2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2025)
October 19-25, 2025. Hangzhou, China

2025 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 979-8-3315-4393-8/25/$31.00 ©2025 IEEE | DOI: 10.1109/IROS60139.2025.11245884

A Unified Framework to Learn Collision-Free Loco-Manipulation via Adversarial Motion Priors

Huayang Yin, Tangyu Qian, Mingrui Li, Guanchen Lu, Mingyu Cai, and Zhen Kan

***Abstract*— Designing a whole-body controller for loco-manipulation in unstructured real-world environments remains a formidable challenge. Previous approaches have primarily focused on extending the workspace of robotic arms while maintaining quadrupedal landing postures. However, these methods fail to fully exploit the mobility of legged robots. To address these limitations, we propose a unified framework for collision-free loco-manipulation in real-world applications. The framework comprises two key modules: (1) a Loco-manipulation Motion Prior, which generates loco-manipulation skill trajectories via Trajectory Optimization (TO), and (2) a Collision-free Manipulation module using a Model Predictive Path Integral (MPPI)-based trajectory generator and a vector-based trajectory follower. Extensive experiments have been conducted in both simulation and real-world scenarios to evaluate our framework's tracking accuracy, whole-body coordination, and workspace expansion capabilities. Supplementary and videos are available at: `https://sites.google.com/view/loco-mani-amp/`**

Fig.1: The loco-manipulation capability of reaching a high platform using our framework in the simulation (top), and its transfer to the real-world environment (bottom).

I. INTRODUCTION

Legged loco-manipulation integrates legged robots with robotic arms, leveraging enhanced mobility to overcome the workspace limitations of fixed-base systems. However, the high degree of freedom and complex interactive dynamics pose significant challenges in designing a whole-body controller for coordinated loco-manipulation. Prior works primarily adopted model-based method such as Model Predictive Control (MPC) [1] or Trajectory Optimization (TO) to develop whole-body controllers, which heavily rely on the precise modeling of the integrated system's dynamics. While effective in structured environments, these controllers generalize poorly to unstructured real-world scenarios, e.g., when the robot must cling to a wall to manipulate objects on an elevated platform or operate in cluttered environments.

An effective solution is reinforcement learning (RL) [2], which empowers robots to learn loco-manipulation skills through interaction with environments. Existing learning-based methods [3]–[6] achieve loco-manipulation by independently tracking the base velocity and end-effector position via multi-stage training. While these approaches have demonstrated robust and agile loco-manipulation, several challenges persist. First, current methods are limited to a quadrupedal landing posture, where all four legs maintain ground contact simultaneously. Although this posture provides stable support, it constrains the robot's ability to reach high platform over barrier or low ground beneath the obstacle, thereby restricting the workspace of the robotic arm. Second, most studies focus on loco-manipulation in structured environments, neglecting the challenges posed by unstructured settings, which are prevalent in real-world applications.

In this work, we propose a unified whole-body control framework that integrates a loco-manipulation motion prior with a collision-free manipulation planner for 6D end-effector pose tracking. To facilitate the learning of manipulation skills, we develop an effective TO method to generate loco-manipulation motion trajectories. By incorporating the loco-manipulation motion prior into the Goal-Conditioned RL (GCRL) [7] paradigm, we achieve both efficient and effective skill learning. This integration combines the rapid iteration capabilities of model-based methods with the generalization ability of RL, enabling our framework to adapt to diverse scenarios. Additionally, we design a sampling-based manipulation planner that utilizes point-cloud data of obstacles, ensuring robust collision-free manipulation. Therefore, while expanding the workspace in various scenarios, our unified framework not only mitigates the complexity of reward engineering but also maintains reliable collision avoidance.

Our contributions can be summarized as follows:

1) We propose a unified framework to learn collision-free loco-manipulation skills, enabling agile and safe end-effector 6D pose tracking with an expanded workspace.

2) Our approach integrates loco-manipulation motion priors into GCRL, eliminating the need for precise system modeling.

H. Yin, T. Qian, M. Li, G. Lu and Z. Kan are with the Department of Automation, University of Science and Technology of China, Hefei, China, 230026.

M. Cai is with the Department of Mechanical Engineering, University of California, Riverside, CA, USA, 92521.

This work was supported in part by the National Natural Science Foundation of China under Grant 62173314.

During skill learning, a sampling-based collision-free manipulation planner is developed, which can effectively execute tasks while considering real-world obstacles, ensuring safety and robustness.

3) We conduct extensive loco-manipulation experiments in both simulation and real-world unstructured terrains, demonstrating the framework's precise tracking, whole-body coordination, and robust collision avoidance.

II. RELATED WORKS

A. Learning in Legged Locomotion

In recent years, RL algorithms have endowed legged robots with advanced mobility capabilities, such as stair climbing [8], and obstacle traversal [9]. To bridge the sim-to-real gap, [10] employs a teacher-student network to encode proprioceptive information, enabling blind locomotion in complex terrains. However, relying solely on proprioception limits the robot's ability to accurately perceive the environment, making it difficult to navigate unstructured terrains. Leveraging deep neural networks for efficient image processing, [11] uses depth image as exteroceptive input, allowing legged robots to traverse more complex terrains.

Building upon prior work, the parkour capabilities of legged robots were developed in [8], [12], [13] by introducing carefully designed tasks and reward functions. To improve safety during locomotion, [14] employs GCRL to train a network to evaluate reach-avoid value of legged robots, enabling them to switch between agile and recovery policies. However, these approaches heavily rely on reward function engineering, which can be labor-intensive and time-consuming. To mitigate this issue, imitation-based learning methods accelerate the learning process by incorporating expert motion trajectories. For instance, [15] employs a GAN-style Adversarial Motion Priors (AMP) using a dataset derived from real dog motions as part of the reward mechanism. Their results demonstrate that AMP can effectively replace complex reward functions in locomotion training. Similarly, [16] generates a dataset of 8-DoF legged robots through TO and integrates it into RL training, achieving bipedal standing and backflipping. Additionally, [17] incorporates motion priors from TO and trains legged robots using a teacher-student network, simplifying the design of reward function while enhancing locomotion robustness. The key difference between [17] and our approach lies in the scope of motion priors. While [17] focuses on locomotion, our work further designs loco-manipulation motion priors and adapts the AMP algorithm to loco-manipulation tasks, extending the capabilities of legged robots beyond pure locomotion.

B. Legged Loco-Manipulation

Despite remarkable advancements in legged locomotion, enabling robots to perform manipulation during locomotion remains a challenge. Existing works using model-based methods still struggle to effectively coordinate locomotion and manipulation. As an early work in this field, [18] successfully demonstrated door-pushing and door-pulling tasks by accurately modeling both the robots and environment and designing a whole-body controller through MPC. However, due to the computational burden and limited generalizability of model-based approaches, RL has emerged as a promising alternative.

Among the pioneering learning-based methods, [19] trained a unified policy utilizing Regularized Online Adaptation (ROA), enabling whole-body coordination by jointly tracking base velocity and end-effector position. A fully autonomous loco-manipulation system was developed in [5], which consists of a low-level goal-reaching policy and a high-level task-planning policy, trained through a combination of RL and Imitation Learning (IL). To improve the training efficiency and generalizability, [4] developed two collaborative policies, Loco policy and Arm policy, enabling cross-embodiment deployment across different loco-manipulation tasks. Meanwhile, [20] developed a modular framework, which includes a library of generalizable visuomotor skills and an LLM planner, enabling practical, long-horizon manipulation capabilities. Unlike prior methods that track base velocity and end-effector position separately, [6] enables direct tracking of end-effector trajectories, achieving complex manipulation skills such as dynamic tossing and pushing. This is accomplished by integrating real-world human demonstrations with a diffusion-based policy. Although previous works are able to achieve legged loco-manipulation, few research focuses on developing skills that allow legged robots to effectively expand their workspace in unstructured environments.

III. LOCO-MANIPULATION LEARNING

To achieve collision-free loco-manipulation, we propose a unified framework consisting of two modules: loco-manipulation motion prior and collision-free manipulation planning. The two modules are integrated into our unified GCRL paradigm. An overview of the proposed framework is shown in Fig. 2.

A. Motion Prior for Loco-Manipulation

To learn loco-manipulation skills, existing learning-based works require the design of distinct reward functions for specific actions, significantly increasing training effort. To improve the training efficiency, we propose a method that trains each skill using a single skill trajectory with a unified reward function. Specifically, we designed loco-manipulation motion prior, which is incorporated into our GCRL paradigm as a reward component via AMP.

Let $\mathbf{x}(t) := [\mathbf{r}(t), \theta(t), p_i(t)]$ represent the robot state, where $\mathbf{r}$ and θ represent the linear position and orientation of the robot, respectively, and p_i denotes the position of the robot's ith foot. Given a manually defined kinematic skill trajectory $\tilde{\mathbf{x}}(t)$, we first formulate a trajectory optimization problem that takes into account the system's dynamics and constraints. The TO generates a set of feasible control inputs $\mathbf{u}(t) := [\psi_{n_j}(t)]$, where ψ_{n_j} represents the joint angle of the n_jth joint of the legged robot. Specifically, the TO is

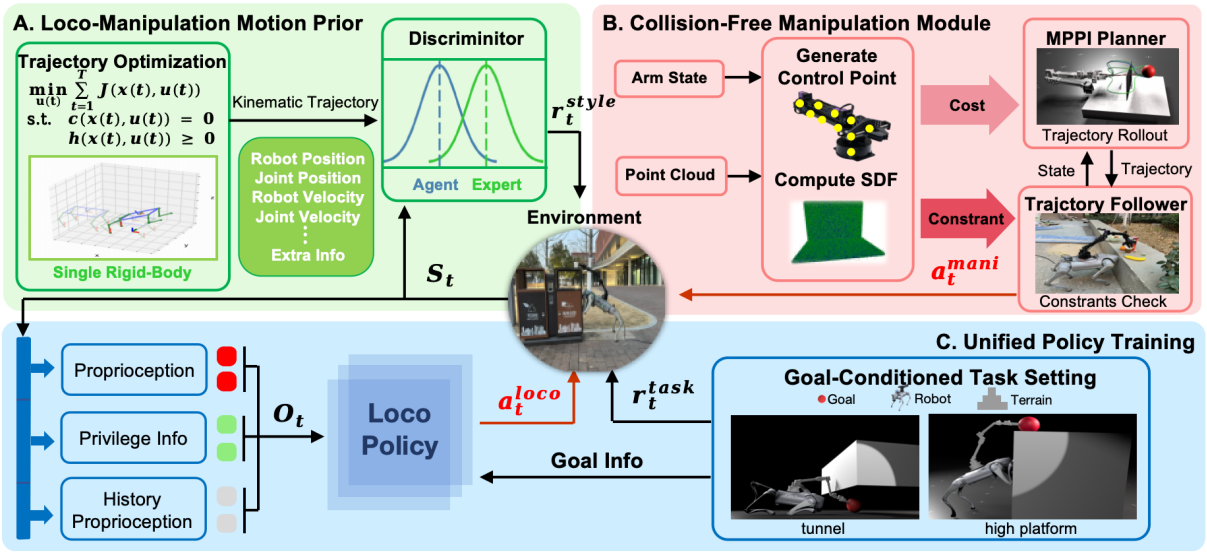

Fig.2: Overview of the loco-manipulation framework. (**A**) A set of loco-manipulation motion priors is first generated using TO. A discriminator then incorporates the motion prior as a style reward during training. (**B**) The manipulation controller consists of an MPPI-based trajectory generator and a vector field-based trajectory follower, ensuring collision-free execution. (**C**) The GCRL paradigm integrates the motion prior with the collision-free manipulation, enabling efficient and adaptive skill learning.

formulated as a Nonlinear Programming Problem

$$\begin{aligned} \min_{\mathbf{u}(t)} \quad & \mathbf{J}(\mathbf{x}(t), \mathbf{u}(t)) \\ \text{s.t.} \quad & \mathbf{c}(\mathbf{x}(t), \mathbf{u}(t)) \geq \mathbf{0}, \\ & \mathbf{h}(\mathbf{x}(t), \mathbf{u}(t)) = \mathbf{0}, \end{aligned} \tag{1}$$

which attempts to find a control $\mathbf{u}(t)$ that minimizes the cost $\mathbf{J}$, while fulfilling the inequality constraints $\mathbf{c} \geq 0$ and the equality constraints $\mathbf{h} = 0$. In (1), the cost is designed as

$$\mathbf{J} := \sum_{t=1}^{T} L(\mathbf{x}(t), \mathbf{u}(t)) + \varphi(\mathbf{x}(T)),$$

where $L(\mathbf{x}(t), \mathbf{u}(t))$ indicates the Linear Quadratic Regulator (LQR) tracking cost of $\tilde{\mathbf{x}}(t)$ and $\varphi(\mathbf{x}(T))$ represents the terminal cost.

To avoid the high-cost computation over a full-order model, the Single Rigid Body (SRB) dynamics that relates $\mathbf{x}(t)$ and $\mathbf{u}(t)$ is employed as constraints in (1). Adapted from [21], the SRB dynamics constraints are designed as Newton-Euler Equations by separating the linear and orientation parts, which describes the state transition of the robot. In contrast to prior methods, such a design allows the algorithm to autonomously select gaits and foot swing phase trajectories. To ensure the feasibility of the generated trajectory, the contact complementary constraints

$$(p_i)_z \geq 0, \tag{2}$$
$$(f_i)_z (p_i)_z = 0, \tag{3}$$

are designed and incorporated in (1), where f_i denotes the force of the robot's ith foot with the subscripts x, y, z indicating the corresponding component of the vector. The constraint (2) restricts the foot position on the ground while the constraint (3) ensures that the feet are subject to force only when in contact with the ground. To allow the leg to fully extend downwards, the kinematic constraints are designed using L_2 norm as

$$0 < \left\| \begin{bmatrix} ({}^{B_i}(p_i))_x(t) \\ ({}^{B_i}(p_i))_y(t) \\ ({}^{B_i}(p_i))_z(t) \end{bmatrix} \right\|_2 \leq L_{max}, \tag{4}$$

where $({}^{B_i}(p_i))$ represents the position of ith foot in its corresponding shoulder frame and L_{max} represents the maximal extension length of the leg. By tracking the hand-crafted demonstration trajectory $\tilde{\mathbf{x}}(t)$ that roughly specifies the skill, we can obtain the trajectory of robot's joint positions using the generated $\mathbf{u}(t)$ from TO and the system dynamics and forward kinematics.

However, relying solely on the tracking of a single demonstration trajectory limits the exploratory nature of RL. To incorporate the motion prior from TO into our training process, the adversarial imitation learning is adopted. Following [22], we employ a discriminator D_ϕ, represented as a neural network parameterized by ϕ. The discriminator is trained to distinguish whether a state transition (s_t, s_{t+1}) originates from the motion prior dataset $\mathcal{D}$. The state s_t contains the base linear velocity, base angular velocity, the joint positions and velocities of the robot. The discriminator is trained with GCRL simultaneously. We adapte the training objective for the discriminator proposed in AMP [22]:

$$\begin{aligned} \arg\min_{\phi} \; & \mathbb{E}_{(s_t, s_{t+1}) \sim \mathcal{D}}[(D_\phi(s_t, s_{t+1}) - 1)^2] \\ & + \mathbb{E}_{(s_t, s_{t+1}) \sim \pi}[(D_\phi(s_t, s_{t+1}) + 1)^2] \\ & + \frac{\alpha}{2} \mathbb{E}_{(s_t, s_{t+1}) \sim \mathcal{D}}[\|\nabla_\phi D_\phi(s_t, s_{t+1})\|_2], \end{aligned} \tag{5}$$

where π is our goal-conditioned policy. The first two terms correspond to the least square GAN formulation.

This formulation aims to encourage the discriminator to distinguish whether the state transition originates from the loco-manipulation motion prior $\mathcal{D}$. The final term introduces a gradient penalty, controlled by coefficient α, which helps prevent the generator from overshooting and deviating from the data manifold.

B. Collision-Free Manipulation Module

For real-world loco-manipulation applications, the robotic arm needs to perform manipulation tasks while satisfying constraints such as avoiding collisions with the environment. Given the target position of the end effector $\mathbf{d_{com}}$ and the joint state of the robotic arm $\mathbf{q(t)}$, our objective is to compute the velocity command $\mathbf{a_t^{mani}}$ for the robotic arm to achieve collision-free manipulation. The considered manipulation module consists of two components: an MPPI-based trajectory generator and a vector field-based trajectory follower.

MPPI-based Trajectory Generator. Due to the dynamic nature of loco-manipulation, the target pose of the end effector is often hard to reach. It is common that there is no feasible solution when solving the inverse kinematics and thus inverse kinematic-based methods are not suitable. To address this issue, we adopt a sampling-based approach to ensure solution feasibility. To enable fast sampling, MPPI method [23] is utilized, which is formulated as a discrete-time, continuous-state system $\mathbf{q}(t+1) = \mathbf{q}(t) + \mathbf{e}(t)$, where $\mathbf{e}(t) \sim \mathcal{N}(\mathbf{d}(t), \Sigma)$, with $\mathbf{d}(t)$ representing a nominal displacement from the joint state $\mathbf{q}(t)$ at time t and Σ denoting its covariance. At each iteration, we sample M sequences of displacements $\mathbf{T}_j := \{\mathbf{e}_j(t_0), \ldots, \mathbf{e}_j(t_{H-1})\}, j = 1, \ldots, M$, given a set of nominal state $\mathbf{q}(t_0), \ldots, \mathbf{q}(t_H)$, and associated nominal displacements $\mathbf{d}(t_0), \ldots, \mathbf{d}(t_{H-1})$ over a control horizon H. The generator then evaluates these sampled trajectories through the model to compute the associated rollout costs $C(\mathbf{T_j})$. Adapted from [23], the posterior displacements are computed via exponential averaging after an MPPI iteration to compute the posterior displacements. Then we update the trajectory tracked by the follower with the new state rollout computed using the posterior displacements as [23].

The cost function $C(\mathbf{T_j})$ consists of three terms: a collision cost $C_{\text{coll}}(\mathbf{T_j})$ to penalize collision, an action cost $C_{\text{act}}(\mathbf{T_j})$ to penalize action norm, and a goal cost $C_{\text{goal}}(\mathbf{T_j})$ to penalize the distance to the target. Specifically, the cost is defined as $C = w_{\text{coll}} C_{\text{coll}} + w_{\text{act}} C_{\text{act}} + w_{\text{goal}} C_{\text{goal}}$, where

$$C_{\text{coll}}(\mathbf{T}_j) := \sum_{t=0}^{H-1} \frac{1}{\text{CSDF}_C(\mathbf{q}_j(t))}, \tag{6}$$

$$C_{\text{act}}(\mathbf{T}_j) := \sum_{t=0}^{H-1} \|\mathbf{e}_j(t)\|_2, \tag{7}$$

$$C_{\text{goal}}(\mathbf{T}_j) := \|f(\mathbf{q}_j(t_H)) - \mathbf{d}_{\text{cmd}}\|_2, \tag{8}$$

where $w_{\text{coll}}, w_{\text{act}}, w_{\text{goal}} \in \mathbb{R}^+$ are tunable weights, $\text{CSDF}_C(\cdot)$ is the C-SDF calculated by point cloud of the obstacles and the control points of the arm from [23], $\mathbf{q}_j$ is the joint state calculated by the sampled displacements, and $f(\cdot)$ is the forward kinematics of the robotic arm computed by curobo [24]. To improve the utilization of sampled data, we calculate the forward kinematics of the sampled joint angles and use the sampled end-effector positions to define the workspace.

Vector Field-based Trajectory Follower. To track the generated trajectory, the trajectory follower should output the joint velocity commands while ensuring obstacle avoidance. Specifically, the velocity commands $\mathbf{a_t^{mani}}$ are derived from the vector field as follows:

$$\mathbf{a_t^{mani}} := -k \nabla V(\mathbf{q}), \quad V(\mathbf{q}) := \frac{\|\mathbf{q} - \mathbf{r}^*\|^2 + \varepsilon}{\text{CSDF}_C(\mathbf{q}) + \varepsilon}. \tag{9}$$

Here, $\mathbf{r}^*$ represents the furthest point alone the trajectory, and $V(\mathbf{q})$ denotes the potential field, where $\varepsilon \in \mathbb{R}^+$ is a small positive value, and $k \in \mathbb{R}^+$ is a positive constant. To ensure collision-free trajectory following, collision-avoidance constraints are projected onto the null space of the constraint and a gradient term is incorporated to drive any deviation from the constraint manifold to zero. To minimize energy consumption, the robotic arm remains in its default position during locomotion. The velocity commands $\mathbf{a_t^{mani}}$ are executed only when the workspace encompasses the target position. Through asynchronous communication between the trajectory generator and trajectory follower, collision-free manipulation controller ensures computational efficiency and collision-avoidance during loco-manipulation.

C. Unified GCRL Framework

To integrate loco-manipulation motion priors with collision-free manipulation, we propose a unified GCRL framework to achieve loco-manipulation in a collaborative manner.

1) Policy Input: Given a target 6D pose for the end-effector $\mathbf{d_{cmd}}$, GCRL learns to track the target pose via a goal-conditioned policy. The policy observation $\mathbf{O_t}$ consists of the proprioception $\mathbf{x_t}$, the privileged state $\mathbf{x_t^P}$, and the history proprioception $\mathbf{x_t^H}$. The policy action $\mathbf{a_t^{loco}}$ represents joint position offsets, which determines the target positions for the leg joint motors.

2) Reward Terms: The reward function is designed as $r_t = w^{\text{task}} r_t^{\text{task}} + w^{\text{style}} r_t^{\text{style}}$, where $w^{\text{style}}, w^{\text{goal}} \in \mathbb{R}^+$ are tunable weights, and each term is explained in detail as follows.

Task Rewards: Unlike previous works [3]–[5], which track the base velocity and end-effector pose separately, our framework enables the robot to autonomously decide whether to prioritize locomotion or manipulation. To achieve this, we define the task reward as a function of the discrepancy between the sampled and commanded end-effector poses, i.e.,

$$r_t^{\text{task}} = \frac{1}{1 + \|\mathbf{d_{cmd}} - \mathbf{d_{sam}}\|^2} \cdot \frac{\mathbf{1}(t > T - T_r)}{T_r}, \tag{10}$$

where T is the episode length and T_r is a time threshold and $\mathbf{d_{sam}} := \min_j f(\mathbf{q}_j(t_H))$ is the position with the shortest distance from the target in the sampling point. This formulation ensures that the robot only needs to reach the goal before $T - T_r$ to maximize task rewards.

Style Rewards: To improve training efficiency, we incorporate the loco-manipulation motion prior and the discriminator designed in Sec. III-A into our learning framework. The style reward is defined as

$$r_t^{style}(s_t, s_{t+1}) = \max[0, 1 - 0.25(D(s_t, s_{t+1}) - 1)^2]. \tag{11}$$

3) Domain Randomization: To bridge the sim-to-real gap, we implement domain randomization to ease the transfer of learned behaviors from simulation to the real world. During training, at the command sampling stage, we randomize the base mass, PD controller gains, and introduce sampled noise to perturb end-effector pose commands. Finally, we train our policy using PPO [25] in IsaacGym [26].

IV. EXPERIMENTS AND RESULTS

A. Hardware Setup

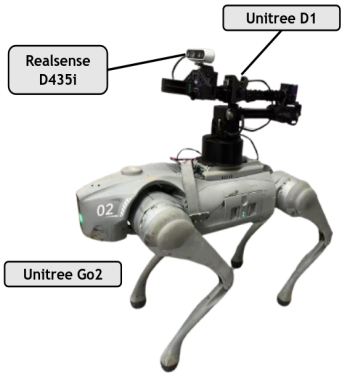

Fig.3: Overview of hardware.

Our robot system integrates a 12-DoF legged robot, the Unitree Go2, with a robotic arm, the Unitree D1. The D1 features 6 joints and is equipped with a parallel gripper. As illustrated in Fig. 3, the robotic arm is securely mounted on the back of the legged robot. Additionally, a RealSense D435i camera is installed above the arm's gripper. Throughout both training and deployment, the system operates at a control frequency of 50Hz, ensuring stable and responsive performance.

B. Simulation Experiment and Baselines

The objective of our simulation experiments is to address the following questions: 1) Can our proposed method effectively expand the operational workspace and enable seamless whole-body collaboration? 2) Do policies trained with loco-manipulation motion prior enhance workspace expansion more effectively than those trained with complex reward functions? 3) How do the tracking accuracy and obstacle avoidance of our collision-free manipulation compare with existing methods? To comprehensively evaluate our approach, we benchmark it against the following baseline approaches: (1) **Deep Whole-Body Control (DBC)** [3]: A unified RL-based policy trained with ROA and advantage mixing. (2) **RoboDuet** [4]: A learning-based framework that employs two collaborative policies for loco-manipulation. (3) **RL with Reward Fusion Module (RFM)** [27]: A RL paradigm that integrates task-specific rewards in a nonlinear manner. (4) **Ours w/o Collision-Free manipulation (w/o Collision-Free)**: A variant of our framework that replaces the collision-free manipulation with an inverse kinematics-based controller. (5) **Ours w/o Loco-manipulation Motion Prior (w/o Motion-Prior)**: A version of our framework trained without incorporating the loco-manipulation motion prior. (6) **Ours** : Our full unified framework which integrates both the loco-manipulation motion prior and the collision-free manipulation.

To quantitatively measure and compare the algorithm performance, the following key metrics are employed: (1) **Position Tracking Error**, (2) **Orientation Tracking Error**, (3) **Survival Rate (SR)**, (4) **Workspace (WS)**. A sample is considered successful if the end-effector tracking satisfies the following conditions: (i) the position error remains below 0.05; (ii) the orientation error is less than $\pi/18$, and (iii) the robotic arm avoids self-collision throughout the process.

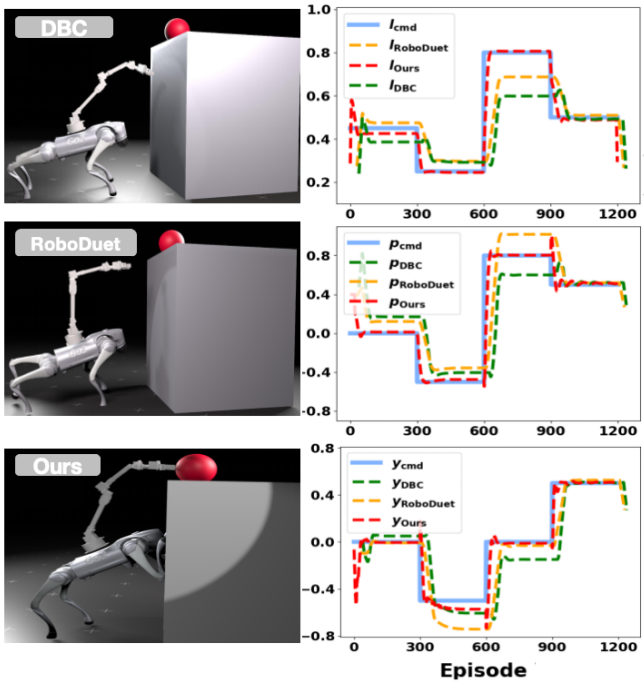

Fig.4: Performance Evaluation. Left: The performance of RoboDuet, DBC, and ours with the command $[l, p, y, roll, pitch, yaw] = [0.8, 0.8, 0, 0, 0, 0]$. Right: Trajectory tracking performance for fixed commands and sudden changes.

C. Main Results

1) Whole-Body Control: To evaluate the effectiveness of our unified framework, we compare it against DBC [3], RoboDuet [4], and RFM [27]. To ensure a fair comparison, we employ the average Euclidean distance error as a metric to assess both tracking accuracy and workspace expansion. In Table I, the columns represent the position tracking error, the orientation tracking error, the survival rate and the workspace range. As shown in Fig. 5 and Table I, our framework demonstrates state-of-the-art performance in tracking accuracy. Specifically, our method extends the workspace by 15.47% compared to the best-performing baseline. Additionally, our approach demonstrates superior tracking accuracy in both position and orientation. The workspace comparison further highlights that our whole-body control strategy significantly enhances the robotic arm's manipulation capabilities. This

Table I: Comparison of algorithms

Method	Position Error (m) ↓	Orientation Error (rad) ↓	SR (%) ↑	WS (m^3) ↑
DBC	0.11± 0.07	×	81.7	0.79
RoboDuet	0.08±0.05	0.37±0.09	88.5	0.82
RFM	0.05±0.03	0.09±0.03	87.3	0.84
w/o Collision-Free	0.05±0.02	**0.04±0.02**	73.3	0.89
w/o Motion Prior	0.23±0.16	0.42±0.20	36.8	0.70
Ours	**0.04±0.02**	0.05±0.02	**94.7**	**0.97**

improvement is not only theoretically meaningful but also practically valuable, as it expands the range of tasks that the robotic system can perform.

2) Survival Rate: In our experiment, robots perform loco-manipulation in environments cluttered with diverse obstacles. As shown in Fig. 5 and Table I, our method significantly boosts the robots' survival rate. We can see that DBC has the lowest survival rate due to its lack of collision considerations. In contrast, RoboDuet and RFM maintain relatively high survival rates by incorporating self-collision avoidance via reward functions. However, our collision-free manipulation goes further by incorporating external point cloud information, enabling it to account for collisions between the robotic arm, external obstacles, and the legged robot itself. This added awareness significantly enhances the system's robustness in complex environments.

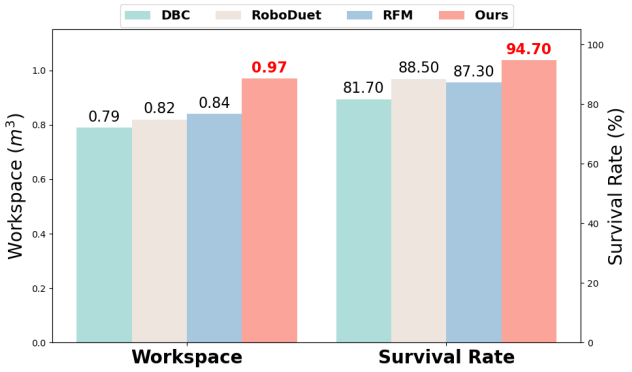

Fig.5: Comparison of workspace range and survival rate.

3) Ablation: Ablation studies are conducted in this section to assess the contribution of different modules in our framework and evaluate their impact on overall performance. Specifically, we compare our unified framework with two variants. The results, presented in Table I, highlight the following insights. First, our collision-free manipulation can effectively prevent robotic arm collisions during manipulation. However, it introduces a slight decrease in end-effector orientation tracking accuracy. Second, integrating loco-manipulation motion priors facilitates the development of agile loco-manipulation skills, significantly enhancing the system's overall performance. These experiments confirm the value of each module, providing valuable insights for further optimization of our framework.

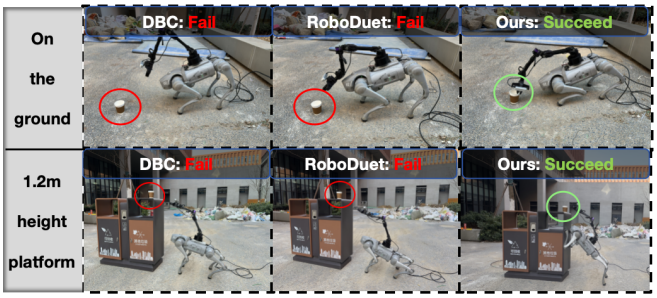

Fig.6: Tracking 6D poses with DBC, RoboDuet, and our framework in different terrains.

D. Real-World Deployment

For real-world experiments, we design two distinct terrains to assess our policy's effectiveness in practical scenarios and its ability to handle various loco-manipulation challenges. We implement DBC, RoboDuet, and our method directly on a physical robotic system for comparative analysis.

To show the expanded workspace and safety of our method in unstructured environments, we introduce obstacles both on elevated platforms and on the ground while setting target poses among them. As shown in Fig. 6, our method significantly improves the robotic arm's workspace expansion while avoiding collisions. Notably, our robot exhibits remarkable adaptability, even scaling walls or lowering itself to the ground to facilitate precise manipulation. This demonstrates the adaptability and versatility of our framework in complex real-world applications.

To comprehensively evaluate the effectiveness of our framework, we establish a long-horizon end-effector pose tracking task, incorporating target points of varying heights and directions. As illustrated in Fig. 7, our framework is capable of generating a diverse range of motion patterns that were previously unattainable. These include wall-climbing and ground-crawling motions, which cannot be attained merely by adjusting the robot's pitch angle. This showcases not only the unique capabilities of our framework but also the effectiveness of our GCRL training paradigm. Moreover, when switching targets, the robot can automatically transitions between manipulation and locomotion states. After approaching the target, the robot can smoothly switch from locomotion to manipulation, thereby achieving fully autonomous loco-manipulation. A key advantage of our GCRL paradigm is its ability to eliminate the need for teleoperating the base velocity and end-effector position. Additionally, during locomotion, the robotic arm remains in a safe state to prevent collision. The arm is only activated for manipulation

Fig.7: Long-horizon trajectory tracking. The robot can switch between locomotion and manipulation to track the target end-effector pose.

once the robot reaches the target point, ensuring an optimal balance between mobility and precision. This result further validates the effectiveness of our collision-free manipulation.

V. CONCLUSIONS

In this work, we present a unified framework for learning collision-free loco-manipulation with AMP. The loco-manipulation motion prior and AMP enhance the robot's ability to acquire flexible and adaptive skills. The collision-free manipulation ensures safety throughout the loco-manipulation process. Future work will focus on high-level regulation and decision-making strategies for complex loco-manipulation tasks.

REFERENCES

[1] T. Qian, Z. Zhou, S. Wang, Z. Li, C.-Y. Su, and Z. Kan, "Vision-based reactive planning and control of quadruped robots in unstructured dynamic environments," 2023. [Online]. Available: https://arxiv.org/abs/2307.10243

[2] R. S. Sutton and A. G. Barto, "Reinforcement learning: An introduction," *Robotica*, vol. 17, no. 2, pp. 229–235, 1999.

[3] Z. Fu, X. Cheng, and D. Pathak, "Deep whole-body control: learning a unified policy for manipulation and locomotion," in *Proc. Mach. Learn. Res.* PMLR, 2023, pp. 138–149.

[4] G. Pan, Q. Ben, Z. Yuan, G. Jiang, Y. Ji, S. Li, J. Pang, H. Liu, and H. Xu, "Roboduet: Whole-body legged loco-manipulation with cross-embodiment deployment," 2024. [Online]. Available: https://arxiv.org/abs/2403.17367

[5] M. Liu, Z. Chen, X. Cheng, Y. Ji, R. Qiu, R. Yang, and X. Wang, "Visual whole-body control for legged loco-manipulation," *The 8th Conference on Robot Learning*, 2024.

[6] H. Ha, Y. Gao, Z. Fu, J. Tan, and S. Song, "Umi on legs: Making manipulation policies mobile with manipulation-centric whole-body controllers," 2024. [Online]. Available: https://arxiv.org/abs/2407.10353

[7] M. Liu, M. Zhu, and W. Zhang, "Goal-conditioned reinforcement learning: Problems and solutions," 2022. [Online]. Available: https://arxiv.org/abs/2201.08299

[8] T. Qian, H. Zhang, Z. Zhou, H. Wang, M. Cai, and Z. Kan, "Leeps: Learning end-to-end legged perceptive parkour skills on challenging terrains," in *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2024, pp. 12 904–12 909.

[9] S. Kareer, N. Yokoyama, D. Batra, S. Ha, and J. Truong, "ViNL: Visual Navigation and Locomotion Over Obstacles," in *International Conference on Robotics and Automation (ICRA)*, 2023.

[10] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "Rma: Rapid motor adaptation for legged robots," *arXiv preprint arXiv:2107.04034*, 2021.

[11] A. Agarwal, A. Kumar, J. Malik, and D. Pathak, "Legged locomotion in challenging terrains using egocentric vision," in *Proc. Mach. Learn. Res.* PMLR, 2023, pp. 403–415.

[12] Z. Zhuang, Z. Fu, J. Wang, C. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao, "Robot parkour learning," in *Proc. Mach. Learn. Res.*, 2023.

[13] X. Cheng, K. Shi, A. Agarwal, and D. Pathak, "Extreme parkour with legged robots," *arXiv preprint arXiv:2309.14341*, 2023.

[14] T. He, C. Zhang, W. Xiao, G. He, C. Liu, and G. Shi, "Agile but safe: Learning collision-free high-speed legged locomotion," in *Robotics: Science and Systems (RSS)*, 2024.

[15] A. Escontrela, X. B. Peng, W. Yu, T. Zhang, A. Iscen, K. Goldberg, and P. Abbeel, "Adversarial motion priors make good substitutes for complex reward functions," 2022. [Online]. Available: https://arxiv.org/abs/2203.15103

[16] Y. Fuchioka, Z. Xie, and M. van de Panne, "Opt-mimic: Imitation of optimized trajectories for dynamic quadruped behaviors," 2023. [Online]. Available: https://arxiv.org/abs/2210.01247

[17] J. Wu, G. Xin, C. Qi, and Y. Xue, "Learning robust and agile legged locomotion using adversarial motion priors," *IEEE Robotics and Automation Letters*, vol. 8, no. 8, pp. 4975–4982, 2023.

[18] J.-P. Sleiman, F. Farshidian, M. V. Minniti, and M. Hutter, "A unified mpc framework for whole-body dynamic locomotion and manipulation," 2021. [Online]. Available: https://arxiv.org/abs/2103.00946

[19] Z. Fu, X. Cheng, and D. Pathak, "Deep whole-body control: Learning a unified policy for manipulation and locomotion," in *Conference on Robot Learning (CoRL)*, 2022.

[20] 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," 2024. [Online]. Available: https://arxiv.org/abs/2411.15131

[21] A. W. Winkler, "Optimization-based motion planning for legged robots," Ph.D. dissertation, ETH Zurich, 2018.

[22] X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa, "Amp: adversarial motion priors for stylized physics-based character control," *ACM Transactions on Graphics*, vol. 40, no. 4, 2021. [Online]. Available: http://dx.doi.org/10.1145/3450626.3459670

[23] V. Vasilopoulos, S. Garg, P. Piacenza, J. Huh, and V. