \in \mathbb{R}^{3 \times \frac{n_u}{2}}$ (with $\tau^{\lim} \geq 0$), and the gravity compensation torque $\tau^g \in \mathbb{R}^{3 \times \frac{n_u}{2}}$ which satisfies $-\tau^{\lim} \leq \tau^g \leq \tau^{\lim}$ to ensure feasibility, we estimate the maximum and minimum admissible forces f_i^{\max}, f_i^{\min} along each Cartesian axis $i \in x, y, z$ by analyzing the worst-case joint torque induced by a unit force applied in each direction. The element-wise force bound can be computed in parallel as:

$$-\tau^{\lim} \leq \tau^g + J_{EE}^T f^{ee} \leq \tau^{\lim} \quad (3)$$

$$f_i^{\max} = \min_j \left(\frac{\tau_j^{\lim} - \tau_j^g}{|J_{EE}^{ji}| + \epsilon} \right), \quad f_i^{\min} = \max_j \left(\frac{-\tau_j^{\lim} - \tau_j^g}{|J_{EE}^{ji}| + \epsilon} \right), \quad (4)$$

where J_{EE}^{ji} denotes the (j, i) -th element of the end-effector Jacobian matrix, and ϵ is a small positive constant to prevent division by zero. After that, we sample the relatively ratio $\gamma = [\gamma_x, \gamma_y, \gamma_z]$ among x, y and z axis through Dirichlet Distribution [44], which satisfy $\sum_{i \in \{x, y, z\}} \gamma_i = 1$. The feasible applied force will be uniformly sampled within the estimated range and expressed as:

$$f_t^{ee} = \sum_{i \in \{x, y, z\}} F_i \cdot e_i, \quad \text{where } F_i \sim \mathcal{U}[\gamma_i \cdot f_i^{\min}, \gamma_i \cdot f_i^{\max}] \quad (5)$$

This approach maximizes force adaptivity while respecting torque limits, leading to more effective training than random sampling, as explained in Section 4.3. Note that applied forces may differ between left and right EEs due to asymmetric upper-body configurations (Figure 2).

Progressive Force Curriculum: To facilitate progressive force adaptation, the estimated EE forces are scaled by a global factor $\alpha_g \in (0, 1)$, increasing over training, so the applied force becomes $F_t^{ee} = \alpha_g \cdot f_t^{ee}$. During walking, planar forces are projected opposite to the velocity. A low-pass filter is applied to reduce force jitter.

Position Randomization of the Applied Force: Learning-based force adaptation leverages proprioceptive history to implicitly compensate for external forces, removing the need for explicit force estimation [40] or sensing [45]. To improve robustness to variations in end-effector (EE) contact points—which alter the torque mapping via the EE Jacobian—we randomize force application along the EE link, from the wrist yaw to the distal segment, as illustrated in Figure 2.

Methods	$E_{\text{tracking}}^{\text{upper}} \downarrow$			$E_{\text{tracking}}^{\text{root}} \downarrow$			
	N-Force	M-Force	L-Force	N-Force	M-Force	L-Force	
Lower-RL	PD-w/o-Force-Curr.	0.46 ± 0.04	0.94 ± 0.05	1.44 ± 0.06	0.38 ± 0.04	0.66 ± 0.05	1.14 ± 0.06
	PD-Force-Curr.	0.44 ± 0.03	0.93 ± 0.03	1.42 ± 0.04	0.33 ± 0.03	0.38 ± 0.03	0.46 ± 0.03
	PID-Force-Curr.	0.24 ± 0.03	0.31 ± 0.03	0.60 ± 0.04	0.32 ± 0.03	0.35 ± 0.03	0.46 ± 0.04
	PD-ID-Force-Curr.	0.29 ± 0.03	0.38 ± 0.03	0.53 ± 0.04	0.40 ± 0.03	0.42 ± 0.03	0.47 ± 0.04
M-WB-RL	w/o-Force-Curr.	0.46 ± 0.05	1.03 ± 0.07	1.65 ± 0.08	0.34 ± 0.04	0.67 ± 0.05	1.28 ± 0.07
	with-Force-Curr.	0.43 ± 0.04	0.50 ± 0.04	0.73 ± 0.05	0.28 ± 0.03	0.32 ± 0.03	0.44 ± 0.04
FALCON	w/o-Force-Curr.	0.14 ± 0.03	0.55 ± 0.04	1.06 ± 0.06	0.32 ± 0.03	0.66 ± 0.05	1.24 ± 0.06
	with-Force-Curr.	0.21 ± 0.03	0.24 ± 0.03	0.37 ± 0.04	0.27 ± 0.03	0.30 ± 0.03	0.45 ± 0.04

Table 1: Loco-Manipulation Evaluation of FALCON and Baselines in IsaacGym.

4 Simulation and Real-World Experiments

In this section, we present extensively quantitative comparison between FALCON and the baselines as well as qualitative results on real-world deployment. We choose Unitree Humanoid G1 and Booster T1 as our humanoid platforms. Specifically, we address the following key questions:

Q1: Can FALCON outperform other baselines in terms of both upper-body manipulation and lower-body locomotion performance?

Q2: Why does FALCON has better training-efficiency compared to Monolithic-Whole-body-RL (M-WB-RL) for force-adaptive loco-manipulation?

Q3: Does FALCON work for different humanoids to show cross-platform generalizability?

