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 *

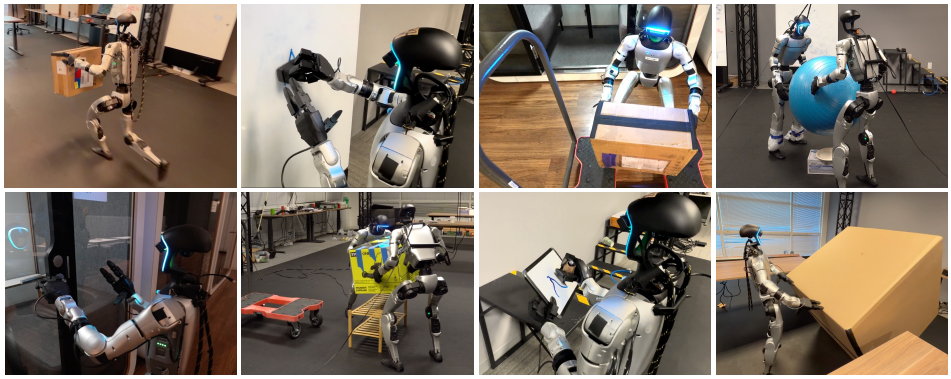


Figure 1: CHIP enables humanoid robots to perform manipulation tasks that require force control, such as wiping a whiteboard, door opening, lifting boxes, flipping a large object, multi-robot collaboration, and writing. It also achieves both compliant and agile humanoid control, such as holding a box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

Abstract: Despite impressive agility in locomotion, humanoid forceful manipulation remains limited by the lack of scalable, easy to train compliant force control in reinforcement learning (RL). We introduce CHIP (Compliant Humanoid natural control through hIndsight Perturbation), a plug-and-play framework that bridges this gap. Our core insight is that a demonstrated motion can be reinterpreted, in hindsight, as a compliant response to an external perturbation. By adjusting only sparse motion goals in response to perturbations while leaving the dense reference trajectories untouched, CHIP allows compliant interaction behaviors to emerge naturally without degrading motion tracking rewards. A single CHIP policy successfully executes forceful interaction tasks (e.g., wiping, cart pushing, door opening) alongside highly agile behaviors (e.g., dancing, running), and seamlessly extends to multi-agent cooperative transport. For more details, please checkout our website: `nvlabs.github.io/CHIP`

Keywords: Humanoid, Learned force control, Humanoid impedance controller

1 Introduction

Recent advances in humanoid robots have achieved impressive agility in locomotion, yet manipulation remains largely limited to lightweight objects due to a lack of controllable interaction forces. While traditional manipulators use model-based compliance (e.g., impedance control), modern reinforcement learning (RL) controllers—which drive humanoid agility—lack a generalizable formulation for compliant force control. Existing RL-based compliance methods for quadrupeds [1, 2] rely

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

10th Conference on Robot Learning (CoRL 2026), Austin TX, USA.

on synthetic spring-damper data that is difficult to scale to natural humanoid motion. Meanwhile, while RL-based motion tracking [3, 4] scales well using human demonstrations, these frameworks often fail under external perturbations. Recent attempts to bridge this gap [5, 6] rely on complex data augmentation that requires modifying dense reference information (e.g., link velocities and poses), which is notoriously difficult to implement reliably without degrading reward signals.

In this work, we ask: can compliant humanoid control emerge without modifying the underlying reference motion itself? Our key insight is that a demonstrated motion can be reinterpreted, in hindsight, as the compliant response to an external perturbation. Instead of modifying the dense reference trajectory used for reward computation, we modify only the sparse motion goals provided to the policy. This insight allows us to preserve the original reference motion distribution while modifying only policy conditioning. Specifically, given an observed perturbation, we subtract its effect from the original tracking targets to construct a hindsight-conditioned goal. Intuitively, this teaches the policy that the demonstrated motion already reflects a compliant response to external forces, rather than a rigid trajectory that must be exactly reproduced. Because the original reference trajectory remains untouched, we preserve stable high-quality motion tracking rewards while enabling compliant interaction behaviors to emerge naturally.

We introduce CHIP (Compliant Humanoid natural control through hIndsight Perturbation), a scalable plug-and-play framework for compliant humanoid control. CHIP enables humanoids to perform forceful interaction tasks—including wiping, cart pushing, and door opening with a single policy while preserving highly agile behaviors such as dancing and running. We further demonstrate that CHIP extends naturally to multi-agent settings, enabling synchronized grasping and cooperative object transport.

2 Related work

2.1 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 [7, 8] 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 [9, 10]. 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.

2.2 Compliant control of legged robot

Recent advancements in humanoid locomotion and loco-manipulation rely heavily on large-scale reinforcement learning (RL) for motion tracking. While frameworks inspired by DeepMimic [11] enable high-fidelity imitation, they often result in "fragile" policies that generate uncontrolled forces during unexpected contact. Current research addresses this through two primary paradigms: force resistance and compliant interaction.

Resistance and Robustness. One approach focuses on maintaining tracking performance despite external disturbances. By employing randomized force curricula, methods like FALCON [12] and others [13, 14] train robots to forcefully oppose perturbations or pull heavy loads. However, this stiffness-centric behavior is ill-suited for contact-rich tasks—such as human-robot collaboration or handling fragile objects—where the robot must yield to its environment.

Comply with Interaction. Conversely, recent work seeks to learn impedance behaviors. "From-scratch" methods like UniFP [1] and FACET [2] train policies to mimic spring-mass-damper dynamics via random perturbation dynamic sampling and use stiffness as a control input. While effective

for quadrupeds, scaling these to humanoids is hindered by the difficulty of generating synthetic compliance data that aligns with natural human movement.

To bridge this gap, concurrent frameworks like SoftMimic [6] and GentleHumanoid [5] integrate compliance into motion tracking via offline IK-augmented datasets. These methods often hard to scale up to augment thousands of trajectories for general tracking training and suffer from optimization conflicts between tracking accuracy and compliance.

In our proposed CHIP, impedance-control objectives is directly integrated into the online RL loop via hindsight perturbation. Unlike prior work [1, 2, 6, 5], CHIP eliminates the need for synthetic interaction curricula or offline augmentation. By modifying the observation space, our method equips standard humanoid tracking systems with adaptive compliance without compromising agility or tracking performance.

2.3 Humanoid control interfaces

Recent advances in reinforcement learning for humanoid whole-body control have produced remarkable results in agile locomotion [3, 15, 16, 17] and whole-body manipulation [13, 18, 19]. Extending motion-tracking approaches, several methods [13, 20] 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 [19, 18] 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 [21, 22, 17, 23, 24, 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 [17], OmniH2O [21], and global keypoint tracking [25, 23], while additionally providing a plug-and-play module that directly integrates task-aware control into existing keypoint-based frameworks.

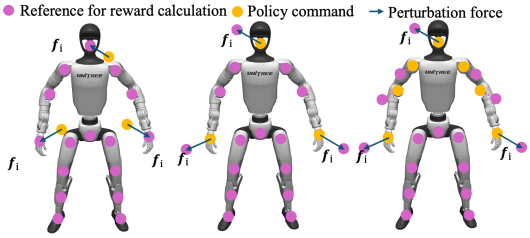


Figure 2: (a) CHIP: Modify hindsight tracking goal as **policy commands**; (b) GentleHumanoid [5]: Modify keypoint goals for reward computation; (c) SoftMimic [6]: Modify dense pose goals for reward computation.

3 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. 3, CHIP takes the 3-point tracking goal $\boldsymbol{g}_t$, desired 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 [5, 6] train policies $\pi(\boldsymbol{s}, \boldsymbol{g})$ that receive tracking goals $\boldsymbol{g}$ from the original reference motion, but modify the tracking goals, $\boldsymbol{g}_{\text{ptb}}$, in the reward calculation

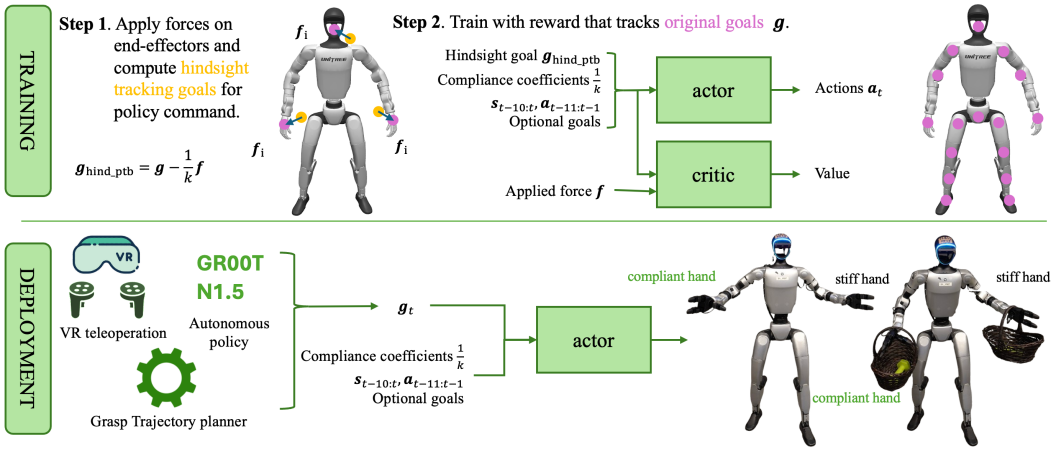


Figure 3: Overview of CHIP. During training, CHIP applies random perturbation forces to humanoid end-effectors and constructs hindsight tracking goals for policy conditioning. The actor receives hindsight goals, compliance coefficients, proprioceptive history, past actions, and optional task goals, while the critic additionally observes the applied perturbation force during training. At deployment, CHIP serves as a lightweight plug-and-play compliance module that can be integrated with diverse high-level controllers, including VR teleoperation, VLA models, and trajectory planners. By adjusting the compliance coefficients, different body parts can exhibit different levels of compliance during loco-manipulation tasks.

to reflect the effect of force perturbation (Fig. 2(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. 2(c)), a process that becomes increasingly more difficult for dynamic skills such as running, where motion editing needs to satisfy both kinematic and dynamic constraints.

Instead, CHIP modifies the *input tracking goals* 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 $\boldsymbol{g}$ are reinterpreted as the compliant responses and are directly used to compute the reward function, $r(\boldsymbol{s}, \boldsymbol{g}) = \exp(-||\boldsymbol{x}_{\text{eef}} - \boldsymbol{g}||)$ (Fig. 2(a)). Using the original goals $\boldsymbol{g}$ in reward eliminates the need to alter the reference motion, while the shifted input goal $\boldsymbol{g}_{\text{hind_ptb}}$ teaches the policy to infer and compensate for external disturbances.

This design has several advantages. First, because the policy receives goals that counteract the perturbation while being rewarded against the original reference motion, the optimal behavior becomes one that yields compliantly to external forces while remaining within the distribution of high-quality reference motions. As a result, CHIP enables compliant interaction behaviors without sacrificing agility or motion naturalness.

Second, CHIP enables a significantly simpler and more flexible training pipeline. Prior approaches require modifying dense reference motions for reward computation, which becomes increasingly difficult for dynamic humanoid skills where edited motions must remain kinematically and dynamically consistent. In contrast, CHIP modifies only the sparse goals provided as policy inputs. This modification is substantially easier because the policy inputs serve only as coarse conditioning signals, whereas the reward targets must remain accurate and physically consistent to provide stable learning supervision.

Departing from prior work that explicitly learns an external force estimator [1, 2], CHIP acquires force awareness directly from proprioceptive observations and action history. During training, both the actor and critic receive a 10-step history of proprioception and past actions, enabling the policy

to infer perturbation-related information from noisy observations. At test time, the policy receives only the tracking goal $\boldsymbol{g}$ and proprioceptive observations, yet still infers external disturbances and generates actions that react according to the commanded compliance.

4 Applications

We showcase two applications that utilize our humanoid compliance 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.

4.1 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:

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

This target space provides a simple interface for commanding robot end-effectors to reach desired locations in the global coordinate frame. 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 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 [9] to a multi-humanoid collaborative grasp setting. We solve an optimization problem to obtain the 6D head poses and 3D wrist positions as well as wrist stiffness of two robots:

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

ensuring a stable, collision-free, compliant grasp.

