HMC: Learning Heterogeneous Meta-Control for Contact-Rich Loco-Manipulation

Lai Wei* **Xuanbin Peng*** **Ri-Zhao Qiu** **Xuxin Cheng** **Xiaolong Wang**

UC San Diego

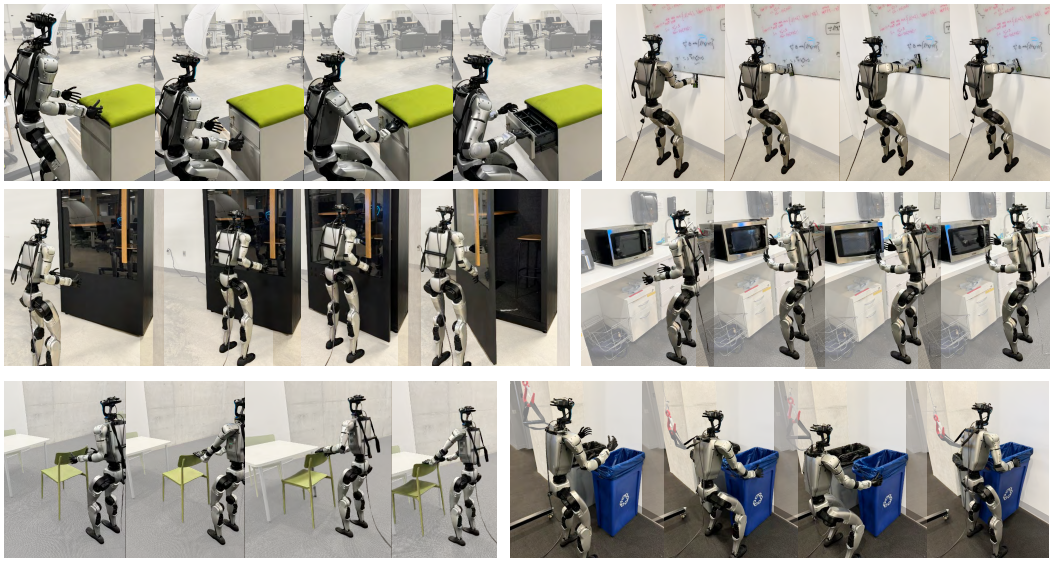

Figure 1: **Rolling out HMC for contact-rich tasks on a humanoid robot.** Compared to naïve position-only policies [1, 2, 3, 4], HMC designs a blending interface (HMC-Controller) that dynamically balances the torques from position tracking and force-aware controllers for contact-rich tasks. Using a heterogeneous architecture, HMC-Policy trains on both large-scale position-only data and fine-grained force-aware demonstrations. (Shown tasks: opening a wheeled cabinet; wiping a whiteboard; pulling a door; opening a microwave; rearranging chairs, and lifting a trash can.)

Abstract: Learning from real-world robot demonstrations hold promises for interacting with complex real-world environments. However, the complexity and variability of interaction dynamics often cause purely positional controllers to struggle with contacts or varying payloads. To address this, we propose a **H**eterogeneous **M**eta-**C**ontrol (HMC) framework for Loco-Manipulation that adaptively stitches multiple control modalities: position, impedance, and hybrid force-position. We first introduce an interface, **HMC-Controller**, for blending actions from different control profiles **continuously in the torque space**. HMC-Controller facilitates both teleoperation and policy deployment. Then, to learn a robust force-aware policy, we propose **HMC-Policy** to unify different controllers into a heterogeneous architecture. We adopt a mixture-of-experts style routing to learn from large-scale position-only data and fine-grained force-aware demonstrations. Experiments on a real humanoid robot show over 50% relative improvement over baselines on challenging tasks such as compliant table wiping and drawer opening, demonstrating the efficacy of HMC.

Keywords: Loco-Manipulation, Contact-Rich, Heterogeneous Learning

*equal contribution

RSS 2025 Workshop on Whole-Body Control and Bimanual Manipulation (RSS2025WCBM).

1 Introduction

Robots that walk and manipulate in a seamless whole — known as *loco-manipulation* systems — extend robot autonomy beyond structured environments to enable versatile operations in homes, warehouses, and disaster sites. Recent advancements in large-scale learning methods, such as imitation learning [2, 1, 5, 3, 4], sim-to-real transfer [6, 7], and zero-shot planning [8, 9, 10, 11], have driven significant progress in robot manipulation. Internet-scale pre-trained visual models [12, 13, 14] and sophisticated teleoperation interfaces [15, 16, 17, 18] have notably enhanced robotic capabilities.

However, these state-of-the-art methods primarily rely on *position-only controllers* [2, 1, 5, 3, 4] that track only joint angles or end effector poses, which excel in tracking accuracy yet fundamentally neglect complex interaction dynamics. On the other hand, real-world tasks inherently involve **contact-rich interactions**, such as wiping surfaces, lifting heavy or delicate objects, or opening the drawers secured by magnets. These scenarios demand precise yet adaptable management of both *motion* and *force*. Consequently, these controllers frequently generate hazardous oscillations and excessive forces in contact-rich tasks.

Traditional compliance methodologies, including impedance control [19, 20] and hybrid force-position control [21, 22], address some of these limitations by either dynamically adapting stiffness [19, 20] or precisely managing forces in normal and shear directions [21, 22]. However, traditional approaches typically focus on a single, manually-tuned setting, lacking generalizability across different scenes. To address such an issue, recent methods [23, 24, 25] incorporate learning-based strategies to learn force-aware actions in a data-driven fashion. FACTR [23] and RDP [24] use force feedback to handle contact-rich interactions, while ACP [25] use imitation learning to dynamically adjust stiffness parameters. However, existing learning-based methods often rely on a specific type of compliant controller and a small set of domain-specific expert data collected by the exact controller, in addition to requiring expensive force sensors.

This paper identifies three types of challenges to develop a robust force-aware robot policy:

1. **Modality Mismatch:** Position, impedance, and hybrid controllers each excel in different task phases, but no single mode is sufficient for the entire task.
2. **Data Imbalance:** Large-scale teleoperation data is predominantly positional, whereas demonstrations leveraging compliance or force control are scarce.
3. **Abrupt Switching:** Hard, discrete transitions between controllers cause torque discontinuities, leading to unstable and unsafe interactions.