4.1 Evaluation Criterion

To evaluate the performance of the learned low-body locomotion and upper-body manipulation capabilities, we consider the following metrics under dynamically unknown and 3D EE forces $\mathbf{F}_t \in \mathbb{R}^3$, given a sequence of target upper-body joints $\mathbf{q}_t^{\text{upper}*}$, target root velocities $\mathbf{v}_t^{\text{lin,ang}*}$ and stance signal ϕ_t^{stance} , where $t = 1, 2, \dots, T$ and T is the sequence length:

$$(ii) \text{Upper-Body Joints Tracking Error: } E_{\text{tracking}}^{\text{upper}}(\mathbf{q}_t^{\text{upper}*}) = \frac{1}{T} \sum_{t=1}^T \left| \mathbf{q}_t^{\text{upper}} - \mathbf{q}_t^{\text{upper}*} \right|$$

$$(iii) \text{Root Velocity Tracking Error: } E_{\text{tracking}}^{\text{root}}(\mathbf{v}_t^{\text{lin,ang}*}) = \frac{1}{T} \sum_{t=1}^T \left| \mathbf{v}_t^{\text{lin,ang}} - \mathbf{v}_t^{\text{lin,ang}*} \right|$$

4.2 Baselines

We consider two types of baseline methods for force adaptation, both trained under the same goal space (e.g., commands) in Section 3.1 and force curriculum described in Section 3.2, with each type further including relevant ablation variants.

Decoupled Lower-body RL with Upper-body IK Controllers. For all variants, RL is used for lower-body locomotion, and IK provides target upper-body joint angles from end-effector poses. The key differences lie in the use of force curriculum and the upper-body joint tracking strategy:

- (a) *Upper-PD-w/o-Force-Curr.:* A baseline following [8, 9], using PD control for upper-body joint tracking without force randomization.
- (b) *Upper-PD:* Extends (a) by incorporating force curriculum, enabling lower-body adaptation to external forces; upper-body remains PD-controlled.
- (c) *Upper-PID:* Extends (b) by adding an integral term to the upper-body controller to reduce steady-state tracking error.

Methods	$E_{\text{tracking}}^{\text{upper}} \downarrow$			$E_{\text{tracking}}^{\text{root}} \downarrow$		
	N-Force	M-Force	L-Force	N-Force	M-Force	L-Force
w/o-Torque-Limit-Aware.	0.42 ± 0.07	0.45 ± 0.08	0.61 ± 0.10	0.45 ± 0.03	0.46 ± 0.05	0.54 ± 0.05
FALCON with-Torque-Limit-Aware.	0.23 ± 0.03	0.26 ± 0.03	0.36 ± 0.04	0.30 ± 0.03	0.32 ± 0.03	0.39 ± 0.02

Table 2: Evaluation of FALCON using torque-limit-aware (Max-Force-Estimation) curriculum versus w/o *torque-limit-aware* force curriculum in IsaacGym. Our curriculum achieves significantly better tracking performance, especially for upper-body manipulation under large forces.

- (d) *Upper-PD-ID*: Extends (a) with a learned force estimator [38] and inverse-dynamics-based torque compensation under quasi-static assumptions (details in Appendix A.3).

Monolithic Whole-body RL

- (e) *Monolithic-WB-RL-w/o-Force-Curr*: Built upon prior designs [11, 15], a single agent is trained with the same goal commands as FALCON, but without applying any force during training.
- (f) *Monolithic-WB-RL-with-Force-Curr*: Based on (e), we adopt force randomization into the training curriculum for force adaptation, while keeping the other training settings identical.

4.3 Simulation Results

To answer **Q1** (*Can FALCON outperform other baselines in terms of both upper-body manipulation and lower-body locomotion performance?*) and **Q2** (*Why does FALCON has better training-efficiency compared to Monolithic-Whole-body-RL (M-WB-RL) for force-adaptive locomotion?*), we conduct quantitative comparisons of our method with other two baselines in IsaacGym on Unitree Humanoid G1.

Loco-Manipulation Performance: We evaluate FALCON and baselines on 252 ACCAD [46] motion targets under three force levels: (i) No-Force ($\alpha_g = 0$), (ii) Middle-Force ($\alpha_g = 0.5$), and (iii) Large-Force ($\alpha_g = 1.0$), applied to both end-effectors. As shown in Table 1, across all settings, FALCON with force curriculum achieves the lowest tracking errors in both upper-body motion ($E_{\text{tracking}}^{\text{upper}}$) and root velocity ($E_{\text{tracking}}^{\text{root}}$), demonstrating robust manipulation under disturbance. Under L-Force, it reduces upper-body error to 0.37, outperforming PID-Force-Curr. (0.60) and M-WB-RL (0.73). Root error remains low at 0.45, indicating stable locomotion. While force curriculum benefits all methods, FALCON gains most due to its decomposed learning structure.

Torque-Limit-Aware Force Curriculum To assess the effectiveness of the proposed torque-limit-aware force curriculum (Section 3), we compare it with a baseline that samples random forces from a wide clipping range ($X : [-100N, 100N]$, $Y : [-100N, 100N]$, $Z : [-100N, 5N]$) without enforcing torque feasibility. Training curves and quantitative results are shown in Figure 3 and Table 2. During evaluation, applied forces remain bounded by the estimated admissible limits.

Figure 3 (a) shows that force curriculum saturates at a force scale $\alpha_g = 0.6$ due to frequent violations of torque limits, which hinder further progression. Additionally, as illustrated in Figure 3 (b), the force curriculum w/o *torque-limit-aware* results in larger upper-body tracking errors, since excessive forces regularly exceed the feasible torque bounds, impairing the learning of effective upper body force compensation. Consequently, as shown in Table 2, policies trained w/o *torque-limit-aware* force cur-

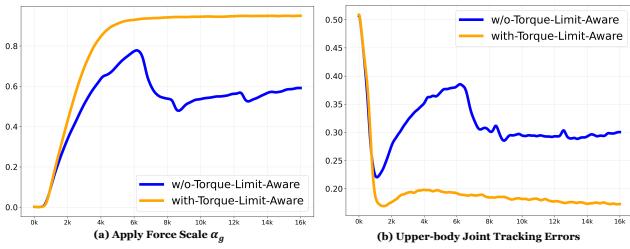


Figure 3: (a) Progression of the Apply Force Scale α_g ; (b) Upper-body Joint Tracking Errors During Training

Methods	$E_{\text{tracking}}^{\text{upper}}$ ↓			$E_{\text{tracking}}^{\text{root}}$ ↓		
	N-Force	M-Force	L-Force	N-Force	M-Force	L-Force
FALCON	Smaller-Force-Clip. 0.21 ± 0.03	0.24 ± 0.03	0.37 ± 0.04	0.27 ± 0.03	0.30 ± 0.03	0.45 ± 0.04
	Larger-Force-Clip. 0.23 ± 0.03	0.26 ± 0.03	0.36 ± 0.04	0.30 ± 0.03	0.32 ± 0.03	0.39 ± 0.02

Table 3: Evaluation of FALCON with a smaller force clip range in Table 1 versus a larger force clip range in Table 2.

riculum tend to overfit to the locomotion objective, compromising upper-body accuracy. In contrast, our torque-limit-aware curriculum facilitates balanced learning of both upper-body joint tracking and root velocity tracking under significant external disturbances.