We then plan a minimum-curvature trajectory between the current and target poses, interpolating targets along the path. The grasping procedure consists of three phases: **Approach**: robots follow the trajectory to pre-grasp poses near the object. **Grasp**: wrist targets are set to establish a compliant grasp. **Transport**: head and wrist targets move simultaneously to lift and move the object. For multi-robot collaboration to work in diverse environments, we use infrastructure-free robot localization following BeyondMimic [3] with two robots sharing the same SLAM map.

4.2 Teleoperation and autonomous policy training

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 diffusion policy. Lower-body trajectories are generated by a kinematic planner that receives commands for root velocity, heading, and height.

Following SONIC [17], the robot is teleoperated using VR-tracked wrist poses along with root velocity and height commands from VR joysticks. Lower-body control is handled by a kinematic

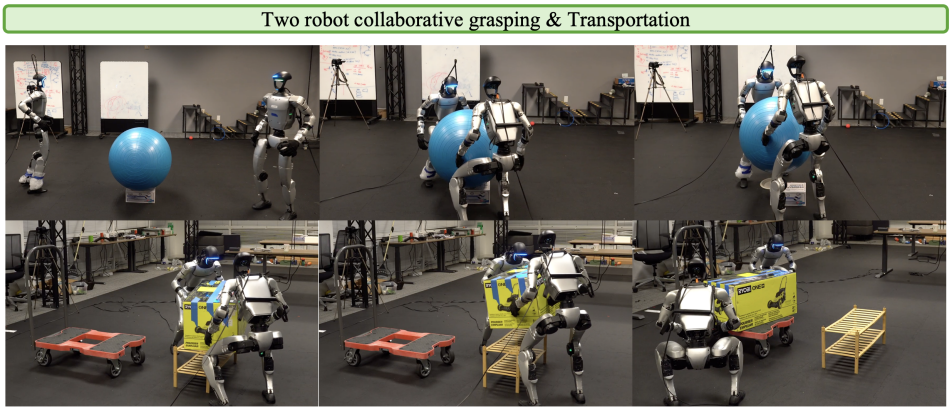


Figure 4: **Multi-robot grasp and object transportation:** In the grasping experiment, two robots walk toward the object, grasp and lift the object with their end-effectors; Two robots can also move the grasped object and place it in a new location.

planner driven by these commands. 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 with VR controller. 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 [26] model.

5 Experiments and Results

We conducted an extensive experiment to answer the following research questions. **Q1**: How well does CHIP track position and compliance commands? **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?

Method	ϵ_p^g **[m]**	ϵ_r^g **[rad]**	ϵ_p^l **[m]**	ϵ_r^l **[rad]**
Ours (0.0)	0.08 (0.04)	0.09 (0.04)	0.02 (0.02)	0.09 (0.06)
Ours (0.02)	0.08 (0.05)	0.08 (0.05)	0.02 (0.02)	0.10 (0.06)
Ours (0.05)	0.08 (0.04)	0.08 (0.04)	0.02 (0.02)	0.11 (0.06)
GH [stiff]	-	-	0.04 (0.03)	0.15 (0.08)
GH [comp]	-	-	0.05 (0.04)	0.16 (0.10)
FALCON	0.09 (0.04)	0.08 (0.04)	0.02 (0.02)	0.08 (0.05)
No force	0.11 (0.04)	0.09 (0.05)	0.02 (0.02)	0.08 (0.05)

Table 1: Mean(std) of end-effector tracking errors.

5.1 Experiment setup

We used the Unitree G1 with OAK-D W camera mounted on the robot's head to provide egocentric binocular vision. All policies were trained for 4 days on 64 Nvidia L40S GPUs, with 4096 environments per GPU; we used a derived Bones-Seed[17] dataset including 10,000 motion trajectories of locomotion and locomanipulation for tracking training. Any motion that includes wrist ground contacts is filtered out to prevent contact forces from confusing the CHIP policies. During deployment, the local 3-point tracking policy, kinematic motion planner, and teleoperation server were running on the 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.