To overcome these challenges, we introduce *Heterogeneous Meta-Control (HMC)*. Our meta controller, HMC-Controller, is a meta-control [26] interface that takes inputs from multiple low-level control modalities (position, impedance, hybrid force-position) and computes unified torque actions. HMC-Controller operates on-the-fly in the torque space, which enables continuous blending of different control profiles based on the evolving task progress and environment. We then build HMC-Policy, which learns all control profiles simultaneously in a heterogeneous fashion. HMC-Policy predicts actions from different control profiles, which are sent to the low-level HMC-Controllers. With a soft Mixture-of-Expert (MoE) routing design, HMC-Policy autonomously weights different controllers across the task execution horizon. Unlike traditional discrete switching strategies, our soft router ensures smooth and interpretable transitions between controllers, prevents expert collapse from imbalanced data distributions, and produces stable behaviors vital for real-world deployment. Furthermore, the heterogenous design intuitively enables a two-stage training-fine-tuning paradigm to leverage existing position-only training demonstrations.

Empirical evaluations conducted on challenging tasks demonstrate that HMC significantly outperforms conventional fixed or discrete-switching baselines. In summary, our contributions are:

- **Unified low-level control interface.** HMC-Controller is a controller interface that blends different control profiles continuously in the torque space, which can be adapted for teleoperation, policy rollout, and across different embodiments.

- **Heterogeneous high-level policy.** HMC-Policy uses heterogeneous learning, which adapts a two-stage training paradigm and MoE-style soft routing to use abundant positional data as a strong prior and fine-grained multi-expert demonstrations to address data imbalance.
- **Real-world evaluation.** Evaluations on various contact-rich tasks to show notable enhancements in stability, compliance, and adaptability in real-world scenarios.

2 Related Work

Manipulation Policies. Classical manipulation policies rely on model-based control or motion planning [27, 28], which often fail to generalize in unstructured real-world environments. In contrast, learning-based manipulation policies have seen rapid progress in recent years. Behavior Cloning [29] is a common approach to directly map from observations to actions. Recent developments include transformer-based models [30, 31, 15] and diffusion-based methods [32, 33, 34, 35], which aim to mitigate multi-modality and improve trajectory generation. Despite their success, these methods often rely on low-level positional controllers, making them ill-suited for contact-rich tasks that require reasoning about forceful interactions or multi-stage behaviors.

Loco-Manipulation. Loco-manipulation has emerged as an increasingly important direction in robotic control, especially in settings that demand both mobility and dexterous interaction. Early efforts such as [36, 7] primarily addressed static or quasi-static tasks, focusing on balance-constrained manipulation. More recent works including [37, 38, 39] tackle dynamic, contact-rich tasks that require whole-body coordination and physical stability during interaction. These systems highlight the growing demand for controllers that account for both task-specific objectives and the robot's whole-body dynamics.

Contact-Rich Compliance Control. Contact-rich manipulation often requires compliant behavior to maintain stability and prevent damage during interaction. Traditional model-based methods such as [19, 21, 40] explicitly regulate force and motion based on physical models, but are difficult to tune and deploy in unstructured or dynamic environments. Learning-based approaches have gained traction, typically through reinforcement learning (RL) or learning from demonstration (LfD). Some RL-based methods [41] predict compliance parameters in real time, which are then passed to a low-level controller. Others [42] integrate the RL agent directly into the low-level control loop. LfD methods [25, 43, 44, 45] often aim to infer both target motion and a set of compliance parameters from human demonstrations. Despite progress, these methods face a common challenge: collecting data with reliable force or impedance supervision remains difficult.

Meta Controllers. Meta-controllers aim to automate the selection and coordination of robot skills, enabling agents to perform diverse tasks by dynamically invoking the most suitable experts. Prior approaches include skill library methods such as [46, 47], which retrieve or blend existing policies for new tasks. More recent efforts leverage large language models (LLMs) to guide skill selection, parameter inference, or reward specification, as seen in [48, 49]. Building on this line of work, [26] introduces a hierarchical control framework that synthesizes customized models and controllers per task via LLMs, mimicking human expert reasoning. However, these LLM-based methods often face latency constraints and struggle to integrate real-time onboard observations—such as contact feedback or phase transitions—into the control loop. This makes them ill-suited for multi-phase, contact-rich tasks that require fast and reactive expert switching. Our approach similarly adopts a structured hierarchy, but focuses on explicit phase switching and expert policy selection, enabling real-time, feedback-driven adaptation rather than offline synthesis.

3 HMC-Controller for Teleoperation and Deployment

Our loco-manipulation meta-control teleoperation system is illustrated in Fig. 2. Though our controller is technically agnostic of robot embodiment, the current version studies the Unitree G1 humanoid with 7 DoF arms for its low cost and wide range of adaptation. Note that HMC applies to

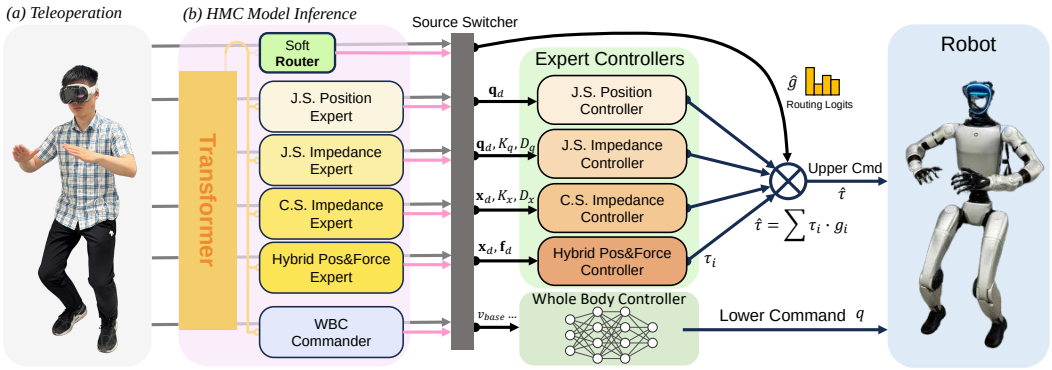

Figure 2: **System overview.** HMC-Controller accepts inputs from either a VR-based teleoperation system or HMC-Policy inference. In the model inference path, multiple expert heads output corresponding control strategies. In the teleoperation path, shared hand poses are distributed to the expert controllers with different controller-specific parameters. All expert controllers output joint-space torque commands, which are blended via soft routing using predicted soft weights. Finally, the fused torque commands and lower-body joint targets are executed by the robot in real time. ("J.S": Joint Space. "C.S": Cartesian Space.)

only the upper body of the robot for manipulation. We use an off-the-shelf whole-body humanoid controller [37] to translates upper head and hand motions into lower-body motion.