Note that in Table 1, we use a narrower force clipping range ($X : [-50N, 50N]$, $Y : [-50N, 50N]$, $Z : [-60N, 5N]$) compared to the wider range in Table 2 ($X : [-100N, 100N]$, $Y : [-100N, 100N]$, $Z : [-100N, 5N]$). The results in Table 3 show that increasing the force range has minimal impact on loco-manipulation performance, highlighting the robustness of our torque-limit-aware force curriculum.

Exploration and Learning:

(i) *Action Noise Std*: As shown in Figure 4 (a), FALCON exhibits faster and smoother noise decay in both upper and lower body actions, indicating more efficient and stable exploration. In contrast, M-WB-RL suffers from prolonged noise due to entangled control objectives, especially for upper body actions.

(ii) *Reward and Postural Stability*: As shown in Figure 4 (b), FALCON achieves less tracking errors in both upper-body joints and base angular velocity while

in M-WB-RL these two reward terms tend to fluctuate. Additionally, M-WB-RL suffers from larger torso and ankle penalties due to excessive whole-body compensation, resulting in unnatural bending and CoM upright misalignment as shown in Figure 5.

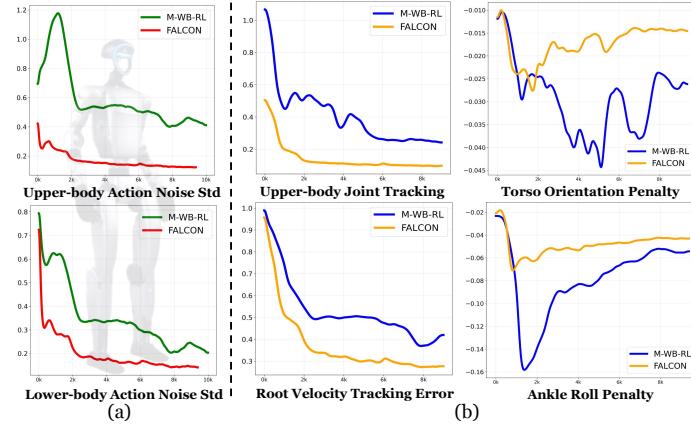


Figure 4: Comparison of FALCON and M-WB-RL: (a) action noise std; (b) tracking errors and penalties.

4.4 Real-World Quantitative Tracking Results

We evaluate FALCON on Unitree G1 with each hand loaded with 1.2kg payload in a real-world task, which is walking at (0.5, 0.0)m/s with zero angular velocity, fixed height and waist, and keeping the upper body in its default position. We compare against two baselines: (i) *Upper-PD with Force Curriculum*, and (ii) *Monolithic-WB-RL with Force Curriculum*. As shown in Table 4, FALCON achieves the lowest tracking errors, and perform stable and natural motion in heavy-duty loco-manipulation.

4.5 Real-World Deployment with Teleoperation

To answer **Q3** (*Does FALCON work for different humanoids to show cross-platform generalizability?*), we deploy policies trained in simulation on the Unitree G1 and Booster T1 humanoids **without any reward or force curriculum modifications**, thanks to FALCON efficient dual-agent training and torque-limit-aware 3D force curriculum. As shown in Figure 1, we evaluate the policies on three forceful loco-manipulation tasks: (1) **Transporting Payloads**, with 0-20N vertical forces while maintaining stable locomotion and precise upper-body joint tracking; (2) **Cart-Pulling**, with up to

Method	$E_{\text{tracking}}^{\text{upper}}$	$E_{\text{tracking}}^{\text{root}}$
Upper-PD-Force-Curr.	1.81 ± 0.13	0.40 ± 0.04
M-WB-RL-Force-Curr.	0.81 ± 0.11	0.58 ± 0.05
FALCON	0.39 ± 0.08	0.42 ± 0.03

Table 4: Real-world Tracking Errors.

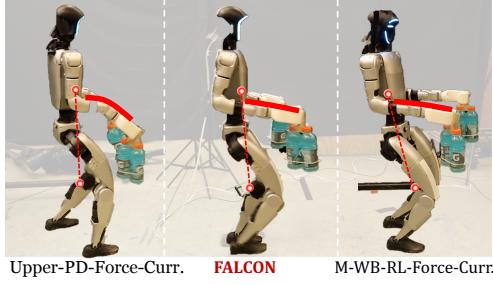


Figure 5: Real-World Payload Transportation.

Figure 6: **Autonomous Tote Logistics:** a humanoid robot walks without a tote, picks up the tote, walks with the tote, and drops off the tote.

100N longitudinal (X-Y) forces during walking; and (3) **Door-Opening**, with up to 40N 3D forces during stance. These force ranges are measured through a force gauge shown in Appendix A.4.

These results demonstrate that FALCON enables robust policy transfer across platforms with different morphologies and actuation. The learned policies exhibit effective whole-body compensation: the upper body responds adaptively to 3D forces, the lower body leans against significant longitudinal forces, and the base height remains stable under vertical loads.