5.2 Tracking performance of CHIP

We evaluated controller tracking performance across varying end-effector pose commands using 100 trajectories from the TWIST dataset [13], which are never scene during training. We compared the results with three baselines: FALCON [12]) which does not consider compliance, GentleHumanoid (GH) [5], and a vanilla baseline trained without hindsight force perturbation (No force). As shown in Tab. 1, our hindsight perturbation training yields global and local position tracking accuracy of

the end effector ($\epsilon_p^g, \epsilon_p^l$) comparable to FALCON and No force. Likewise, our method has a minimal impact on orientation tracking ($\epsilon_r^g, \epsilon_r^l$). Furthermore, our method consistently outperforms the local tracking only GentleHumanoid across all evaluated compliance levels(Stiff, Compliance).

Importantly, varying end-effector stiffness does not significantly degrade tracking accuracy (From 0.0 to 0.05), demonstrating the policy's ability to distinguish between internal driving forces and external perturbations. Unlike GH [5], which sacrifices dense joint-level tracking rewards to enable compliance, CHIP modifies observations via hindsight perturbation. This design choice allows CHIP to achieve superior general tracking accuracy alongside compliant behavior. Finally, we measure the actual compliance against commanded compliance of different methods. Fig. 5 illustrates stiffness tracking under a 10 N perturbation force; our method accurately tracks compliance commands from 5 to 20 mm/N, outperforming the SoftMimic baseline in regions of compliance overlap.

5.3 Multi-robot grasping and object transportation

We evaluated our adaptive compliance controller on multi-robot collaborative grasping of objects with varying heights, defining success as successfully lifting the object. As shown in Tab. 2, our method achieved an 80% average success rate, outperforming a no-compliance FALCON controller by 75% and a baseline without force perturbation by 40%. For the baselines, measured compliance was frozen to optimize only end-effector poses.

Notably, an always-stiff controller performed worse than the no-force baseline, as excessive pushing forces destabilized the object upon contact. Failures with our method primarily stemmed from pre-grasp collisions between the robot's knee and the object. Finally, we demonstrated stable post-grasp transportation and placement using keyboard-commanded end-effector targets (Fig. 4).

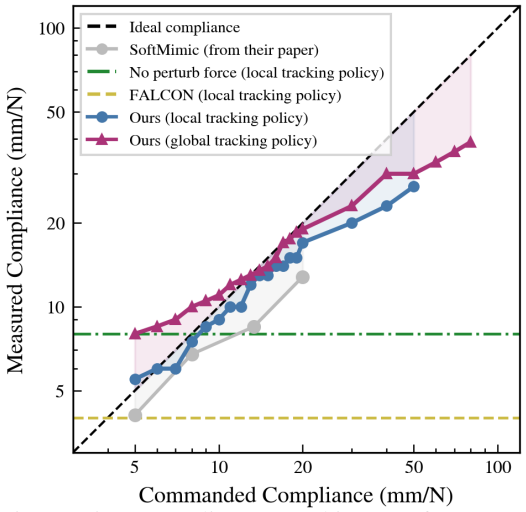


Figure 5: Compliance tracking performance. Green dashed line: measured constant compliance of no force perturbation baseline. Yellow dashed line: measured constant compliance of FALCON baseline

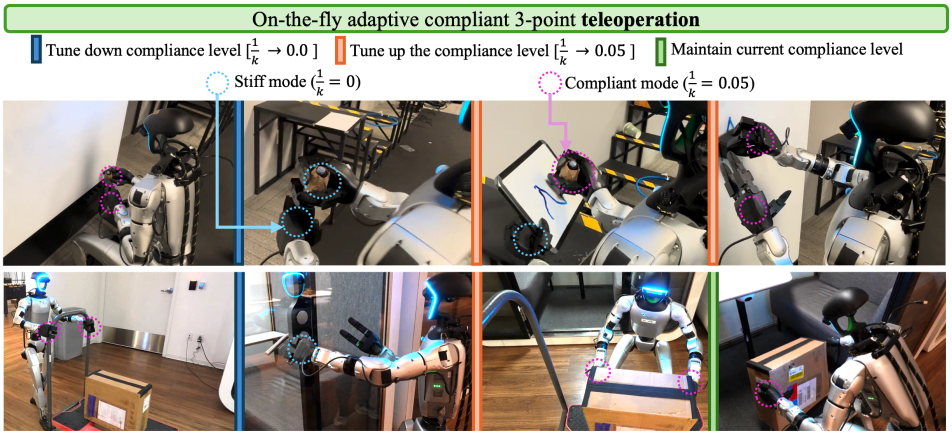


Figure 6: **(a) On-the-fly adaptive compliant teleoperation.** Bars indicate online compliance tuning commands (blue: $\frac{1}{k} \rightarrow 0$; orange: $\frac{1}{k} \rightarrow 0.05$; green: maintain). The unloading task benefits from compliant behavior during cart pushing and box grasping, but stiff behavior during door opening; the writing & wiping task alternates between stiff pen opening and compliant wiping.