HMC-Controller. To get the best out of position-based controllers and force-aware controllers, we implement a suite of primitive controllers: pure position, impedance, and hybrid controllers. Each controller follows standard implementation and accepts desired Cartesian and/or joint setpoints along with a controller-specific profile. The outputs are torque commands for motors.

Given the robot joint positions $\mathbf{q}$, **pure position controller** is a standard PD controller [50] that directly tracks the desired joint positions $\mathbf{q}_d$ with gain coefficients K_p and D_p

$$\boldsymbol{\tau} = K_p(\mathbf{q}_d - \mathbf{q}) + D_p(\dot{\mathbf{q}}_d - \dot{\mathbf{q}}) \tag{1}$$

Joint-space impedance controller modulates compliance at the joint level with desired joint space stiffness K_q and damping D_q. $\tau_{gravity}$ and $\tau_{friction}$ are gravity and friction compensation torques at the current configuration

$$\boldsymbol{\tau} = K_q(\mathbf{q}_d - \mathbf{q}) + D_q(\dot{\mathbf{q}}_d - \dot{\mathbf{q}}) + \tau_{gravity} + \tau_{friction} \tag{2}$$

Cartesian-space impedance controller regulates end-effector compliance in the Cartesian task space using the Jacobian $J(\mathbf{q})$ and cartesian space stiffness K_x and damping D_x:

$$\boldsymbol{\tau} = J^\top \left[K_x(\mathbf{x}_d - \mathbf{x}) + D_x(\dot{\mathbf{x}}_d - \dot{\mathbf{x}})\right] + \tau_{gravity} + \tau_{friction} \tag{3}$$

Finally, **hybrid position-force controller** tracks position along free axes while maintaining a specified force $\mathbf{f}_d$ along constrained directions, using selection matrices S_p and S_f:

$$\boldsymbol{\tau} = J^\top \left[S_p \left(K_x(\mathbf{x}_d - \mathbf{x}) + D_x(\dot{\mathbf{x}}_d - \dot{\mathbf{x}})\right) + S_f(\mathbf{f}_d - \mathbf{f})\right] + \tau_{gravity} + \tau_{friction} \tag{4}$$

Given actions corresponding to different controllers at the same timestamp, HMC-Controller computes torque for each controller, and performs a soft weighted average to obtain the final torques for execution. A low-pass filter is applied to ensure continuity and stability.

Whole-body controller for the lower body. To enable the G1 robot with dexterous movement, we enhance the lower body motion with an off-the-shelf whole-body policy [37]. The purpose of the off-the-shelf whole-body policy is to signficantly extend the robot workspace (as shown in Fig. 1) and to demonstrate how HMC can be plugged-and-played with existing locomotion policies. The policy takes in commands of either $\mathbf{x}_{head}$ or guided directly via the second operator using a remote controller. More concretely, the off-the-shelf policy accepts high-level commands—such

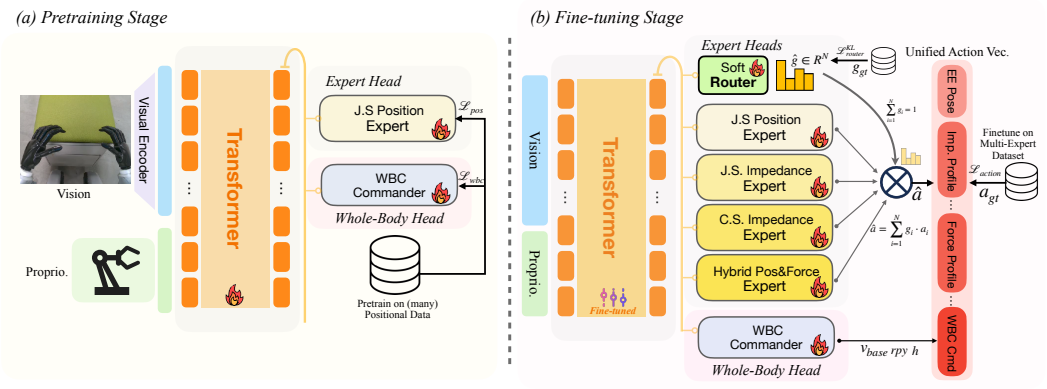

Figure 3: **Overview of Two-stage HMC. (a) Pretraining:** We harness abundant positional demonstrations to train the shared transformer trunk and position expert head, thereby embedding a strong positional prior that boosts generalization. **(b) Fine-tuning Stage:** All parameters are unfrozen and fine-tuned on a smaller, fine-grained multi-expert dataset. A soft routing network learns to blend outputs from multiple experts, producing smooth and adaptive control policies. ("J.S": Joint Space. "C.S": Cartesian Space.)

as base velocity, waist height, and pitch angle—together with upper-limb joint angles, and outputs computed lower-body joint angles at 50 Hz.

Teleoperation Architecture. We use OpenTV [16] to obtain accurate head ($\mathbf{x}_{head}$) and hand ($\mathbf{x}_{hand}$) poses tracking at 50 Hz, which can be solved to joint angles $\mathbf{q}_d$ using inverse kinematics with the Pinocchio library [51, 52]. In addition, our meta-control teleoperation system incorporates a user-friendly dashboard that enables the second operator to switch control modes and adjust parameters (e.g., impedance stiffness) on the fly. In single-user operation mode, the primary operator may likewise issue voice commands—either qualitative ("switch to compliance mode") or quantitative ("increase stiffness by 20%")—to reconfigure the teleoperation controller.

As a low-cost teleoperation solution, our system dispenses with dedicated physical control sticks (e.g., [53, 23]) and end-effector force sensors. Instead, to convey haptic cues during object interaction, we perform an online, coarse estimation of contact forces based on joint motor torques and positional errors. Together we visualize these forces on the virtual robot arm within the 3D scene to allow an immersive perception of contact dynamics than a conventional 2D display.