4.6 Real-World Deployment with Autonomy

We also deploy FALCON on the Unitree G1 for autonomous tote logistics, a representative warehouse task. As illustrated in Figure 6, the robot is required to walk from an initial location to a pickup station, lift a tote of unknown weight, and transport it to a designated area for precise placement. The detailed implementation of the autonomous pipeline can be found in Appendix A.5.

5 Conclusion

In this paper, we introduce FALCON, a dual-agent reinforcement learning framework designed for force-adaptive humanoid loco-manipulation. By decoupling the learning of the upper and lower body, while maintaining coordination through shared proprioceptive feedback, FALCON achieves superior adaptability in handling 3D end-effector forces during complex tasks. Our extensive evaluation demonstrates that FALCON outperforms both Lower-RL-Upper-IK and Monolithic-Whole-body-RL baselines, achieving faster training convergence, reduced tracking errors, and more stable performance across a variety of force regimes. Moreover, FALCON exhibits strong cross-platform generalizability, successfully transferring policies from simulation to physical humanoids, including tasks like transporting payloads, cart-pulling, and door-opening. These results underscore FALCON’s potential for real-world deployment in forceful interaction scenarios.

6 Limitations

Despite its strong performance, FALCON has two key limitations. First, it focuses solely on force disturbances applied to the end-effectors, without accounting for contact forces on other body parts or supporting multi-contact interactions. This restricts its applicability in scenarios involving whole-body support, such as leaning, bracing, or collaborative lifting. Second, the current force curriculum only considers external forces and ignores external torques. As a result, FALCON may struggle in tasks that involve rotational disturbances, such as operating handles or tools with eccentric loading. Addressing these limitations by incorporating multi-contact reasoning and torque-adaptive policies remains an important avenue for future research.

Acknowledgments

We would like to thank our other CMU MRSD Capstone teammates, Ishita Gupta and Shivang Vijay, for their valuable contributions to the autonomy part in this project. We are also grateful to our Capstone advisors, Prof. John Dolan and Prof. Dimitrios Apostolopoulos, for their continuous guidance and support. We acknowledge the hardware support from Unitree Robotics and Booster Robotics. Finally, we thank Haoyang Weng, Wenli Xiao and Yitang Li for their insightful discussions that helped shape the direction of this work.

References

- [1] Z. Gu, J. Li, W. Shen, W. Yu, Z. Xie, S. McCrary, X. Cheng, A. Shamsah, R. Griffin, C. K. Liu, et al. Humanoid locomotion and manipulation: Current progress and challenges in control, planning, and learning. [arXiv preprint arXiv:2501.02116](https://arxiv.org/abs/2501.02116), 2025.
- [2] K. Darvish, L. Penco, J. Ramos, R. Cisneros, J. Pratt, E. Yoshida, S. Ivaldi, and D. Pucci. Teleoperation of humanoid robots: A survey. *IEEE Transactions on Robotics*, 39(3):1706–1727, 2023.
- [3] B. Dynamics. Atlas® and beyond: the world’s most dynamic robots, 2024. URL <https://bostondynamics.com/atlas/>.
- [4] Unitree. Humanoid agent ai avatar, 2024. URL <https://www.unitree.com/g1>.
- [5] F. AI. Figure 02 humanoid robot, 2024. URL <https://www.figure.ai/>.
- [6] B. Robotics. Booster t1 humanoid robot, 2025. URL <https://www.boosterobotics.com/>.
- [7] E. Krotkov, D. Hackett, L. D. Jackel, M. Perschbacher, J. Pippine, J. Strauss, G. A. Pratt, and C. Orlowski. The DARPA robotics challenge finals: Results and perspectives. *J. Field Robotics*, 34(2):229–240, 2017. doi:[10.1002/ROB.21683](https://doi.org/10.1002/ROB.21683). URL <https://doi.org/10.1002/rob.21683>.
- [8] C. Lu, X. Cheng, J. Li, S. Yang, M. Ji, C. Yuan, G. Yang, S. Yi, and X. Wang. Mobile-television: Predictive motion priors for humanoid whole-body control. [arXiv preprint arXiv:2412.07773](https://arxiv.org/abs/2412.07773), 2024.
- [9] Q. Ben, F. Jia, J. Zeng, J. Dong, D. Lin, and J. Pang. HOMIE: humanoid loco-manipulation with isomorphic exoskeleton cockpit. *CoRR*, abs/2502.13013, 2025. doi:[10.48550/ARXIV.2502.13013](https://doi.org/10.48550/ARXIV.2502.13013). URL <https://doi.org/10.48550/arXiv.2502.13013>.
- [10] M. Ji, X. Peng, F. Liu, J. Li, G. Yang, X. Cheng, and X. Wang. Exbody2: Advanced expressive humanoid whole-body control. [arXiv preprint arXiv:2412.13196](https://arxiv.org/abs/2412.13196), 2024.
- [11] X. Cheng, Y. Ji, J. Chen, R. Yang, G. Yang, and X. Wang. Expressive whole-body control for humanoid robots. [arXiv preprint arXiv:2402.16796](https://arxiv.org/abs/2402.16796), 2024.