5.4 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. 6. 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.

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

Table 2: Multi-robot grasp success rate; all settings were evaluated over five experiments.

We further show that it can complete tasks requiring different compliance settings on different hands, for example, holding a small whiteboard rigidly with one hand ($\frac{1}{k} = 0.001$) while writing on it with the other ($\frac{1}{k} = 0.05$). The user can also adjust compliance online during long-horizon tasks, such as first opening a marker with a stiff end-effector and then writing on the whiteboard in a compliant manner. Additional results are shown in our video and on our website.

5.5 Autonomous Policy

We further demonstrate that diffusion policies trained on compliant teleoperation data can autonomously execute contact-rich manipulation. Fig. 7 illustrates three evaluation tasks: (c) single-arm wiping requiring compliance in one end-effector; (a,b) bimanual holding and wiping requiring asymmetric stiffness; and (d) bimanual pick-and-place requiring bimanual compliance for grasp stability. Success rates were measured across 20 rollouts per task, with a 2-minute timeout indicating a failure.

Large whiteboard wiping. The robot approaches the whiteboard, grasps a wiper, and erases text using compliant contact. A VLA policy fine-tuned on 400 teleoperated trajectories (compliance coefficient 0.05) achieved a 60% success rate. Even during partial failures where some text remained, the robot maintained stable contact. The primary failure modes were hand occlusion and limited camera field of view.

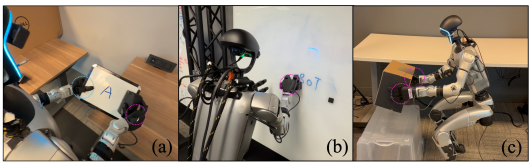


Figure 7: **Autonomous Tasks.**

Bimanual small whiteboard wiping. To test asymmetric, end-effector-specific stiffness, the robot rigidly holds a small whiteboard with its left hand while compliantly wiping with its right (Fig. 7). Fine-tuning a VLA policy on 200 teleoperated trajectories achieved an 80% success rate.

Box lifting. Demonstrating the benefits of bimanual compliance, the robot squats to retrieve a box, lifts it, turns 90°, and places it on a table (Fig. 7). Lower stiffness in both hands improves grasp stability during motion. A VLA policy fine-tuned on 200 teleoperated trajectories achieved a 90% success rate, with unsuccessful initial grasps being the primary failure mode.

6 Limitations

Due to limitations in the proprioceptive accuracy of the Unitree G1 platform, our real-robot implementation currently employs *unidirectional* stiffness to specify end-effector positional compliance. The lack of accurate wrist force torque sensing further limits precise and versatile admittance control and may hinder fine-grained contact tasks such as peg-in-hole insertion. In addition, for global tracking tasks, responsive and accurate global localization remains challenging, limiting the type of tasks our multirobot collaboration can achieve.

7 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, opening the door, and carrying boxes. Moreover, CHIP supports a globally adaptive compliant 3-point tracking controller, enabling multiple humanoid robots to manipulate and transport large objects collaboratively.

8 Acknowledgement

We'd like to thank Tairan He, Wenli Xiao, Haoru Xue, Qingzhou Lu, Yao Feng for insightful discussions and experimental support. We also thank Runyu Ding for help regarding VLA training. Finally, we'd like to thanks Jeremy Chimienti, Tri Cao, Justin Tran for helping with robot repair and data collection.

References

[1] P. Zhi, P. Li, J. Yin, B. Jia, and S. Huang. Learning unified force and position control for legged loco-manipulation. *arXiv preprint arXiv:2505.20829*, 2025.

[2] B. Xu, H. Weng, Q. Lu, Y. Gao, and H. Xu. Facet: Force-adaptive control via impedance reference tracking for legged robots. *arXiv preprint arXiv:2505.06883*, 2025.

