arXiv:2512.14689v1 [cs.RO] 16 Dec 2025

CHIP: Adaptive Compliance for Humanoid Control through Hindsight Perturbation

Sirui Chen[1,2,†], Zi-Ang Cao[1,3,†], Zhengyi Luo[1], Fernando Castañeda[1], Chenran Li[1]
Tingwu Wang[1], Ye Yuan[1], Linxi "Jim" Fan[1], C. Karen Liu[2,‡], Yuke Zhu[1,3,‡]
[†] Co-First Authors [‡] Equal Advising
[1]NVIDIA [2]Stanford University [3]UT Austin

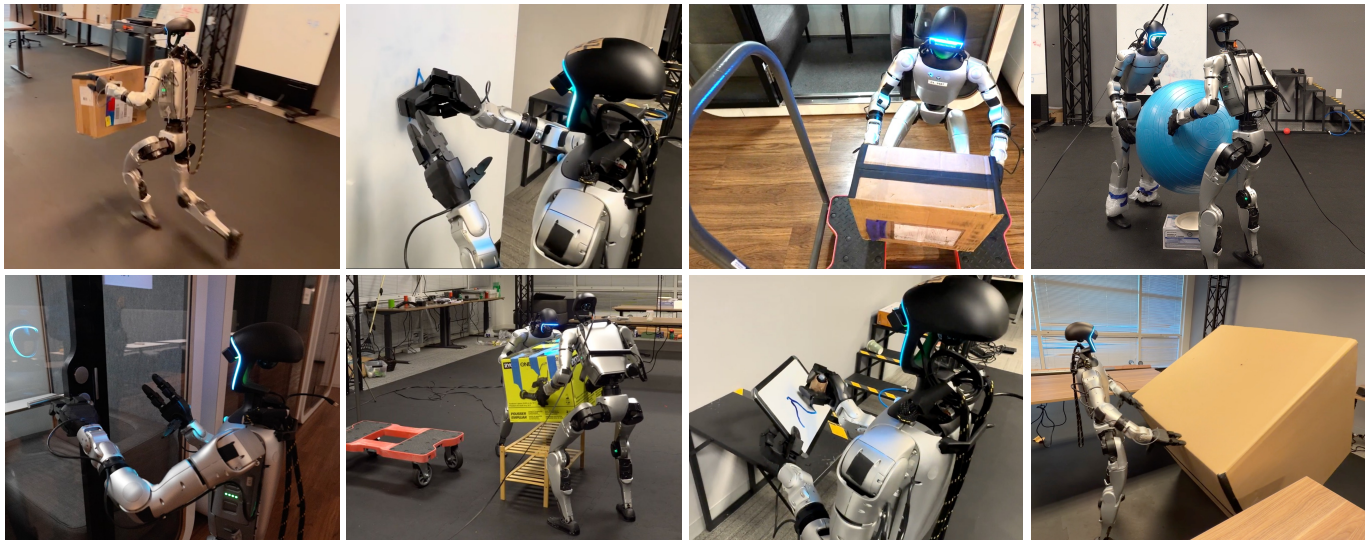

Fig. 1: CHIP enables humanoid robots to perform manipulation tasks that require force control, such as wiping a whiteboard, door opening, lifting boxes, flipping a large object, multi-robot collaboration, and writing. It also achieves both compliant and agile humanoid control, such as holding a box while running.

***Abstract*—Recent progress in humanoid robots has unlocked agile locomotion skills, including backflipping, running, and crawling. Yet it remains challenging for a humanoid robot to perform forceful manipulation tasks such as moving objects, wiping, and pushing a cart. We propose adaptive Compliance Humanoid control through hIsight Perturbation (CHIP), a plug-and-play module that enables controllable end-effector stiffness while preserving agile tracking of dynamic reference motions. CHIP is easy to implement and requires neither data augmentation nor additional reward tuning. We show that a generalist motion-tracking controller trained with CHIP can perform a diverse set of forceful manipulation tasks that require different end-effector compliance, such as multi-robot collaboration, wiping, box delivery, and door opening. For more details, please visit our website: https://nvlabs.github.io/CHIP/**

I. Introduction

Recent advances in humanoid robots have enabled impressive progress in agile locomotion skills, allowing robots to walk, run, backflip, and even dance with human-like fluidity. However, despite these achievements, humanoid robots remain limited in manipulation and can often interact only with lightweight objects. A key challenge in humanoid manipulation is controlling robots to apply consistent and controllable interaction forces. In traditional tabletop manipulators, this problem is typically addressed using a model-based compliance force controller, such as impedance or admittance control. Yet, the agility of modern humanoids primarily stems from reinforcement learning (RL) based controllers, which lack a generalizable formulation for compliant force control.

RL-based compliance controllers have shown promising results on quadruped robots [1, 2] by tracking motion spring-damper-like reference motion. However, these methods require substantial effort to generate large volumes of synthetic data that emulate spring-damper end-effector dynamics. Extending these approaches to humanoids is challenging, as it is difficult to ensure that such synthetic perturbation data remains within the distribution of natural human motion. In parallel, advances in RL-based motion tracking [3, 4, 5] have demonstrated that learning from human demonstrations is a scalable approach for constructing general-purpose humanoid controllers. However, these methods do not provide a principled way to integrate

This work was done during their internships of Sirui Chen and Zi-Ang Cao at NVIDIA GEAR Lab.

compliant or impedance control into humanoid tracking frameworks. In this work, we pose the central question: **Can we develop a scalable and generalizable approach for reconciling the high-gain stiffness required for humanoid agile motions with the variable compliance needed for safe contact-rich manipulation?** Some concurrent efforts [6, 7] attempt to extend FACET [2] to humanoids by augmenting retargeted human motion data with synthetic observations per responses under force perturbations, and then relying on tracking rewards to encourage imitation of the augmented trajectories. Yet, as is common in RL-based motion tracking [8, 9, 4, 10, 11], reward computation typically depends on rich reference information such as link poses, velocities, joint positions, and joint velocities, which are challenging to modify reliably during data augmentation.

Our key insight for tackling this problem is that reference motion can be interpreted, in hindsight, as the robot's compliant response to perturbations, allowing it to be used directly for dense reward computation without modification. Rather than altering the reference trajectory, we modify the sparse tracking goals as input obsesration to the policy, which are far easier to edit in a controlled manner. Specifically, we subtract the effect of a perturbation from the original reference motion to define the tracking goal in hindsight, while keeping the reference motion intact to provide high-quality, dense tracking rewards.