4 Learning Algorithm: Heterogeneous Meta-Control Policy

Problem Formulation and Motivation. We frame the task of meta-control for loco-manipulation as a *heterogeneous behavioral cloning* problem, where the goal is to replicate expert-level, multi-modal control behaviors demonstrated via teleoperation. Formally, we consider a demonstration dataset consisting of trajectories:

$$\tau = \{(o_t, a_t, g_t)\}_{t=1}^{T}, \quad a_t = \{\mathbf{EE}, \text{Imp. Stiff}, \cdots, \mathbf{F}\} \tag{5}$$

where at each timestep t, the robot receives multi-sensory observations o_t, including visual o_t^{vis} and proprioceptive o_t^{prop} data. The action $a_t \in \mathbb{R}^{N_a}$ is unified across different controllers, including all potential values such as end-effector (EE) pose and attributes (*e.g.,* impedance stiffness, force vectors). For an action in the demonstration, zero-padding is applied for irrelevant controller attributes to maintain a consistent unified vector space. A soft control belief state $g_t \in [0, 1]$ is used to indicate the relative contribution of each expert at time t, where $\sum_{i=1}^{N} g_{t,i} = 1$.

Critically, teleoperation logs exhibit a highly imbalanced distribution—positional data dominate, while impedance and force modality data are scarce yet crucial for contact-rich interactions. A naive single-head network tends to collapse toward purely positional behavior, neglecting critical compliant behaviors. Conversely, separate specialist networks fail to handle smooth and continuous

modality transitions required in real-world applications. To tackle these challenges, we propose a two-stage training strategy and a Soft Mixture-of-Experts (Soft MoE) architecture capable of dynamically blending control modalities smoothly.

4.1 HMC-Policy: Architecture and Two-Stage Training

Our proposed Heterogeneous Meta-Control Policy (HMC-Policy) approach involves two main stages: (a) a positional pretraining stage leveraging abundant, easily acquired positional data, and (b) a fine-tuning stage utilizing fine-grained multi-expert demonstrations to enable smooth transitions among control modes. The architecture includes a shared Transformer-based encoder that extracts task-relevant latent embeddings, modality-specific expert heads, a soft gating mechanism (router), and a dedicated Whole-Body Control (WBC) commander head, which continuously outputs high-level whole-body directives (e.g., base velocity V_{base}, base height h_{base}, waist posture rpy, etc.) to the low-level HMC whole-body controller, ensuring stable and coordinated locomotion-manipulation behavior throughout all task phases.

Positional data are cheap while contact dynamics vary widely; hence we adopt a shared Transformer trunk for general spatial reasoning and modality-specific experts that specialise in their respective dynamic regimes.

Shared Transformer Trunk. Given the observation $o_t = (o_t^{\text{vis}}, o_t^{\text{prop}})$, we first tokenize each modality using separate tokenizers: a visual tokenizer $f_{\theta_{\text{vis}}}$ and a proprioceptive tokenizer $f_{\theta_{\text{prop}}}$. These tokenizers output fixed-length token embeddings:

$$v_t = f_{\theta_{\text{vis}}}(o_t^{\text{vis}}) \in \mathbb{R}^{N_v \times d}, \quad p_t = f_{\theta_{\text{prop}}}(o_t^{\text{prop}}) \in \mathbb{R}^{N_p \times d}, \tag{6}$$

where v_t and p_t represent token sequences from visual and proprioceptive modalities respectively. The combined tokens are then passed into a Transformer trunk encoder $f_{\theta_{\text{trunk}}}$, which outputs a shared latent embedding $z_t = f_{\theta_{\text{trunk}}}(v_t, p_t) \in \mathbb{R}^D$.

Modality-Specific Expert Heads. We construct N modality-specific experts, each implemented as separate MLPs:

$$a_{t,i} = g_{\theta_i}(z_t), \quad i \in \{\text{pos}, \text{imp}, \text{hybrid force}\}, \tag{7}$$

where each expert outputs an action profile including the EE pose and modality-specific parameters (joint stiffness for joint-space impedance, Cartesian stiffness for Cartesian-space impedance, force vectors for hybrid pos&force control, etc.).

Soft Router. The router head, parameterized by ϕ, predicts a gating weight (soft belief state) distribution:

$$g_t = \text{softmax}(r_\phi(z_t)), \quad g_t \in [0,1]^N, \quad \sum_{i=1}^{N} g_{t,i} = 1\,. \tag{8}$$

Soft routing prevents expert collapse under imbalanced data by encouraging exploring, yields smooth transitions between control modalities, avoids abrupt torque discontinuities and enhances hardware safety during deployment. The final unified action prediction at each timestep is computed by blending the expert outputs weighted by the router outputs $\hat{a}_t = \sum_{i=1}^{N} g_{t,i} \cdot a_{t,i}$.

Whole-Body Control (WBC) Commander. A dedicated WBC commander head, $h_\psi(z_t)$, continuously outputs whole-body high-level commands:

$$a_t^{\text{WBC}} = h_\psi(z_t). \tag{9}$$

This head operates independently of the expert routing mechanism, ensuring consistent whole-body stability and coordination. Its output is concatenated into the final unified action vector alongside the blended expert predictions.

Pretraining Stage (Positional Prior Learning). Initially, we freeze all expert heads except the positional expert and WBC commander, and pretrain the Transformer trunk on large positional-only datasets. Positional data is abundant and easier to acquire, naturally covering diverse variations such

as initial poses, recovery behaviors, and positional distributions. This embeds a strong positional prior into the shared representation:

$$\mathcal{L}_{\text{pretrain}} = \|a_t^{\text{pos}} - a_t^{\text{pos, gt}}\|_1 + \|a_t^{\text{WBC}} - a_t^{\text{WBC, gt}}\|_1 . \tag{10}$$

Fine-tuning Stage (Multi-Expert Integration). In the fine-tuning stage, we employ a smaller, balanced multi-expert dataset. All heads are unfrozen and co-trained, while the Transformer trunk receives a reduced learning rate to preserve the pretrained positional embedding. A balanced multi-expert dataset and the soft router keep gradients flowing to every expert, avoiding modality collapse and enabling smooth blending. The final fine-tuning objective is given by

$$\mathcal{L}_{\text{finetune}} = \lambda_{\text{action}} \cdot \underbrace{\|\hat{a}_t - a_t^{gt}\|_1}_{\mathcal{L}_{\text{action}}} + \lambda_{\text{router}} \cdot \underbrace{D_{\text{KL}}(g_t^{gt} \| g_t)}_{\mathcal{L}_{\text{router}}}, \tag{11}$$

where g_t^{gt} in $\mathcal{L}_{\text{router}}$ is a smoothed ground truth signal obtained via low-pass filtering of tele-operated modality IDs, and λ are loss-balancing hyper-parameters. At deployment, the HMC-Policy sends each expert's output to the HMC-Controller, which converts them into torque commands and blends these torques according to the soft-router weights g_t, allowing the experts to cooperate adaptively with the task's changing dynamics. Together, the pre-trained positional prior and soft-routed fine-tuning yield experts that are both robust and seamlessly blendable.

5 Experiments

5.1 Experiment Setup

Hardware. We conduct experiments using the Unitree G1 humanoid robot, equipped with two 7-DoF arm for loco-manipulation tasks. The visual information is acquired from an Intel RealSense D435i camera mounted on the robot's head, and all proprioceptive data are obtained via the Unitree SDK. The robot is controlled through our HMC-Controller interface (shown in Fig. 2), enabling seamless demonstration collection and precise multi-modal control execution.

Task Setup. We evaluate our approach across three contact-rich loco-manipulation tasks (Fig. 4).

- **Wipe Table:** The humanoid robot needs to wipe the table surface to remove marker traces (green in the figure). This task requires **appropriate force regulation** capability—insufficient force fails to erase the marks, while excessive force may cause large unsafe movement. A success is recorded if marks are wiped in two loops.
- **Lift Bottle with Both Hands:** The robot lifts a bottle using only the bare end-effectors without grippers or hand, only relying on arm friction to secure the object. This task requires the following capabilities: (1) **bimanual coordination** to maintain closed-chain dynamic contact and lift the bottle stably; and (2) **Precision**, as the robot needs to lift bottles with different size, shape, and material. The task is considered successful if the bottle is lifted 10 cm above the table and held steadily for 3 seconds.
- **Open Drawer:** The humanoid robot crouches and inserts its hand into a narrow handle slot (Hand Insertion), then pulls to open a drawer secured by a magnetic latch, requiring over 20N of force to overcome closure resistance (Pulling). This task requires: (1) **stage-aware control** to switch between compliant insertion and forceful pulling stages to achieve high success rate; (2) **dynamic stiffness modulation** to adjust impedance properties during interaction with the drawer; and (3) **whole-body stability** to maintain lower-body balance while the upper body applies large interaction forces. The task is considered successful if the drawer is pulled open by at least 20 cm.

Figure 4: Tasks visualization.

	Wipe Table	Lift Bottle	Open Drawer
Stiff Policy	5/15	10/15	12/15
ACT (vanilla)	5/15	9/15	6/15
ACT (meta)	7/15	10/15	7/15
HMC (w/o soft routing)	13/15	**14/15**	12/15
HMC (ours)	**14/15**	**14/15**	**13/15**

Table 1: Overall policy success rate comparisons of HMC and baselines on three contact-rich tasks.

	Lift Bottle		Open Drawer			
	seen	unseen	seen		unseen	
			Hand Insertion	Pulling	Hand Insertion	Pulling
Stiff Policy	10/15	5/15	12/15	12/15	5/15	3/15
Compliant Policy	12/15	6/15	12/15	8/15	**10/15**	2/15
HMC (w/o soft routing)	**14/15**	8/15	12/15	12/15	8/15	7/15
HMC (from scratch)	13/15	9/15	9/15	9/15	7/15	5/15
HMC (ours)	**14/15**	**12/15**	**13/15**	**13/15**	**10/15**	**10/15**

Table 2: Ablation experiment results across seen and unseen settings for the Lift Bottle and Open Drawer tasks. The proposed soft routing and position-only pretraining increases success rate considerably. Hand insertion and pulling correspond to two separate stages in opening a drawer.

Baselines. we implement the baselines as follows:

- **ACT (vanilla)**: Action Chunking Transformer (ACT) policy, trained solely on positional data.
- **ACT (meta)**: ACT policy trained with positional data and controller profiles. Hard routing switch.
- **Stiff Policy**: Our model architecture trained exclusively on stiff positional expert data.
- **Compliant Policy**: Our model architecture trained exclusively on impedance experts data.
- **HMC w/o soft routing**: Our HMC architecture with a hard `argmax` router instead of soft routing.
- **HMC (from scratch)**: Our HMC architecture trained from scratch instead of two stages training.
- **HMC (ours)**: The complete implementation of our proposed HMC-Policy.

Evaluation Protocols. For the Open Drawer and Lift Bottle tasks, we evaluate generalization under seen and unseen settings, where the seen settings correspond to configurations encountered during training, and unseen ones involve novel object instances or initial positions for Lift Bottle and Open Drawer tasks respectively. Each policy variant is evaluated over 15 trials, with performance assessed based on the task success rate.

5.2 Evaluation Results

The general experiment results across different policies are shown in Table 1, and the ablation studies result are demonstrated in Table 2. We try to answer important questions in our evaluation:

- **Q1:** What advantages does our HMC-Policy have compared to baselines?
- **Q2:** Why do we need HMC-Controller for contact-rich and multi-stage tasks?
- **Q3:** What are benefits of Soft Routing?
- **Q4:** Does pretrain and finetune approach helps training and generalization performance?
- **Q5:** Why do our HMC-Policy excels at contact-rich unseen tasks?

A1: Our HMC consistently outperforms baseline approaches across all three tasks. In contrast, traditional stiff policies—which are commonly used in manipulation tasks—suffer from poor adaptability in contact-rich scenarios. For instance, in the Wipe Table task, the high rigidity of the stiff controller resulted in excessive torques on the arm, occasionally triggering motor overheating protection due to unsafe contact forces.

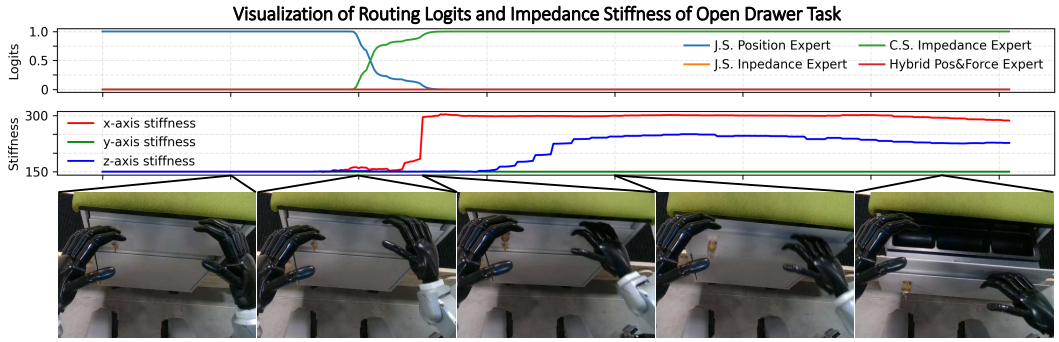

Figure 5: **Interpretablility of routing logits** in visualization of an Open Drawer Episode. The upper figure is the predicted routing logits, while the lower figure shows the predicted right hand's stiffness (N/m). Five images from the head camera are demonstrated.

While ACT (meta) improves upon its vanilla counterpart by simultaneously predicting both motion trajectories and meta-controller profiles, its single-output architecture introduces significant learning challenges. Specifically, the model must infer both low-stage control commands and high-level control modes through a shared output head, which leads to unstable behaviors in complex tasks due to brittle profile predictions.

In contrast, our HMC offers a more modular and robust framework. In contact-rich interactions it dynamically selects the appropriate expert controller for each stage of a multi-stage task. This separation of meta-level decision-making from trajectory generation significantly reduces learning complexity but also enables more reliable execution across challenging manipulation scenarios.

A2: An HMC-Controller is essential as it enables adaptive control switching across task phases. Multi-stage, contact-rich manipulation tasks inherently require dynamic adaptation of control strategies across distinct phases. Take the Open Drawer task as an illustrative example: in the Hand Insertion stage, the robot must approach and insert its end-effector into the drawer handle. This phase benefits from a compliant control mode, as the flexibility allows the end-effector to slide smoothly along the drawer surface and align with the handle slot despite slight positional errors. Conversely, the Pulling stage demands sufficient stiffness along the pulling direction to exert the force necessary to overcome the magnetic resistance and successfully open the drawer.

As shown in Table 2, in unseen settings, fixed policies such as fully stiff or purely compliant control exhibit low success rates, as neither can accommodate the distinct requirements of both stages. Stiff controllers often fail to compliant during insertion, while compliant ones lack the force required for effective drawer pulling.

Our meta-controller addresses this challenge by providing flexible and phase-aware control routing. Specifically, as shown in Fig. 5, HMC-Policy initially employs a positional controller to approach the handle, then dynamically transits to a Cartesian impedance controller with moderate stiffness as the robot makes contact. Upon successful insertion, the controller further adjusts the stiffness matrix to increase rigidity along the pulling direction. This behavior is facilitated by the HMC-Controller's routing mechanism, which selects the appropriate expert controllers based on inferred task phase, while each expert specializes in producing the optimal control commands for its respective subtask.

A3: Soft routing enables smooth and stable transitions between expert controllers. This is critical in contact-rich tasks, where abrupt switching can destabilize the interaction. In our HMC-Controller framework, soft routing allows for continuous blending of control outputs, ensuring robustness during phase transitions.

In the Lift Bottle task under unseen settings, we observed that the HMC-Policy without soft routing often triggered an abrupt expert controller switch when the arm made contact with the bottle. Subtle discrepancies between expert controllers caused sudden changes in gripping force, occasionally

leading to slippage of the cylindrical bottle. By contrast, soft routing allowed the robot to interpolate control outputs across experts, maintaining stability during transitions.

A4: Pretraining followed by finetuning improves both training stability and generalization. Compared to the training-from-scratch variant in Table 2, the pretrain-and-finetune strategy significantly enhances performance, particularly in unseen scenarios. During pretraining on position-only data, the shared stem network learns fundamental task representations and spatial alignment. Subsequent finetuning on heterogeneous controller data focuses learning on expert controller outputs, leveraging a well-initialized stem. This two-stage process mitigates conflicts among expert losses and stabilizes gradients during training. Moreover, the pretrained stem provides a strong prior, improving generalization and reducing the risk of overfitting to training environments.

A5: Our HMC-Policy excels in contact-rich unseen tasks by combining adaptive control routing with strong generalization. Contact-rich tasks often involve unpredictable variations in contact dynamics, requiring adaptive and robust control strategies. Our HMC-Policy framework addresses this by combining meta-level scheduling of diverse expert controllers with smooth transitions enabled by soft routing. Additionally, the pretrain-and-finetune paradigm provides a strong initialization, improving the generalizability with existing position-only data.

6 Conclusion

Contact-rich loco-manipulation exposes the limits of single-mode, position-only control. In this work, we tackled this gap with Heterogeneous Meta-Control (HMC), a unified framework in which HMC-Controller blends position, impedance and hybrid force–position commands directly in torque space, while HMC-Policy learns via a two-stage pretrain-finetune paradigm, and Mixture-of-Experts to route those modalities smoothly in response to task phase and contact dynamics. This heterogeneous design compensates for the heavy positional bias in teleoperation data, keeps all experts trainable, and removes the torque discontinuities that plague hard switching .

Comprehensive real-robot experiments on the Unitree G1 across diverse tasks—table wiping, bimanual bottle lifting, and drawer opening—show that HMC-Policy consistently outperforms stiff, compliant-only, and hard-switching baselines, delivering higher success rates, smoother force profiles, and better generalization to unseen objects and configurations. Taken together, these results demonstrate that adaptive, soft routing of complementary control modalities is a practical path toward robust whole-body loco-manipulation in the wild. Future work will explore scaling to a broader array of experts (e.g., tactile-conditioned manipulation) and integrating high-level task planners to further extend long-horizon autonomy.

Acknowledgments

This work was supported, in part, by NSF CCF-2112665 (TILOS), and gifts from Amazon and Meta.

References

[1] K. Black, N. Brown, D. Driess, A. Esmail, M. Equi, C. Finn, N. Fusai, L. Groom, K. Hausman, B. Ichter, et al. π_0: A vision-language-action flow model for general robot control. *arXiv preprint arXiv:2410.24164*, 2024.

[2] S. Liu, L. Wu, B. Li, H. Tan, H. Chen, Z. Wang, K. Xu, H. Su, and J. Zhu. Rdt-1b: a diffusion foundation model for bimanual manipulation. *arXiv preprint arXiv:2410.07864*, 2024.

[3] M. J. Kim, K. Pertsch, S. Karamcheti, T. Xiao, A. Balakrishna, S. Nair, R. Rafailov, E. Foster, G. Lam, P. Sanketi, et al. Openvla: An open-source vision-language-action model. In *Conference on Robot Learning (CoRL)*, 2024.

[4] Octo Model Team, D. Ghosh, H. Walke, K. Pertsch, K. Black, O. Mees, S. Dasari, J. Hejna, C. Xu, J. Luo, T. Kreiman, Y. Tan, L. Y. Chen, P. Sanketi, Q. Vuong, T. Xiao, D. Sadigh, C. Finn, and S. Levine. Octo: An open-source generalist robot policy. In *Proceedings of Robotics: Science and Systems*, 2024.

[5] A. O'Neill, A. Rehman, A. Gupta, A. Maddukuri, A. Gupta, A. Padalkar, A. Lee, A. Pooley, A. Gupta, A. Mandlekar, et al. Open x-embodiment: Robotic learning datasets and rt-x models. *arXiv preprint arXiv:2310.08864*, 2023.

[6] T. Lin, K. Sachdev, L. Fan, J. Malik, and Y. Zhu. Sim-to-real reinforcement learning for vision-based dexterous manipulation on humanoids. *arXiv preprint arXiv:2502.20396*, 2025.

[7] M. Liu, Z. Chen, X. Cheng, Y. Ji, R.-Z. Qiu, R. Yang, and X. Wang. Visual whole-body control for legged loco-manipulation. *arXiv preprint arXiv:2403.16967*, 2024.

[8] M. Ji, R.-Z. Qiu, X. Zou, and X. Wang. Graspsplats: Efficient manipulation with 3d feature splatting. In *CoRL*, 2024.

[9] W. Shen, G. Yang, A. Yu, J. Wong, L. P. Kaelbling, and P. Isola. Distilled feature fields enable few-shot language-guided manipulation. In *CoRL*, 2023.

[10] A. Rashid, S. Sharma, C. M. Kim, J. Kerr, L. Y. Chen, A. Kanazawa, and K. Goldberg. Language embedded radiance fields for zero-shot task-oriented grasping. In *CoRL*, 2023.

[11] W. Huang, C. Wang, Y. Li, R. Zhang, and L. Fei-Fei. Rekep: Spatio-temporal reasoning of relational keypoint constraints for robotic manipulation. In *CoRL*, 2024.

[12] S. Liu, Z. Zeng, T. Ren, F. Li, H. Zhang, J. Yang, Q. Jiang, C. Li, J. Yang, H. Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *ECCV*, 2024.

[13] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

[14] M. Oquab, T. Darcet, T. Moutakanni, H. Vo, M. Szafraniec, V. Khalidov, P. Fernandez, D. Haziza, F. Massa, A. El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

[15] T. Z. Zhao, V. Kumar, S. Levine, and C. Finn. Learning fine-grained bimanual manipulation with low-cost hardware. *arXiv preprint arXiv:2304.13705*, 2023.

[16] X. Cheng, J. Li, S. Yang, G. Yang, and X. Wang. Open-television: Teleoperation with immersive active visual feedback. *arXiv preprint arXiv:2407.01512*, 2024.

[17] A. Iyer, Z. Peng, Y. Dai, I. Guzey, S. Haldar, S. Chintala, and L. Pinto. Open teach: A versatile teleoperation system for robotic manipulation. *arXiv preprint arXiv:2403.07870*, 2024.

[18] C. Chi, Z. Xu, C. Pan, E. Cousineau, B. Burchfiel, S. Feng, R. Tedrake, and S. Song. Universal manipulation interface: In-the-wild robot teaching without in-the-wild robots. *arXiv preprint arXiv:2402.10329*, 2024.

[19] N. Hogan. Impedance control: An approach to manipulation: Part i—theory. *Journal of Dynamic Systems, Measurement, and Control*, 107(1):1–7, 03 1985. ISSN 0022-0434. doi:10.1115/1.3140702. URL https://doi.org/10.1115/1.3140702.

[20] T. Lozano-Pérez, M. T. Mason, and R. H. Taylor. Automatic synthesis of fine-motion strategies for robots. *The International Journal of Robotics Research*, 3(1):3–24, 1984. doi:10.1177/027836498400300101. URL https://doi.org/10.1177/027836498400300101.

[21] M. H. Raibert and J. J. Craig. Hybrid position/force control of manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 103(2):126–133, 06 1981. ISSN 0022-0434. doi:10.1115/1.3139652. URL https://doi.org/10.1115/1.3139652.

[22] M. Uchiyama and P. Dauchez. A symmetric hybrid position/force control scheme for the coordination of two robots. In *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, pages 350–356 vol.1, 1988. doi:10.1109/ROBOT.1988.12073.

[23] J. J. Liu, Y. Li, K. Shaw, T. Tao, R. Salakhutdinov, and D. Pathak. Factr: Force-attending curriculum training for contact-rich policy learning, 2025. URL https://arxiv.org/abs/2502.17432.

[24] H. Xue, J. Ren, W. Chen, G. Zhang, Y. Fang, G. Gu, H. Xu, and C. Lu. Reactive diffusion policy: Slow-fast visual-tactile policy learning for contact-rich manipulation. *arXiv preprint arXiv:2503.02881*, 2025.

[25] 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. *arXiv preprint arXiv:2410.09309*, 2024.

[26] T. Wei, L. Ma, R. Chen, W. Zhao, and C. Liu. Meta-control: Automatic model-based control synthesis for heterogeneous robot skills, 2024. URL https://arxiv.org/abs/2405.11380.