- [12] M. Sombolestian and Q. Nguyen. Hierarchical adaptive loco-manipulation control for quadruped robots. In 2023 IEEE International Conference on Robotics and Automation (ICRA), pages 12156–12162. IEEE, 2023.
- [13] M. Sombolestian and Q. Nguyen. Adaptive force-based control of dynamic legged locomotion over uneven terrain. IEEE Transactions on Robotics, 2024.
- [14] N. Fey, G. B. Margolis, M. Peticco, and P. Agrawal. Bridging the sim-to-real gap for athletic loco-manipulation. arXiv preprint arXiv:2502.10894, 2025.
- [15] T. He, W. Xiao, T. Lin, Z. Luo, Z. Xu, Z. Jiang, C. Liu, G. Shi, X. Wang, L. Fan, and Y. Zhu. Hover: Versatile neural whole-body controller for humanoid robots. arXiv preprint arXiv:2410.21229, 2024.
- [16] T. He, Z. Luo, W. Xiao, C. Zhang, K. Kitani, C. Liu, and G. Shi. Learning human-to-humanoid real-time whole-body teleoperation. arXiv preprint arXiv:2403.04436, 2024.
- [17] T. He, Z. Luo, X. He, W. Xiao, C. Zhang, W. Zhang, K. Kitani, C. Liu, and G. Shi. Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. arXiv preprint arXiv:2406.08858, 2024.
- [18] T. He, J. Gao, W. Xiao, Y. Zhang, Z. Wang, J. Wang, Z. Luo, G. He, N. Sobanbab, C. Pan, et al. Asap: Aligning simulation and real-world physics for learning agile humanoid whole-body skills. arXiv preprint arXiv:2502.01143, 2025.
- [19] Z. Zhuang and H. Zhao. Embrace collisions: Humanoid shadowing for deployable contact-agnostics motions. arXiv preprint arXiv:2502.01465, 2025.
- [20] Whole body model predictive control with a memory of motion: Experiments on a torque-controlled talos. In 2021 IEEE International Conference on Robotics and Automation (ICRA), pages 8202–8208. IEEE, 2021.
- [21] J. Li, J. Ma, O. Kolt, M. Shah, and Q. Nguyen. Dynamic loco-manipulation on hector: Humanoid for enhanced control and open-source research. arXiv preprint arXiv:2312.11868, 2023.
- [22] M. Murooka, K. Chappellet, A. Tanguy, M. Benallegue, I. Kumagai, M. Morisawa, F. Kanehiro, and A. Kheddar. Humanoid loco-manipulations pattern generation and stabilization control. IEEE Robotics and Automation Letters, 6(3):5597–5604, 2021.
- [23] K. Bouyarmane, K. Chappellet, J. Vaillant, and A. Kheddar. Quadratic programming for multirobot and task-space force control. IEEE Transactions on Robotics, 35(1):64–77, 2018.
- [24] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim. Dynamic locomotion in the mit cheetah 3 through convex model-predictive control. In 2018 IEEE/RSJ international conference on intelligent robots and systems (IROS), pages 1–9. IEEE, 2018.
- [25] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In 2003 IEEE international conference on robotics and automation (Cat. No. 03CH37422), volume 2, pages 1620–1626. IEEE, 2003.
- [26] Z. Fu, Q. Zhao, Q. Wu, G. Wetzstein, and C. Finn. Humanplus: Humanoid shadowing and imitation from humans. In Conference on Robot Learning (CoRL), 2024.
- [27] C. Zhang, W. Xiao, T. He, and G. Shi. Wococo: Learning whole-body humanoid control with sequential contacts. arXiv preprint arXiv:2406.06005, 2024.
- [28] J. Cheng, D. Kang, G. Fadini, G. Shi, and S. Coros. Rambo: Rl-augmented model-based optimal control for whole-body loco-manipulation. arXiv preprint arXiv:2504.06662, 2025.

- [29] Y. Zhong, C. Zhang, T. He, and G. Shi. Bridging adaptivity and safety: Learning agile collision-free locomotion across varied physics. *arXiv preprint arXiv:2501.04276*, 2025.
- [30] H. Xue, C. Pan, Z. Yi, G. Qu, and G. Shi. Full-order sampling-based mpc for torque-level locomotion control via diffusion-style annealing. *arXiv preprint arXiv:2409.15610*, 2024.
- [31] Y. Zhang, T. Liang, Z. Chen, Y. Ze, and H. Xu. Catch it! learning to catch in flight with mobile dexterous hands. *arXiv preprint arXiv:2409.10319*, 2024.
- [32] J. Dao, H. Duan, and A. Fern. Sim-to-real learning for humanoid box loco-manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 16930–16936. IEEE, 2024.
- [33] F. Liu, Z. Gu, Y. Cai, Z. Zhou, S. Zhao, H. Jung, S. Ha, Y. Chen, D. Xu, and Y. Zhao. Opt2skill: Imitating dynamically-feasible whole-body trajectories for versatile humanoid loco-manipulation. *arXiv preprint arXiv:2409.20514*, 2024.
- [34] S. Gronauer and K. Diepold. Multi-agent deep reinforcement learning: a survey. *Artificial Intelligence Review*, 55(2):895–943, 2022.
- [35] K. Zhang, Z. Yang, and T. Başar. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *Handbook of reinforcement learning and control*, pages 321–384, 2021.
- [36] Y. Wang, R. Sagawa, and Y. Yoshiyasu. Learning advanced locomotion for quadrupedal robots: A distributed multi-agent reinforcement learning framework with riemannian motion policies. *Robotics*, 13(6):86, 2024.
- [37] A. Rigo, M. Hu, S. K. Gupta, and Q. Nguyen. Hierarchical optimization-based control for whole-body loco-manipulation of heavy objects. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 15322–15328. IEEE, 2024.
- [38] T. Portela, G. B. Margolis, Y. Ji, and P. Agrawal. Learning force control for legged manipulation. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 15366–15372. IEEE, 2024.
- [39] J. Li and Q. Nguyen. Kinodynamics-based pose optimization for humanoid loco-manipulation. *arXiv preprint arXiv:2303.04985*, 2023.
- [40] T. Mattioli and M. Venditti. Interaction force reconstruction for humanoid robots. *IEEE Robotics and Automation Letters*, 2(1):282–289, 2016.
- [41] B. Wen, W. Yang, J. Kautz, and S. Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17868–17879, 2024.
- [42] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *CoRR*, abs/1707.06347, 2017. URL <http://arxiv.org/abs/1707.06347>.
- [43] N. Mahmood, N. Ghorbani, N. F. Troje, G. Pons-Moll, and M. J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pages 5442–5451, Oct. 2019.
- [44] K. W. Ng, G.-L. Tian, and M.-L. Tang. Dirichlet and related distributions: Theory, methods and applications. 2011.
- [45] X. Guo, G. He, J. Xu, M. Mousaei, J. Geng, S. Scherer, and G. Shi. Flying calligrapher: Contact-aware motion and force planning and control for aerial manipulation. *IEEE Robotics and Automation Letters*, 2024.