To this end, we introduce **C**ompliant **H**umanoid natural control through h**I**ndsight **P**erturbation (CHIP). This method integrates reinforcement learning with adaptive compliance control and can be directly incorporated into existing humanoid motion tracking frameworks with minimal modification. We demonstrate that generalist humanoid motion trackers trained with CHIP can perform a diverse range of forceful manipulation tasks—such as wiping, writing, cart pushing, and door opening while preserving their agility in tasks like dancing, running, and squatting. Furthermore, for tasks that require world-space coordination among multiple robots, we train a global 3-point tracking policy (i.e., tracking the head and two hands) using CHIP, enabling compliant multi-robot collaboration, including synchronized grasps and cooperative object transport. In summary, our core contributions are:

- CHIP, a generalizable and scalable plug-and-play module that enables adaptive compliance control in a humanoid tracking framework through hindsight perturbation.
- A compliant and natural local 3-point tracking controller that unlocks forceful whole-body teleoperation and autonomous Vision Language Action (VLA) policy learning, while maintaining the agile motion capabilities.
- A compliant global 3-point tracking controller that unlocks stable multi-robot object grasping and moving.

II. Related work

A. Variable and adaptive compliance manipulation

Adaptive compliance control has been used in manipulation to handle contact uncertainty and solve tasks that require force control. Prior work [12, 13] shows that variable impedance control could enable a robot to perform tasks that require significant contact force control, such as door opening, or to maintain surface contact, such as wiping. Under contact uncertainty, an adaptive compliance controller can also help robots perform stable grasps [14, 15]. Most prior adaptive compliance control studies use a tabletop arm with a model-based impedance controller. In our work, we draw inspiration from task design in these works, but apply it to a learned adaptive compliance controller on a humanoid robot.

B. Compliant control of legged robot

Recent success in legged locomotion and loco-manipulation benefits from large-scale reinforcement learning in simulations. RL motion tracking has become a pivotal task enabling general humanoid control. Policies trained purely for motion tracking, often inspired by frameworks such as DeepMimic [16], tend to be stiff and fragile under force perturbations. When the objective treats any deviation from a reference motion as an error that must be aggressively corrected, the robot generates large, uncontrolled forces during unexpected contact. To address this, recent research has pursued two distinct paradigms: learning to resist external forces or learning to comply with them.

Learning to Resist Disturbances. One line of work develops controllers designed to maintain tracking performance despite large, unknown external forces. These methods typically train policies with randomized external pushes to encourage robustness. Approaches that combine motion-tracking rewards with random external forces instruct the robot to adhere to the reference trajectory despite interaction forces [17, 18, 19]. A prominent example is FALCON [19], which uses a torque-aware 3D force curriculum, enabling the robot to gradually learn in simulation to resist strong disturbances and pull heavy carts. While effective for applications requiring forceful opposition, this stiffness-centric approach is ill-suited for contact-rich manipulation scenarios—such as robot-human collaboration, wiping, or handling fragile objects—which require the robot to yield in a controlled, spring-like manner.

Learning to Comply with Interactions. In contrast to force rejection, a second line of research aims to learn controllers that yield to external forces. One approach creates impedance behaviors "from scratch" using RL. For example, UniFP [1] and FACET [2] train unified policies to mimic target spring-mass-damper dynamics. UniFP learns a force estimator for admittance control, while FACET modulates policy stiffness as a control input, enabling tasks such as wiping using only proprioception. However, because they are primarily designed for quadruped robots with a single end-effector, scaling these methods to humanoids is challenging because generating synthetic compliance data that matches the distribution of natural human movement is difficult.

To sidestep the difficulty of discovering behaviors from scratch, prior work has attempted to build compliance into motion-tracking frameworks. SoftMimic [7] proposes a learning-from-examples framework, using an offline inverse kinematics (IK) solver to generate an augmented dataset of

stylistically compliant motions. An RL policy then learns to reproduce these pre-authored responses. Similarly, GentleHumanoid [6] augments keypoint reference motions with desired spring-damper dynamics during reward calculation. However, this creates an optimization conflict between motion tracking rewards and compliance objectives, often leading to degraded tracking accuracy or reduced agility. To make things worse, relying on offline data augmentation or reward tuning limits their applicability for scaling up to large, diverse motion datasets.

Our work builds upon these insights but proposes a more direct and scalable alternative. We introduce CHIP, a training recipe that integrates principled impedance-control objectives directly into the online RL loop via *hindsight perturbation*, without compromising tracking performance. CHIP obviates the need for the extensive synthetic interaction curriculum [1, 2] or the offline motion augmentation pipelines of SoftMimic [7] and GentleHumanoid [6]. By specifically targeting modifications in the input space, our method offers a simple yet powerful way to equip standard humanoid tracking systems with adaptive compliance, narrowing the gap between agile motion and safe, controllable physical interaction.

C. Humanoid control interfaces

Recent advances in reinforcement learning for humanoid whole-body control have produced remarkable results in agile locomotion [3, 20, 21, 22] and whole-body manipulation [17, 10, 23]. Extending motion-tracking approaches, several methods [17, 24] use human whole-body motion—encompassing both link- and joint-level states—to drive humanoid robot control. However, such control interfaces are complicated to use for both teleoperation and specifying high-level task commands. To simplify humanoid control, several methods [23, 10] adopt a decoupled upper-lower-body policy, in which the robot receives root-velocity and height commands for locomotion and upper-body target-motion commands for manipulation. Other approaches explore keypoint-based control [9, 8, 22, 11, 4, 25], enabling intuitive teleoperation by mapping head and wrist movements from a VR headset to humanoid motion. CHIP upgrades humanoid keypoint control with an adaptive-compliance module. It is compatible with both local keypoint tracking, as in SONIC [22], OmniH2O [9], and global keypoint tracking [25, 11], while additionally providing a plug-and-play module that directly integrates task-aware control into existing keypoint-based frameworks.

III. Method

Our method is a lightweight plug-and-play module that integrates seamlessly into any keypoint-based motion-tracking framework and enables adaptive whole-body control policies with end-effector-level compliance. As shown in Fig. 2, CHIP takes the 3-point tracking goal $\boldsymbol{g}_t$, end-effector compliance coefficients $\frac{1}{k}$, proprioception $\boldsymbol{s}_{t-10:t}$, and past actions $\boldsymbol{a}_{t-11:t-1}$ as input, and outputs actions that enables a humanoid robot to track the 3-point targets while yielding to external perturbations, with the desired level of compliance modulated by the continuous input coefficients.

The core of CHIP lies in modifying the training procedure within the motion tracking framework. During training, we apply a perturbation force $\boldsymbol{f}_i$ in a random direction to the end-effector i for a random period of time. The policy also receives compliance coefficients $\frac{1}{k}$ that allow it to follow an impedance control law, mimicking spring–damper dynamics under external disturbances.

Prior work [1, 2] and recent concurrent work [6, 7] train policies $\pi(\boldsymbol{s}, \boldsymbol{g})$ that observe tracking goals $\boldsymbol{g}$ from the original reference motion, but use tracking goals modified by perturbations, $\boldsymbol{g}_{\text{ptb}}$, to calculate the reward that encourages the robot to track the desired compliant response under force perturbations rather than the original motions (Fig. 3(b)):

$$r(\boldsymbol{s}, \boldsymbol{g}_{\text{ptb}}) = \exp(-||\boldsymbol{x}_{\text{eef}} - \boldsymbol{g}_{\text{ptb}}||), \qquad \boldsymbol{g}_{\text{ptb}} = \boldsymbol{g} + \frac{1}{k}\boldsymbol{f},$$

where $\boldsymbol{x}_{\text{eef}}$ is the current end-effector pose. This approach requires modifying the reference trajectory while maintaining its feasibility (Fig. 3(c)), a process that becomes increasingly difficult for dynamic skills such as running, where motion editing needs to satisfy both kinematic and dynamic constraints.