[27] H. Guo, F. Wu, Y. Qin, R. Li, K. Li, and K. Li. Recent trends in task and motion planning for robotics: A survey. *ACM Comput. Surv.*, 55(13s), July 2023. ISSN 0360-0300. doi:10.1145/3583136. URL https://doi.org/10.1145/3583136.

[28] C. Zhou, B. Huang, and P. Fränti. A review of motion planning algorithms for intelligent robotics, 2021. URL https://arxiv.org/abs/2102.02376.

[29] M. Bojarski, D. D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao, and K. Zieba. End to end learning for self-driving cars, 2016. URL https://arxiv.org/abs/1604.07316.

[30] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. J. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K.-H. Lee, S. Levine, Y. Lu, U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. Ryoo, G. Salazar, P. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and

B. Zitkovich. Rt-1: Robotics transformer for real-world control at scale, 2023. URL https://arxiv.org/abs/2212.06817.

[31] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, X. Chen, K. Choromanski, T. Ding, D. Driess, A. Dubey, C. Finn, P. Florence, C. Fu, M. G. Arenas, K. Gopalakrishnan, K. Han, K. Hausman, A. Herzog, J. Hsu, B. Ichter, A. Irpan, N. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, L. Lee, T.-W. E. Lee, S. Levine, Y. Lu, H. Michalewski, I. Mordatch, K. Pertsch, K. Rao, K. Reymann, M. Ryoo, G. Salazar, P. Sanketi, P. Sermanet, J. Singh, A. Singh, R. Soricut, H. Tran, V. Vanhoucke, Q. Vuong, A. Wahid, S. Welker, P. Wohlhart, J. Wu, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich. Rt-2: Vision-language-action models transfer web knowledge to robotic control, 2023. URL https://arxiv.org/abs/2307.15818.

[32] C. Chi, Z. Xu, S. Feng, E. Cousineau, Y. Du, B. Burchfiel, R. Tedrake, and S. Song. Diffusion policy: Visuomotor policy learning via action diffusion. *The International Journal of Robotics Research*, page 02783649241273668, 2023.

[33] X. Zhang, M. Chang, P. Kumar, and S. Gupta. Diffusion meets dagger: Supercharging eye-in-hand imitation learning. In *RSS*, 2024.

[34] Y. Ze, G. Zhang, K. Zhang, C. Hu, M. Wang, and H. Xu. 3d diffusion policy: Generalizable visuomotor policy learning via simple 3d representations, 2024. URL https://arxiv.org/abs/2403.03954.

[35] M. Janner, Y. Du, J. B. Tenenbaum, and S. Levine. Planning with diffusion for flexible behavior synthesis, 2022. URL https://arxiv.org/abs/2205.09991.

[36] Z. Fu, X. Cheng, and D. Pathak. Deep whole-body control: Learning a unified policy for manipulation and locomotion, 2022. URL https://arxiv.org/abs/2210.10044.

[37] 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.

[38] 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.

[39] 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.

[40] A. Q. Keemink, H. van der Kooij, and A. H. Stienen. Admittance control for physical human–robot interaction. *The International Journal of Robotics Research*, 37(11): 1421–1444, 2018. doi:10.1177/0278364918768950. URL https://doi.org/10.1177/0278364918768950.

[41] C. C. Beltran-Hernandez, D. Petit, I. G. Ramirez-Alpizar, T. Nishi, S. Kikuchi, T. Matsubara, and K. Harada. Learning force control for contact-rich manipulation tasks with rigid position-controlled robots. *IEEE Robotics and Automation Letters*, 5(4):5709–5716, Oct. 2020. ISSN 2377-3774. doi:10.1109/lra.2020.3010739. URL http://dx.doi.org/10.1109/LRA.2020.3010739.

[42] 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.

[43] M. Aburub, C. C. Beltran-Hernandez, T. Kamijo, and M. Hamaya. Learning diffusion policies from demonstrations for compliant contact-rich manipulation. *arXiv preprint arXiv:2410.19235*, 2024.

[44] T. Kamijo, C. C. Beltran-Hernandez, and M. Hamaya. Learning variable compliance control from a few demonstrations for bimanual robot with haptic feedback teleoperation system. *arXiv preprint arXiv:2406.14990*, 2024.

[45] W. Liu, J. Wang, Y. Wang, W. Wang, and C. Lu. Forcemimic: Force-centric imitation learning with force-motion capture system for contact-rich manipulation. *arXiv preprint arXiv:2410.07554*, 2024.

[46] Z. Liang, Y. Mu, H. Ma, M. Tomizuka, M. Ding, and P. Luo. Skilldiffuser: Interpretable hierarchical planning via skill abstractions in diffusion-based task execution, 2024. URL https://arxiv.org/abs/2312.11598.

[47] R.-Z. Qiu, Y. Song, X. Peng, S. A. Suryadevara, G. Yang, M. Liu, M. Ji, C. Jia, R. Yang, X. Zou, and X. Wang. Wildlma: Long horizon loco-manipulation in the wild, 2025. URL https://arxiv.org/abs/2411.15131.

[48] J. Liang, W. Huang, R. Singh, P. Sermanet, and A. Grover. Code as policies: Language model programs for embodied control. In *International Conference on Learning Representations (ICLR)*, 2023. URL https://arxiv.org/abs/2209.07753.

[49] Y. Ma, J. Liang, W. Huang, and A. Grover. Eureka: Human-level reward design via coding large language models. *arXiv preprint arXiv:2310.01894*, 2023. URL https://arxiv.org/abs/2310.01894.

[50] K. H. Ang, G. Chong, and Y. Li. Pid control system analysis, design, and technology. *IEEE transactions on control systems technology*, 13(4):559–576, 2005.

[51] S. Caron. pink: Python inverse kinematics. https://github.com/stephane-caron/pink, 2021. Accessed: 2025-05-23.

[52] J. Carpentier, G. Saurel, A. Del Prete, and N. Mansard. Pinocchio: fast forward and inverse dynamics for poly-articulated systems. In *IEEE International Symposium on System Integrations (SII)*, pages 614–619. IEEE, 2019.

[53] S. Yang, M. Liu, Y. Qin, R. Ding, J. Li, X. Cheng, R. Yang, S. Yi, and X. Wang. Ace: A cross-platform visual-exoskeletons system for low-cost dexterous teleoperation, 2024. URL https://arxiv.org/abs/2408.11805.