- [46] Advanced Computing Center for the Arts and Design. ACCAD MoCap Dataset. URL <https://accad.osu.edu/research/motion-lab/mocap-system-and-data>.
- [47] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012, pages 5026–5033. IEEE, 2012. doi:[10.1109/IROS.2012.6386109](https://doi.org/10.1109/IROS.2012.6386109). URL <https://doi.org/10.1109/IROS.2012.6386109>.

A Appendix

A.1 Reward Terms

We adopt the similar reward terms from [17, 11], but introduce some important penalties to ensure the locomotion stability under significant external forces, and other tracking rewards for squat and waist twist. The additional reward terms are summarized in Table 5:

Table 5: Additional Reward components and weights: penalty rewards for preventing undesired behaviors for sim-to-real transfer, and task rewards to achieve desired loco-manipulation capability.

Term	Expression	Weight
Penalty		
Hip pos	$\ \mathbf{q}_{roll,pitch}^{hip}\ $	-2.5
Negative knee joint	$\sum_j \mathbb{1}[q_j < q_j^{\min}]$	-1.0
Stance tap feet	$ (\mathbf{p}_{left,foot} - \mathbf{p}_{right,foot})_x $ in base frame	-5.0
Stance root	$ (\mathbf{p}_{root} - \text{mid}(\mathbf{p}_{feet}))^y $	-5.0
Stand still	$\mathbb{1}[\text{no contact}]$	-0.15
Ankle roll	$\sum_j q_j^{roll} $	-2.0
Task Reward		
Root linear velocity x	$\exp(-4.0\ \mathbf{v}_t^x - \mathbf{v}_t^{x*}\ _2)$	2
Root linear velocity y	$\exp(-4.0\ \mathbf{v}_t^x - \mathbf{v}_t^{x*}\ _2)$	1.5
Root angular velocity	$\exp(-4.0\ \mathbf{v}_t^{ang} - \mathbf{v}_t^{ang*}\ _2)$	4
Root walk height	$\exp\left(-\frac{ \text{command}_z - p_z^{\text{root}} }{0.05}\right)$	2
Waist dofs	$\exp\left(-\frac{\sum_{\theta \in \{\text{yaw}, \text{roll}, \text{pitch}\}} (\theta^{\text{sim}} - \theta^{\text{cmd}})^2}{0.05}\right)$	2
Upper body dofs	$\exp\left(-\frac{\ \mathbf{q}_{upper} - \mathbf{q}_{ref}\ _2^2}{0.01}\right)$	4

A.2 Domain Randomization

We apply the following domain randomization terms during training, which are important for successful sim-to-real transfer.

Table 6: Domain randomization terms including dynamics randomization and external perturbation.

Term	Value
Dynamics Randomization	
Friction	$\mathcal{U}(0.5, 1.25)$
Link mass	$\mathcal{U}(0.9, 1.2) \times \text{default kg}$
Base mass	$\mathcal{U}(-1.0, 3.0) \text{ kg}$
P Gain	$\mathcal{U}(0.9, 1.1) \times \text{default}$
D Gain	$\mathcal{U}(0.9, 1.1) \times \text{default}$
Control delay	$\mathcal{U}(0, 20)\text{ms}$
External Perturbation	
Push robot	interval = 5s, $v_{xy} = 1\text{m/s}$

A.3 Lower-RL-Upper-IK with Force Estimator

We jointly train a 3D force estimator, following a similar approach to [38], using the robot's proprioception as input $\mathbf{s}_t^P \triangleq [\mathbf{q}_{t-4:t}, \dot{\mathbf{q}}_{t-4:t}, \omega_{t-4:t}^{\text{root}}, \mathbf{g}_{t-4:t}, \mathbf{a}_{t-5:t-1}^l]$. As illustrated in Figure 7 (a), the estimator predicts the end-effector forces $\tilde{\mathbf{F}}_t^{ee}$, which are then concatenated with full-body proprioception and fed into the lower-body RL policy. Meanwhile, the upper-body joint torques with force compensation are computed as $\boldsymbol{\tau} = K_p(\mathbf{q}_t^{\text{upper}} - \mathbf{q}_t^{\text{upper}*}) + K_d\dot{\mathbf{q}}_t^{\text{upper}} + \mathbf{J}_{EE}^T \tilde{\mathbf{F}}_t^{ee}$.

We compare the estimated and applied forces in Figure 7 (b), showing close alignment between the two. However, even with accurate force estimates, changes in the contact point on the end-effector

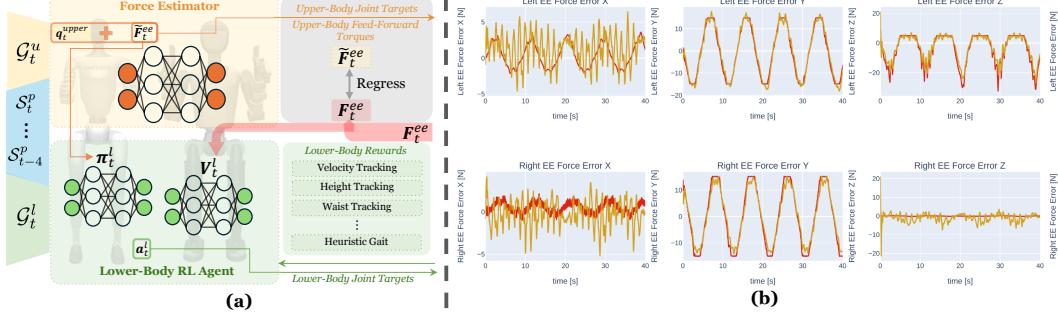


Figure 7: (a) Lower-RL-Upper-IK with Force Estimator; (b) Force Estimator Results: yellow lines are the estimated forces while the red lines are the actual forces.

alter the effective Jacobian \mathbf{J}'_{EE}^T , making the compensation term $\mathbf{J}_{EE}^T \tilde{\mathbf{F}}_t^{ee}$ inaccurate. Therefore, a force sensor is still necessary during deployment to localize the force application and compute the correct \mathbf{J}'_{EE}^T . Moreover, the compensation assumes quasi-static conditions, introducing additional error when upper-body joints are moving.