Instead, our method modifies the *observed tracking goals* as input to the policy in a hindsight manner,

$$\pi(\boldsymbol{s}, \boldsymbol{g}_{\text{hind\_ptb}}), \qquad \boldsymbol{g}_{\text{hind\_ptb}} = \boldsymbol{g} - \frac{1}{k}\boldsymbol{f},$$

such that the original goals are reinterpreted as the perturbation's resulting motion and are directly used to compute the reward function, $r(\boldsymbol{s}, \boldsymbol{g})$ (Fig. 3(a)), eliminating the need to alter the reference motion.

This design has several advantages:

- It takes less effort to modify the sparse goal in the observation than the dense reference motion in reward computation, allowing such modification to be done online during training.
- Robots are always encouraged to produce motion in the distribution of reference motion, which improves motion naturalness.
- Robots are exposed to observations from both the motion dataset and outside it, thereby improving policy robustness at test time.

This formulation streamlines training while enhancing robustness and agility. Unlike methods such as [1, 2], which explicitly learn an external force estimator, our approach uses a single policy model optimized solely with the PPO objective. The policy implicitly acquires force awareness through proprioceptive observations and past actions. To further improve sensitivity to external perturbations during training, we provide the critic with the ground-truth external force as a privileged observation. In addition, we supply both the actor and critic with a 10-step history of proprioception and past actions, allowing the policy to infer perturbation-related information from noisy observations. At test time, the policy only receives the tracking goal $\boldsymbol{g}$ and proprioception. It implicitly estimates

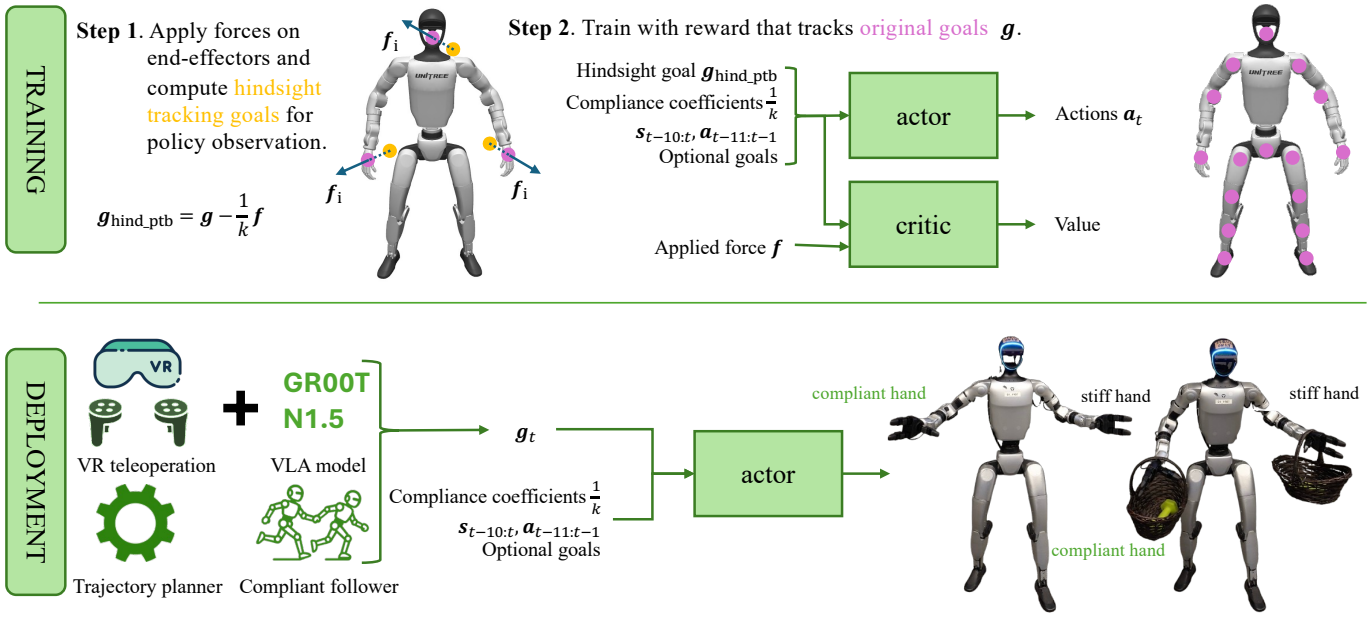

Fig. 2: Overview of training and deployment of CHIP: this general formulation takes tracking targets, end-effector compliance coefficients, and state history as default inputs. For the local tracking policy, we provide an additional lower-body reference pose, generated via the kinematic planner used in SONIC [22]. It adapts end-effector compliance based on the input compliance coefficient and force implicitly estimated from proprioception history. During training, the policy observes recovered keypoint poses by removing the hindsight perturbation, and tracking rewards are calculated from the original reference motion. The trained adaptive compliance policy could be controlled via multiple interfaces, such as VR teleoperation, a kinematic planner, and a vision-language-action (VLA) model.

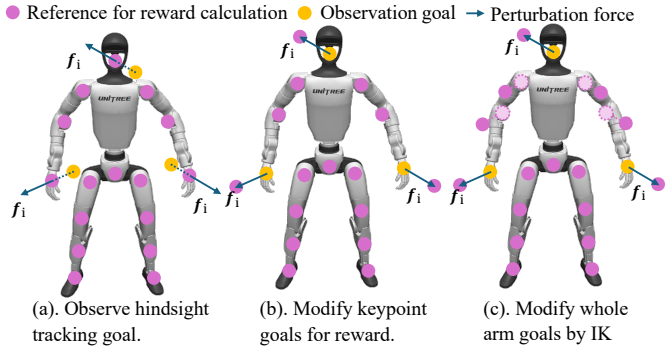

Fig. 3: Different formulations for adaptive compliant tracking policy training. (a) Ours (CHIP), the policy observes hindsight tracking goal, a modified end-effectors goal in the opposite direction of the perturbation force. (b) Modify keypoint goals for reward as in GentleHumanoid [6]. (c) Edit entire kinematic reference motion to account for end-effector perturbation as in SoftMimic [7].

the perturbation force and generates actions that counteract disturbances according to the commanded compliance. To enable compliant following, similar to FACET [2], we also implement a damper model on the target end-effector pose. Under an applied force, the end-effector pose x_{eef} deviates from the previous target g_{t-1}; we update the target using:

$$g_t = \alpha x_{\text{eef}} + (1 - \alpha) g_{t-1}.$$

This update allows the robot to adjust its target in response to force-induced deviations smoothly.