[3] Q. Liao, T. E. Truong, X. Huang, G. Tevet, K. Sreenath, and C. K. Liu. Beyondmimic: From motion tracking to versatile humanoid control via guided diffusion. *arXiv preprint arXiv:2508.08241*, 2025.

[4] Z. Zhang, J. Guo, C. Chen, J. Wang, C. Lin, Y. Lian, H. Xue, Z. Wang, M. Liu, J. Lyu, et al. Track any motions under any disturbances. *arXiv preprint arXiv:2509.13833*, 2025.

[5] Q. Lu, Y. Feng, B. Shi, M. Piseno, Z. Bao, and C. K. Liu. Gentlehumanoid: Learning upper-body compliance for contact-rich human and object interaction. *arXiv preprint arXiv:2511.04679*, 2025.

[6] G. B. Margolis, M. Wang, N. Fey, and P. Agrawal. Softmimic: Learning compliant whole-body control from examples. *arXiv preprint arXiv:2510.17792*, 2025.

[7] R. Martín-Martín, M. A. Lee, R. Gardner, S. Savarese, J. Bohg, and A. 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.

[8] Y. Hou, Z. Liu, C. Chi, E. Cousineau, N. Kuppuswamy, S. Feng, B. Burchfiel, and S. 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.

[9] S. Chen, J. Bohg, and C. K. Liu. Springgrasp: Synthesizing compliant, dexterous grasps under shape uncertainty. *arXiv preprint arXiv:2404.13532*, 2024.

[10] L. Wei, X. Peng, R.-Z. Qiu, T. Huang, X. Cheng, and X. Wang. Hmc: Learning heterogeneous meta-control for contact-rich loco-manipulation. *arXiv preprint arXiv:2511.14756*, 2025.

[11] X. B. Peng, P. Abbeel, S. Levine, and M. Van de Panne. Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. *TOG*, 2018.

[12] Y. Zhang, Y. Yuan, P. Gurunath, T. He, S. Omidshafiei, A.-a. Agha-mohammadi, M. Vazquez-Chanlatte, L. Pedersen, and G. Shi. Falcon: Learning force-adaptive humanoid loco-manipulation. *arXiv preprint arXiv:2505.06776*, 2025.

[13] Y. Ze, Z. Chen, J. P. AraÃšjo, Z.-a. Cao, X. B. Peng, J. Wu, and C. K. Liu. Twist: Teleoperated whole-body imitation system. *arXiv preprint arXiv:2505.02833*, 2025.

[14] Z. Fu, X. Cheng, and D. Pathak. Deep whole-body control: Learning a unified policy for manipulation and locomotion. *arXiv preprint arXiv:2210.10044*, 2022.

[15] W. Xie, J. Han, J. Zheng, H. Li, X. Liu, J. Shi, W. Zhang, C. Bai, and X. Li. Kungfubot: Physics-based humanoid whole-body control for learning highly-dynamic skills. *arXiv preprint arXiv:2506.12851*, 2025.

[16] T. Zhang, B. Zheng, R. Nai, Y. Hu, Y.-J. Wang, G. Chen, F. Lin, J. Li, C. Hong, K. Sreenath, et al. Hub: Learning extreme humanoid balance. *arXiv preprint arXiv:2505.07294*, 2025.

[17] Z. Luo, Y. Yuan, T. Wang, C. Li, S. Chen, F. Castaneda, Z.-A. Cao, J. Li, D. Minor, Q. Ben, X. Da, R. Ding, C. Hogg, L. Song, E. Lim, E. Jeong, T. He, H. Xue, W. Xiao, Z. Wang, S. Yuen, J. Kautz, Y. Chang, U. Iqbal, LinxiJimFan, and Y. Zhu. Sonic: Supersizing motion tracking for natural humanoid whole-body control. 2025. URL https://api.semanticscholar.org/CorpusID:282922163.

[18] J. Li, X. Cheng, T. Huang, S. Yang, R. Qiu, and X. Wang. Amo: Adaptive motion optimization for hyper-dexterous humanoid whole-body control. *Robotics: Science and Systems 2025*, 2025.

[19] Q. Ben, F. Jia, J. Zeng, J. Dong, D. Lin, and J. Pang. Homie: Humanoid loco-manipulation with isomorphic exoskeleton cockpit. *arXiv preprint arXiv:2502.13013*, 2025.