A.4 Force Measurement

Here, we use Mxmoonfree-Digital-500N-Force-Gauge to measure the peak forces needed for the following force-adaptive tasks: (1) **Cart-Pulling** for Booster T1 with a Unitree G1 and a Unitree H1 in the cart; (2) **Door-Opening**; (3) **Stance-Pulling** for the Unitree G1 and Booster T1.

Here, **Stance-Pulling** refers to applying longitudinal forces along the X-Y plane while the robot maintains a static stance, and measuring the maximum force it can resist without losing balance. Notably, the Booster T1 demonstrates a higher peak resistive force compared to the Booster T1, primarily due to its lower center of mass (CoM), which contributes to better stability with longitudinal resistance.

A.5 Autonomy Pipeline

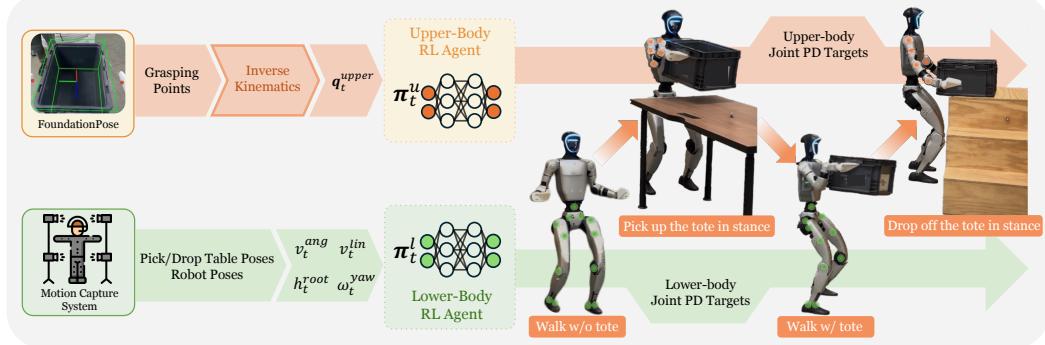


Figure 9: Overview of the autonomy pipeline for FALCON. The system integrates FALCON with 6-DoF object pose estimation via FoundationPose, MoCap-based global localization, and inverse kinematics for grasp planning to enable a humanoid robot to perform tote logistics tasks: walk without a tote, pick up the tote in stance, walk with the tote, and drop off the tote.

We develop an hierarchically autonomous pipeline for tote logistics, leveraging a Motion Capture (MoCap) system to localize positions of the robot and desks. The robot is controlled by a state-machine framework with four states: (1) walking without the tote, (2) picking up the tote in stance, (3) walking with the tote, and (4) dropping off the tote in stance, as illustrated in Fig.9. To estimate