IV. Applications

We showcase two applications that utilize our compliance humanoid control: multi-humanoid manipulation and compliant teleoperation. The former application requires 3-point tracking in the global frame, and the latter depends only on a local tracking policy.

A. Compliant multi-robot grasping

For tasks requiring global coordination, such as multi-robot collaborative grasping or object transport, we train a whole-body control policy that tracks the 6D pose of the head and the 3D positions of both wrists in the world frame:

$$g = \{p^{\text{ref}}_{\text{head}}, r^{\text{ref}}_{\text{head}}, p^{\text{ref}}_{\text{lwrist}}, p^{\text{ref}}_{\text{rwrist}}\}.$$

This target space provides a simple interface for commanding robot end-effectors to reach desired locations in the global coordinate frame. Training uses a global tracking reward that penalizes the 3-point deviation between the robot and the reference motion in the world frame. However, because 3-point global poses provide only sparse information, it remains ambiguous for the robot to determine whether to walk or extend its arm and head when the 3-point target moves. We use 0.2 seconds of future information (5 frames, four frame skip per step) to resolve such ambiguity.

With an adaptive, compliant global 3-point controller, multiple robots can position their end-effectors in the world frame and apply desired forces, enabling multi-robot collaborative grasping and the movement of large objects beyond the grasping capability of a single robot.

To obtain a two-robot grasp, we extend the dexterous compliant grasp generation algorithm SpringGrasp [14] to a multi-humanoid collaborative grasp setting. We solve an optimization problem to obtain the 6D head poses and 3D wrist positions of two robots:

$$\{\boldsymbol{p}^i_{\text{lwrist}}, \boldsymbol{p}^i_{\text{rwrist}}, \boldsymbol{p}^i_{\text{head}}, \boldsymbol{r}^i_{\text{head}}\}, \qquad i \in \{1, 2\},$$

ensuring a stable, collision-free, compliant grasp.

We then plan a minimum-curvature trajectory between the current and target poses, interpolating wrist targets along the path. The grasping procedure consists of three phases:

1) **Approach**: robots follow the trajectory to pre-grasp poses near the object.
2) **Grasp**: wrist targets penetrate slightly into the object to establish a compliant grasp.
3) **Lift**: head and wrist targets move upward to lift the object collaboratively.

After the object has been lifted, we can simultaneously translate the 3-point target poses of both robots to move the grasped large object. For multi-robot collaboration to work in diverse environments, we require high-frequency, infrastructure-free robot localization. Following BeyondMimic [3], we combine:

- Foot-odometer based root velocity estimation at 50 Hz,
- Fast-LIO2 based pose updates at 10 Hz [26],
- Root-IMU angular velocity at 50 Hz,

yielding a fused 50 Hz root state estimate.

Both robots also need to share a consistent global frame. We first built an environment point map using Fast-LIO2. During initialization, each robot registers its live point cloud to the prebuilt map, aligning its coordinate frames.

B. Teleoperation and vision language action model training

1) Local 3-point tracking policy: For tasks that do not require global coordination in the task space, we train a local policy that tracks the SE(3) poses of the head and wrists relative to the robot's root frame, while also tracking lower-body joint positions and velocities to prevent drift in the absence of global information. Such a formulation acts as a decoupled policy that separates upper-body movement from lower-body locomotion at the command level.

$$\boldsymbol{g} = \{\boldsymbol{p}^{\text{ref}}_{\text{head}}, \boldsymbol{r}^{\text{ref}}_{\text{head}}, \boldsymbol{p}^{\text{ref}}_{\text{lwrist}}, \boldsymbol{r}^{\text{ref}}_{\text{lwrist}}, \boldsymbol{p}^{\text{ref}}_{\text{rwrist}}, \boldsymbol{r}^{\text{ref}}_{\text{rwrist}}, \boldsymbol{q}^{\text{ref}}_{\text{lower}}\}.$$

The reward minimizes the difference between the reference and robot 3-point poses expressed in the root frame. At deployment time, desired 3-point poses are provided either through a VR-based teleoperation system or via rollouts from a Vision-Language-Action (VLA) policy. Lower-body trajectories are generated by a kinematic planner that receives commands for root velocity, heading, and height.

For tasks that do not require global localization, the compliant local 3-point policy is straightforward to deploy. Following SONIC [22], the robot is teleoperated using VR-tracked wrist poses along with root velocity and height commands from VR joysticks as shown in Fig. 4. Lower-body control is handled by a kinematic planner driven by these commands.

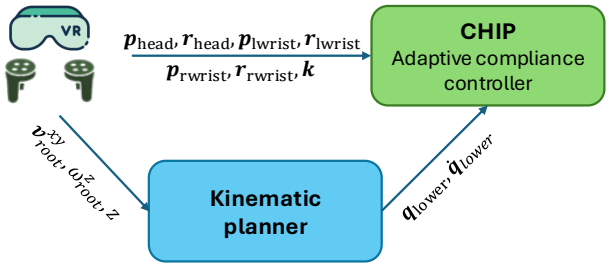

Fig. 4: Teleoperation interface, VR directly gives wrist and head pose commands to the whole body controller, root commands are sent to the kinematic planner to generate lower body joint motion.

The kinematic planner takes root velocity, angular velocity, and height as commands and outputs planned joint positions and velocities. Compliance coefficients for each end-effector are adjusted by the operator in VR. Teleoperation sessions collect paired action data and compliance coefficients from the teleoperation interface, as well as binocular RGB observations from the robot's egocentric OAK camera. These data are used to finetune the GR00T N1.5 [27] VLA model.

V. Experiments and Results

We conducted an extensive experiment to answer the following research questions.

- Q1: How well does CHIP track position and force targets?
- Q2: What task(s) do adaptive compliance unlock through teleoperation?
- Q3: How does CHIP facilitate multi-robot collaboration?
- Q4: Can CHIP enable VLA models to perform autonomous forceful manipulation?

We compared CHIP with two baselines: 1) FALCON [19], a policy that does not observe compliance but was trained with force perturbation, and 2) No force, a standard tracking policy that neither observes compliance nor was trained with force perturbation.

A. Experiment setup

We used Unitree G1 as our testing platform, and an OAK-D W camera was mounted on the robot head to provide egocentric binocular vision. All policies were trained on 64 Nvidia L40S GPUs, with 4096 environments per GPU, for 4 days. During deployment, the local 3-point tracking policy, kinematic motion planner, and teleoperation server were running on Unitree G1 Jetson NX onboard computer with TensorRT acceleration. The global 3-point tracking policy, together with the state estimator, was running on a desktop machine with an i7-13700K CPU and an RTX 3090 GPU. We use the same desktop computer to run VLA policies.