[20] Z. Chen, M. Ji, X. Cheng, X. Peng, X. B. Peng, and X. Wang. Gmt: General motion tracking for humanoid whole-body control. *arXiv:2506.14770*, 2025.

[21] 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.

[22] 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.

[23] S. Chen, Y. Ye, Z.-A. Cao, J. Lew, P. Xu, and C. K. Liu. Hand-eye autonomous delivery: Learning humanoid navigation, locomotion and reaching. *arXiv preprint arXiv:2508.03068*, 2025.

[24] Y. Ze, S. Zhao, W. Wang, A. Kanazawa, R. Duan, P. Abbeel, G. Shi, J. Wu, and C. K. Liu. Twist2: Scalable, portable, and holistic humanoid data collection system. *arXiv preprint arXiv:2511.02832*, 2025.

[25] Y. Li, Y. Lin, J. Cui, T. Liu, W. Liang, Y. Zhu, and S. Huang. Clone: Closed-loop whole-body humanoid teleoperation for long-horizon tasks. *arXiv preprint arXiv:2506.08931*, 2025.

[26] J. Bjorck, F. Castañeda, N. Cherniadev, X. Da, R. Ding, L. Fan, Y. Fang, D. Fox, F. Hu, S. Huang, et al. Gr00t n1: An open foundation model for generalist humanoid robots. *arXiv preprint arXiv:2503.14734*, 2025.

8.1 Reward Design

Our tracking reward design is inspired by prior work [17, 3]. Tab. 3 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 onlyw rewards $\mathcal{R}(\boldsymbol{s}_t^p, \boldsymbol{s}_t^g)$		
3-point pos (abm.)	$r_{\text{pos}}^{\text{3pt}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{p}_{t,b}^{p,\text{abm}} - \boldsymbol{p}_{t,b}^{g,\text{abm}}\|_2^2/0.3^2\Big)$	0.5
3-point ori (abm.)	$r_{\text{ori}}^{\text{3pt}}(t) = \exp\Big(-\frac{1}{\lvert\mathcal{B}\rvert}\sum_{b\in\mathcal{B}}\|\boldsymbol{o}_{t,b}^{p,\text{abm}} - \boldsymbol{o}_{t,b}^{g,\text{abm}}\|_2^2/0.4^2\Big)$	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	$r_{\text{jlim}}(t) = \sum_j \mathbf{1}[\boldsymbol{q}_{t,j} \notin [\boldsymbol{q}_{t,j}^{\min}, \boldsymbol{q}_{t,j}^{\max}]]$	-10.0
Undesired contacts	$r_{\text{contact}}(t) = \sum_{c\notin\{\text{ankles, wrists}\}} \mathbf{1}[\|\boldsymbol{F}_c\| > 1.0\ \text{N}]$	-0.1

Table 3: Reward design details. $\mathcal{B}$: tracked body links.

8.2 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.

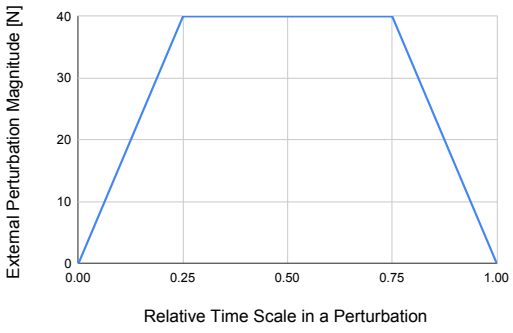


Figure 8: Unified trapezoid profile of external force magnitude schedule within 1-3 seconds perturbation duration.

Module	**Architecture**	**Dims**
Network configuration		
Actor	MLP	hidden $= [2048, 2048, 1024, 1024, 512, 512]$
Action dimension	Diagonal Gaussian	29
Critic	MLP	hidden $= [2048, 2048, 1024, 1024, 512, 512]$
Motion command		
Future frames	$F_{future} = 5$ frames, $F_{hist} = 10$ frames	
Frame interval	$\Delta dt_{future} = 0.04s$, $\Delta dt_{hist} = 0.02s$	

Table 4: Universal control policy hyperparameters.