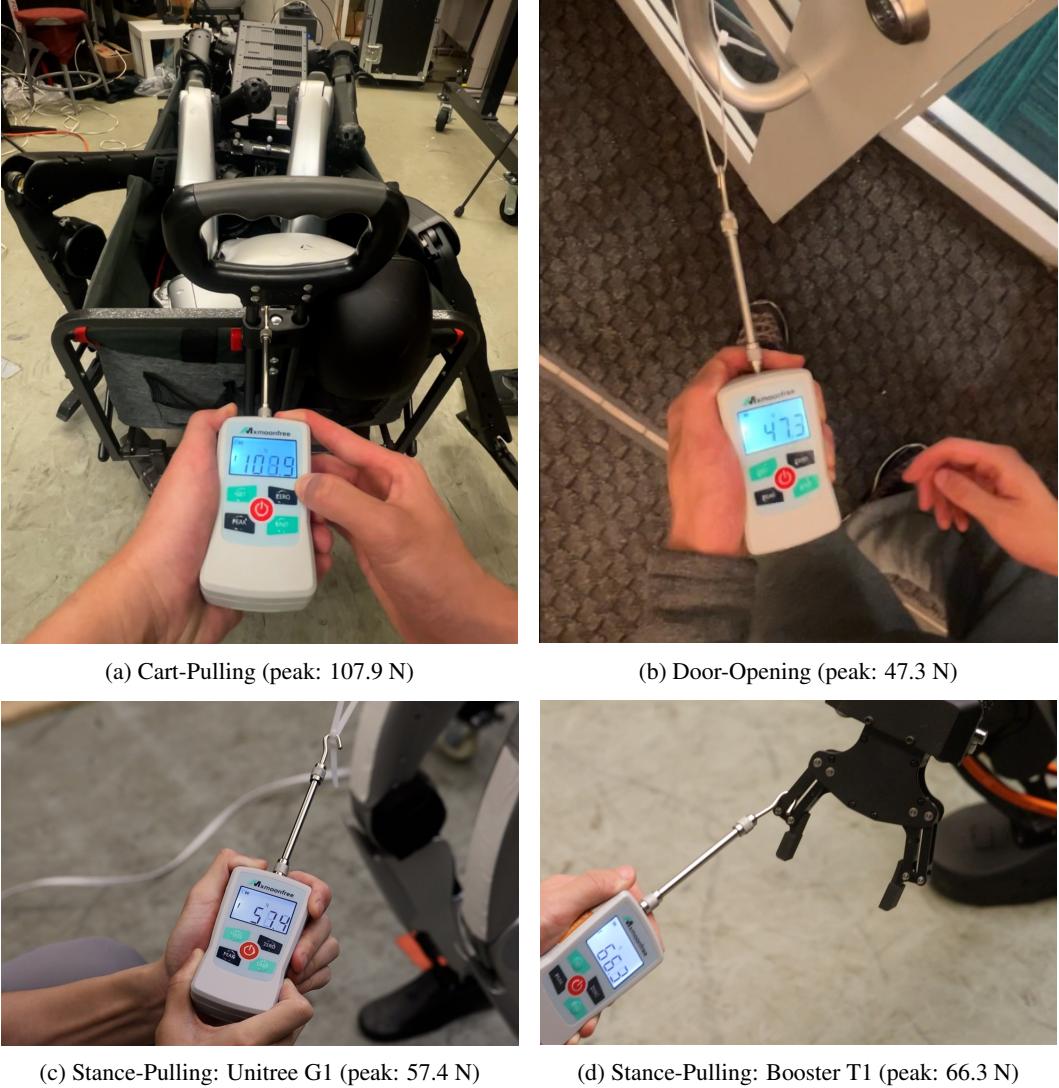


Figure 8: Maximum force readings captured during different force-adaptive tasks using a handheld force gauge. Subfigures (a)–(d) show peak force values during individual tasks.

the tote’s pose relative to the camera, we use FoundationPose [41], a state-of-the-art method for accurate and reliable 6-DoF pose estimation.

A.5.1 Perception - Pose Estimation

To set up the FoundationPose pipeline [41], we first acquire a high-fidelity 3D model of the industrial tote by performing a raw 3D scan, followed by manual post-processing in a 3D modeling tool. The resulting texture and .obj files serve as inputs to FoundationPose, enabling 6-DoF pose estimation of the tote from the G1 robot’s image stream. Additionally, we predefine grasp points longitudinally on the tote’s surfaces (Fig. 10), which are transformed into the robot base frame using the calibrated extrinsics between the camera and the robot.

A.5.2 Motion Capture System

The motion capture system provides accurate 6-DoF pose estimates—position (x, y, z) and orientation (yaw, pitch, roll)—for the robot base, pickup table, and drop table in a global reference frame, enabling consistent spatial localization across the system.

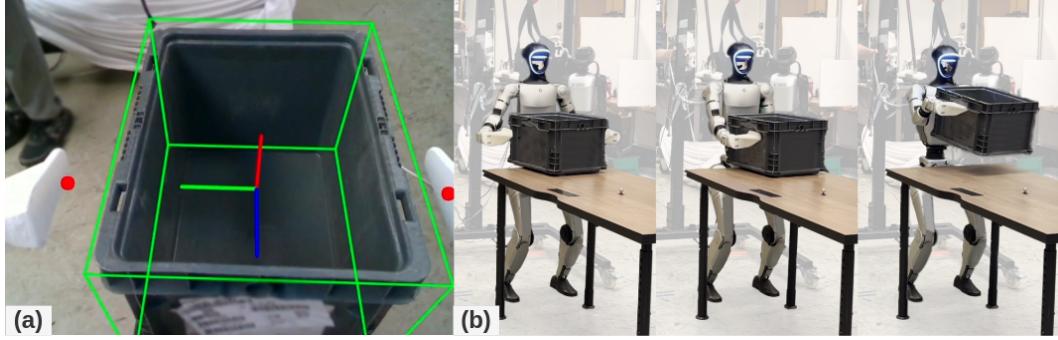


Figure 10: Humanoid G1 Tote Logistics (a) First person view of tote pose estimation (grasping points shown in red). (b) Sequence of actions from left to right- approach, grasp, pickup.

When the state machine transitions to “(2) Pick up the tote in stance” (see Fig. 9), triggered by mocap feedback, FoundationPose is executed in real time to estimate the tote pose. The predefined grasp points are then passed to an inverse kinematics (IK) solver, formulated as a go-to-pose problem for the upper-body manipulation.

A.6 Hardware Limits in Real-World Deployment

During sim-to-real deployment of the policy trained with FALCON, we observe that the humanoid robot struggles to sustain high joint torques over extended periods, often leading to rapid motor overheating—particularly at the wrists, as shown in Figure 11 (a). This significantly limits our ability to perform payload transport exceeding 2kg per arm at its default joint position. In contrast, as shown in Figure 11 (b) and (c), the same policy evaluated in MuJoCo [47], with torque clipping to respect joint limits but without modeling thermal constraints, successfully transports payloads over 3kg per end effector while accurately tracking the linear velocity commanded -1 m / s along the x-axis. This highlights a key gap between simulated and real-world actuator endurance.

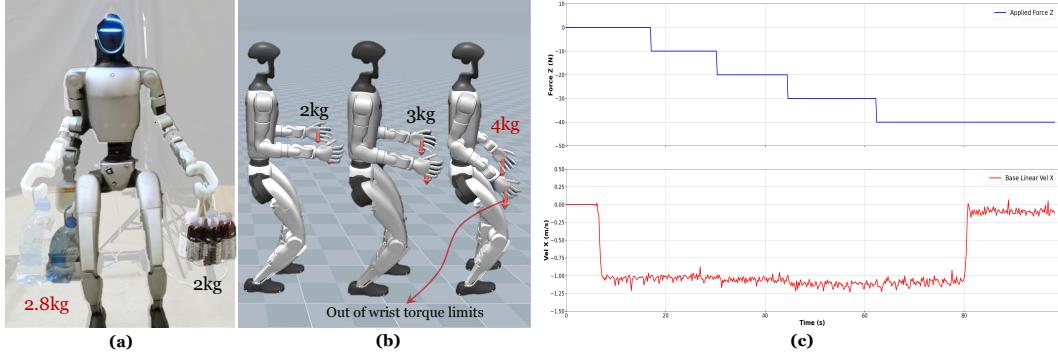


Figure 11: Transporting 0-4kg Payloads in Mujoco

However, for heavy-duty tasks such as cart-pulling—which require only brief bursts of high torque—the motors are less prone to overheating, as sustained high torque output is not necessary. This enables the robot to successfully perform such tasks in the real world.