B. Tracking performance of CHIP

We analyze the tracking accuracy and the robot's response to force perturbations under different end-effector compliance settings. Fig. I shows the tracking accuracy measured over 100 trajectories sampled from the TWIST dataset [17]; it shows that the use of hindsight perturbation training yields

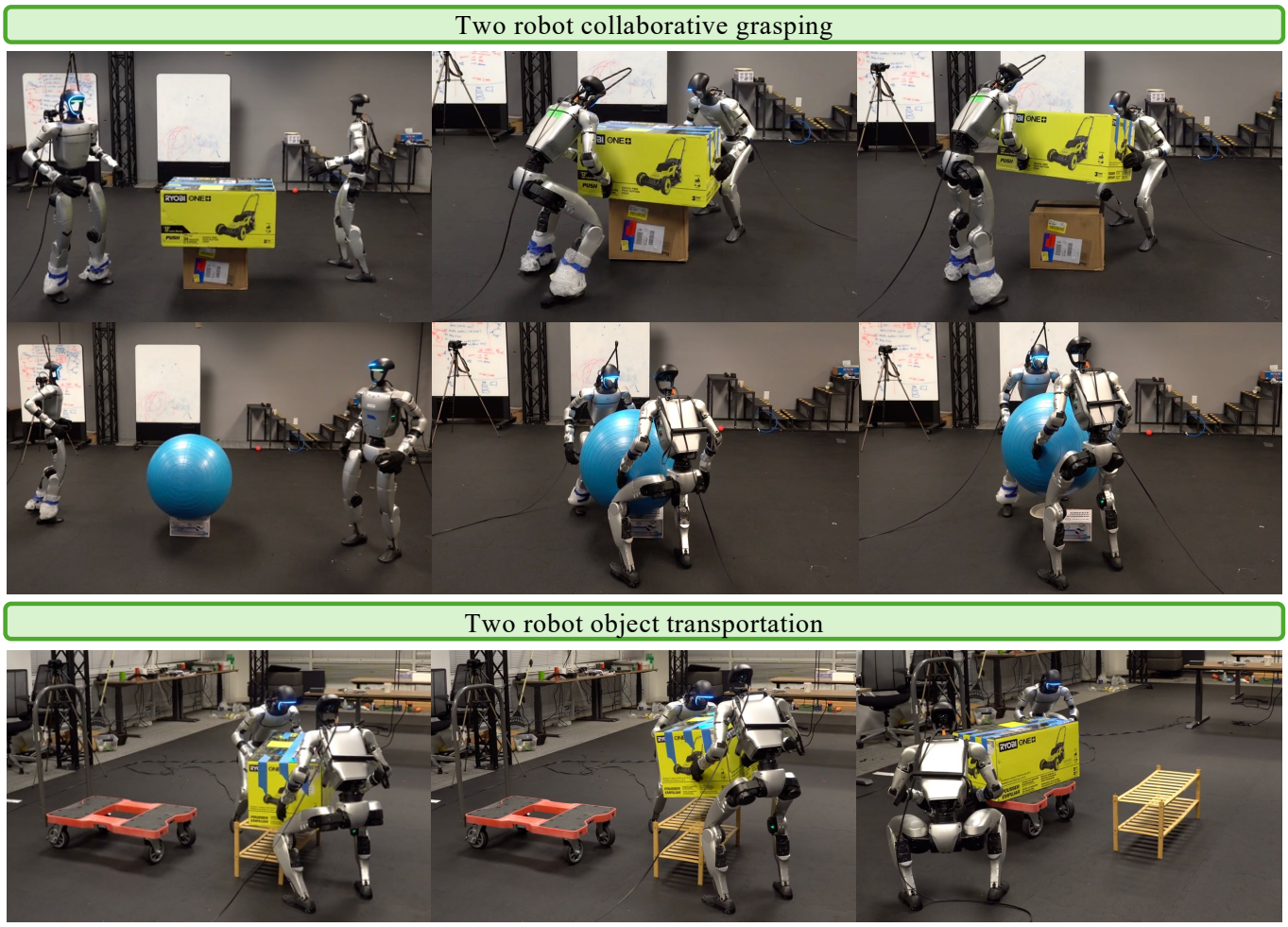

Fig. 5: Multi-robot grasp and object transportation: In the grasping experiment, two robots walk toward the object, grasp and lift the object with their end-effectors; Two robots can also move the grasped object and place it in a new location.

Method / Error	ϵ_p^g **[m]**	ϵ_r^g **[rad]**	ϵ_p^l **[m]**	ϵ_r^l **[rad]**
Ours ($\frac{1}{k}$ = 0.0)	0.08 (0.04)	0.09 (0.04)	0.02 (0.02)	0.09 (0.06)
Ours ($\frac{1}{k}$ = 0.02)	0.08 (0.05)	0.08 (0.05)	0.02 (0.02)	0.10 (0.06)
Ours ($\frac{1}{k}$ = 0.05)	0.08 (0.04)	0.08 (0.04)	0.02 (0.02)	0.11 (0.06)
FALCON	0.09 (0.04)	0.08 (0.04)	0.02 (0.02)	0.08 (0.05)
No perturb force	0.11 (0.04)	0.09 (0.05)	0.02 (0.02)	0.08 (0.05)

TABLE I: Tracking errors are measured in the absence of external forces for our method under three distinct evaluation-time settings, and compared against two baseline approaches. Entries are reported as mean (standard deviation). Metrics: ϵ_p^g – global position error [m], ϵ_r^g – global orientation error [rad], ϵ_p^l – local position error [m], and ϵ_r^l – local orientation error [rad].

position tracking on par with the no-compliance and no-force-perturbation baseline, and only slightly affects orientation tracking performance. Meanwhile, the results show that varying end-effector stiffness does not significantly affect tracking performance, indicating that the policy can distinguish between the joint's internal driving force required for agile performance and the external perturbation force. We also measured how the compliance coefficient affects the robot's responses to force perturbations. Fig. 6 shows that for CHIP, displacement increases linearly while increasing the compliance coefficient. For the baseline, which naively adds a reward for tracking the modified end-effector trajectory, the robot is less responsive to changes in the compliance coefficient because of conflicts among the tracking reward terms.

C. Multi-robot grasping and object transportation

To demonstrate the effectiveness of our adaptive compliance controller for multi-robot collaborative grasp, we evaluated our method on the collaborative grasping of two objects of different heights, as shown in Fig. 5. Robots first approached and grasped the object with their end-effectors, and the grasp was considered successful if they could lift it. Tab. II shows the grasping success rate. Overall, using an adaptive compliance controller achieves an average grasp success rate of 80%, which is +75% higher than that of an always-stiff controller and +40% higher than that of a controller without force perturbation. In failure cases with our method, the main reason was the robot's knee colliding with the object before grasping,

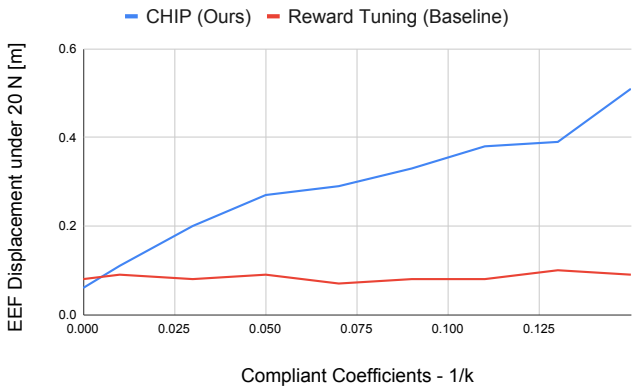

Fig. 6: EEF displacement under 20 N external force in global tracking setting. Reward Tuning: baseline that adds a modified end-effector tracking reward but removes the hindsight perturbation mechanism.

which did not fully showcase our controller's capabilities. We noticed that an always-stiff controller performed worse than the controller trained without force, because a stiff controller usually applied a much larger force to push the object, creating a large opposing force between the two robots and the object, causing the object to destabilize immediately after the contact. However, thanks to the intrinsic compliance, the controller without force perturbation could still grasp the object from time to time. We also show that after grasping the object, robots can stably move the object and place it in a new location, as shown in Fig. 5. We used the keyboard command to translate end-effector target locations to move the object.

Setting / Method	Ours	FALCON(No compliance)	No force perturbation
Box - 18 cm	0.6	0.0	0.4
Box - 35 cm	0.8	0.0	0.4
Sphere - 12 cm	1.0	0.2	0.4
Sphere - 35 cm	0.8	0.0	0.4
Average	0.8	0.05	0.4

TABLE II: Multi-robot grasp success rate; all settings were evaluated over five experiments.

D. Teleoperation

We demonstrated that the compliant local 3-point tracking controller can be teleoperated to accomplish a wide variety of contact-rich manipulation tasks, as illustrated in Fig. 7. The same controller supports both compliance-demanding tasks—such as lifting, transporting boxes, and wiping surfaces-as well as force-demanding tasks—such as opening a spring-loaded door, actuating a gantry, or flipping heavy boxes.

We further show that it can complete tasks requiring different compliance settings on both hands, for example, holding a small whiteboard rigidly with one hand (stiff mode) while writing on it with the other (compliant mode). The user can also adjust compliance online during long-horizon tasks, such as first opening a marker in stiff mode and then writing on the whiteboard in compliant mode. Additional results are shown in our video and on our website.

E. VLA inference

Using data collected from teleoperation, we fine-tuned a VLA policy to perform tasks autonomously. For the large whiteboard wiping task, we collected 400 teleoperated trajectories. During data collection, the compliance coefficients of the wiping hand were set to 0.05. After fine-tuning, the robot was able to complete the task autonomously, as shown in Fig. 7. We evaluated the policy over 10 autonomous rollouts. A rollout was considered successful if the robot erased all text on the whiteboard within 2 minutes. Under this criterion, the policy achieved a 60% success rate. The dominant failure modes were (i) the text being occluded by the robot hand and (ii) the text leaving the field of view of the robot-mounted camera.

We also demonstrate that the VLA model can exploit different stiffness settings for different end-effectors. As illustrated in Fig. 7, the hand holding the small whiteboard was commanded with high stiffness, while the wiping hand used lower stiffness to maintain compliant contact. Using 200 teleoperated trajectories for this bimanual setup, we fine-tuned a separate VLA policy that enables the robot to autonomously hold and wipe the whiteboard, achieving an 80% success rate.

VI. Conclusion

We introduced CHIP, a framework for learning humanoid natural adaptive compliance through hindsight replay. CHIP serves as a lightweight plug-and-play module for keypoint-based humanoid motion-tracking systems, enabling end-effector-level adaptive compliance control with minimal modification to existing pipelines. We demonstrated that our method produces adaptive local compliance policies capable of executing force-demanding tasks such as wiping, door opening, and box carrying. Moreover, CHIP supports a global adaptive 3-point tracking controller that enables multiple humanoid robots to manipulate and transport large objects collaboratively. Due to limitations in the proprioceptive accuracy of the Unitree G1 platform, our current implementation employs unidirectional stiffness to specify end-effector compliance. Incorporating accurate wrist force–torque sensing would enable more precise and versatile impedance control, potentially unlocking fine-grained manipulation tasks such as peg-in-hole insertion. We leave these improvements to future work.

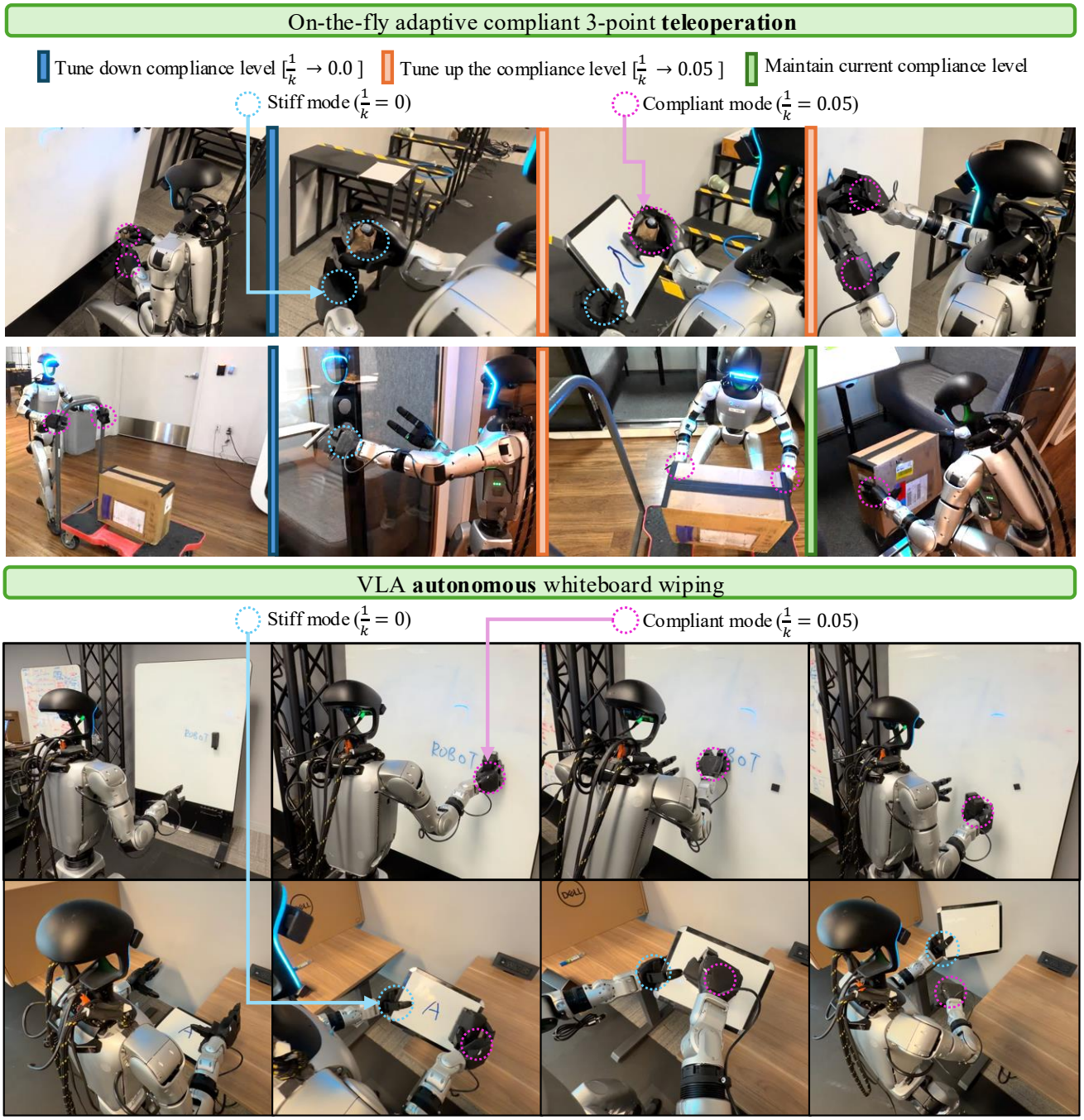

Fig. 7: On-the-fly adaptive compliant 3-point teleoperation: Hand compliance is annotated by $\frac{1}{k}$. Dashed cyan and magenta rings denote stiff mode ($\frac{1}{k} = 0$) and compliant mode ($\frac{1}{k} = 0.05$), respectively. Vertical bars indicate online compliance tuning commands (blue: tune down $\frac{1}{k} \rightarrow 0$; orange: tune up $\frac{1}{k} \rightarrow 0.05$; green: maintain). The unloading task benefits from compliant behavior during cart pushing and box grasping, but stiff behavior during door opening; the writing & wiping task alternates between stiff pen opening and bimanual hybrid compliance during writing and wiping (supporting hand stiff, writing/wiping hand compliant). **VLA autonomous whiteboard wiping:** A VLA policy autonomously approaches the board, grasps the wiper, and erases text, both on a wall-mounted whiteboard and while holding a small whiteboard.

References

[1] Peiyuan Zhi, Peiyang Li, Jianqin Yin, Baoxiong Jia, and Siyuan Huang. Learning unified force and position control for legged loco-manipulation. *arXiv preprint arXiv:2505.20829*, 2025.

[2] Botian Xu, Haoyang Weng, Qingzhou Lu, Yang Gao, and Huazhe Xu. Facet: Force-adaptive control via impedance reference tracking for legged robots. *arXiv preprint arXiv:2505.06883*, 2025.

[3] Qiayuan Liao, Takara E Truong, Xiaoyu Huang, Guy

Tevet, Koushil Sreenath, and C Karen Liu. Beyondmimic: From motion tracking to versatile humanoid control via guided diffusion. *arXiv preprint arXiv:2508.08241*, 2025.

[4] Yanjie Ze, Siheng Zhao, Weizhuo Wang, Angjoo Kanazawa, Rocky Duan, Pieter Abbeel, Guanya Shi, Jiajun Wu, and C. Karen Liu. Twist2: Scalable, portable, and holistic humanoid data collection system. *arXiv preprint arXiv:2511.02832*, 2025.

[5] Zhikai Zhang, Jun Guo, Chao Chen, Jilong Wang, Chenghuai Lin, Yunrui Lian, Han Xue, Zhenrong Wang, Maoqi Liu, Jiangran Lyu, et al. Track any motions under any disturbances. *arXiv preprint arXiv:2509.13833*, 2025.

[6] Qingzhou Lu, Yao Feng, Baiyu Shi, Michael Piseno, Zhenan Bao, and C Karen Liu. Gentlehumanoid: Learning upper-body compliance for contact-rich human and object interaction. *arXiv preprint arXiv:2511.04679*, 2025.

[7] Gabriel B Margolis, Michelle Wang, Nolan Fey, and Pulkit Agrawal. Softmimic: Learning compliant whole-body control from examples. *arXiv preprint arXiv:2510.17792*, 2025.

[8] Tairan He, Wenli Xiao, Toru Lin, Zhengyi Luo, Zhenjia Xu, Zhenyu Jiang, Changliu Liu, Guanya Shi, Xiaolong Wang, Linxi Fan, and Yuke Zhu. Hover: Versatile neural whole-body controller for humanoid robots. *arXiv preprint arXiv:2410.21229*, 2024.

[9] 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. *arXiv preprint arXiv:2406.08858*, 2024.

[10] Jialong Li, Xuxin Cheng, Tianshu Huang, Shiqi Yang, Rizhao Qiu, and Xiaolong Wang. Amo: Adaptive motion optimization for hyper-dexterous humanoid whole-body control. *Robotics: Science and Systems 2025*, 2025.

[11] Sirui Chen, Yufei Ye, Zi-Ang Cao, Jennifer Lew, Pei Xu, and C Karen Liu. Hand-eye autonomous delivery: Learning humanoid navigation, locomotion and reaching. *arXiv preprint arXiv:2508.03068*, 2025.

[12] Roberto Martín-Martín, Michelle A Lee, Rachel Gardner, Silvio Savarese, Jeannette Bohg, and Animesh Garg. Variable impedance control in end-effector space: An action space for reinforcement learning in contact-rich tasks. In *2019 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 1010–1017. IEEE, 2019.

[13] Yifan Hou, Zeyi Liu, Cheng Chi, Eric Cousineau, Naveen Kuppuswamy, Siyuan Feng, Benjamin Burchfiel, and Shuran Song. Adaptive compliance policy: Learning approximate compliance for diffusion guided control. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4829–4836. IEEE, 2025.

[14] Sirui Chen, Jeannette Bohg, and C Karen Liu. Springgrasp: Synthesizing compliant, dexterous grasps under shape uncertainty. *arXiv preprint arXiv:2404.13532*, 2024.

[15] Lai Wei, Xuanbin Peng, Ri-Zhao Qiu, Tianshu Huang, Xuxin Cheng, and Xiaolong Wang. Hmc: Learning heterogeneous meta-control for contact-rich loco-manipulation. *arXiv preprint arXiv:2511.14756*, 2025.

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

[17] Yanjie Ze, Zixuan Chen, JoÃĢo Pedro AraÃšjo, Ziang Cao, Xue Bin Peng, Jiajun Wu, and C Karen Liu. Twist: Teleoperated whole-body imitation system. *arXiv preprint arXiv:2505.02833*, 2025.

[18] Zipeng Fu, Xuxin Cheng, and Deepak Pathak. Deep whole-body control: Learning a unified policy for manipulation and locomotion. *arXiv preprint arXiv:2210.10044*, 2022.

[19] Yuanhang Zhang, Yifu Yuan, Prajwal Gurunath, Tairan He, Shayegan Omidshafiei, Ali-akbar Agha-mohammadi, Marcell Vazquez-Chanlatte, Liam Pedersen, and Guanya Shi. Falcon: Learning force-adaptive humanoid loco-manipulation. *arXiv preprint arXiv:2505.06776*, 2025.

[20] Weiji Xie, Jinrui Han, Jiakun Zheng, Huanyu Li, Xinzhe Liu, Jiyuan Shi, Weinan Zhang, Chenjia Bai, and Xuelong Li. Kungfubot: Physics-based humanoid whole-body control for learning highly-dynamic skills. *arXiv preprint arXiv:2506.12851*, 2025.

[21] Tong Zhang, Boyuan Zheng, Ruiqian Nai, Yingdong Hu, Yen-Jen Wang, Geng Chen, Fanqi Lin, Jiongye Li, Chuye Hong, Koushil Sreenath, et al. Hub: Learning extreme humanoid balance. *arXiv preprint arXiv:2505.07294*, 2025.

[22] Zhengyi Luo, Ye Yuan, Tingwu Wang, Chenran Li, Sirui Chen, Fernando Castaneda, Zi-Ang Cao, Jiefeng Li, David Minor, Qingwei Ben, Xingye Da, Runyu Ding, Cyrus Hogg, Lina Song, Edy Lim, Eugene Jeong, Tairan He, Haoru Xue, Wenli Xiao, Zi Wang, Simon Yuen, Jan Kautz, Yan Chang, Umar Iqbal, LinxiJimFan, and Yuke Zhu. Sonic: Supersizing motion tracking for natural humanoid whole-body control. 2025.

[23] Qingwei Ben, Feiyu Jia, Jia Zeng, Junting Dong, Dahua Lin, and Jiangmiao Pang. Homie: Humanoid loco-manipulation with isomorphic exoskeleton cockpit. *arXiv preprint arXiv:2502.13013*, 2025.

[24] Zixuan Chen, Mazeyu Ji, Xuxin Cheng, Xuanbin Peng, Xue Bin Peng, and Xiaolong Wang. Gmt: General motion tracking for humanoid whole-body control. *arXiv:2506.14770*, 2025.

[25] Yixuan Li, Yutang Lin, Jieming Cui, Tengyu Liu, Wei Liang, Yixin Zhu, and Siyuan Huang. Clone: Closed-loop whole-body humanoid teleoperation for long-horizon tasks. *arXiv preprint arXiv:2506.08931*, 2025.

[26] Wei Xu, Yixi Cai, Dongjiao He, Jiarong Lin, and Fu Zhang. Fast-lio2: Fast direct lidar-inertial odometry. *IEEE Transactions on Robotics*, 38(4):2053–2073, 2022.

[27] Johan Bjorck, Fernando Castañeda, Nikita Cherniadev, Xingye Da, Runyu Ding, Linxi Fan, Yu Fang, Dieter Fox, Fengyuan Hu, Spencer Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

A. Reward Design

Our tracking reward design is inspired by prior work [22, 3]. Tab. III shows our reward terms and weights.

Reward term	**Equation**	**Weight**
Tracking rewards $\mathcal{R}(\boldsymbol{s}_t^p, \boldsymbol{s}_t^g)$		
Root orientation	$r_{\text{ori}}^{\text{root}}(t) = \exp\big(-\|\boldsymbol{o}_{t,r}^p - \boldsymbol{o}_{t,r}^g\|_2^2/0.4^2\big)$	0.5
Body link pos (rel.)	$r_{\text{pos}}^{\text{body}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{p}_{t,b}^{p,\text{rel}} - \boldsymbol{p}_{t,b}^{g,\text{rel}}\|_2^2/0.3^2\Big)$	1.0
Body link ori (rel.)	$r_{\text{ori}}^{\text{body}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{o}_{t,b}^{p,\text{rel}} - \boldsymbol{o}_{t,b}^{g,\text{rel}}\|_2^2/0.4^2\Big)$	1.0
Body link lin. vel	$r_{\text{lin}}^{\text{body}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{v}_{t,b}^{p} - \boldsymbol{v}_{t,b}^{g}\|_2^2/1.0^2\Big)$	1.0
Body link ang. vel	$r_{\text{ang}}^{\text{body}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{\omega}_{t,b}^{p} - \boldsymbol{\omega}_{t,b}^{g}\|_2^2/3.14^2\Big)$	1.0
3-point pos (rel.)	$r_{\text{pos}}^{\text{3pt}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{p}_{t,b}^{p,\text{rel}} - \boldsymbol{p}_{t,b}^{g,\text{rel}}\|_2^2/0.3^2\Big)$	1.0
3-point ori (rel.)	$r_{\text{ori}}^{\text{3pt}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{o}_{t,b}^{p,\text{rel}} - \boldsymbol{o}_{t,b}^{g,\text{rel}}\|_2^2/0.4^2\Big)$	1.0
Global tracking only rewards $\mathcal{R}(\boldsymbol{s}_t^p, \boldsymbol{s}_t^g)$		
3-point pos (abs.)	$r_{\text{pos}}^{\text{3pt}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{p}_{t,b}^{p,\text{abs}} - \boldsymbol{p}_{t,b}^{g,\text{abs}}\|_2^2/0.3^2\Big)$	0.5
3-point ori (abs.)	$r_{\text{ori}}^{\text{3pt}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{o}_{t,b}^{p,\text{abs}} - \boldsymbol{o}_{t,b}^{g,\text{abs}}\|_2^2/0.4^2\Big)$	0.5
Penalty terms $\mathcal{P}(\boldsymbol{s}_t^p, \boldsymbol{a}_t)$		
Action rate	$r_{\text{act}}(t) = \|\boldsymbol{a}_t - \boldsymbol{a}_{t-1}\|_2^2$	-0.1
Joint limit	$r_{\text{jlim}}(t) = \sum_j \mathbb{1}[\boldsymbol{q}_{t,j} \notin [\boldsymbol{q}_{t,j}^{\min}, \boldsymbol{q}_{t,j}^{\max}]]$	-10.0
Undesired contacts	$r_{\text{contact}}(t) = \sum_{c\notin\{\text{ankles, wrists}\}} \mathbb{1}[\|\boldsymbol{F}_c\| > 1.0\ \text{N}]$	-0.1

TABLE III: Reward design details. $\mathcal{B}$: tracked body links.

B. Force perturbation

During training, we apply random force perturbation to the robot's end-effectors. We sample uniformly random force magnitude between 0 and 40 N for each perturbation in a uniformly random direction. For each perturbation, we draw a duration uniformly from 1 to 3 seconds. During the perturbation, the force magnitude changes following the schedule shown in Fig. 8. We design this pattern to mimic fundamental contact forces under impedance control, in which the force magnitude gradually increases to its peak at contact and then decreases to zero at break.

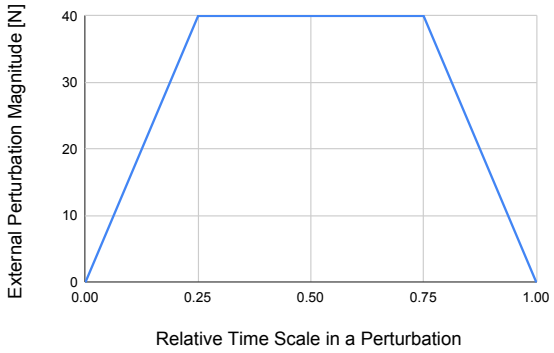

Fig. 8: Unified trapezoid profile of external force magnitude schedule within 1-3 seconds perturbation duration.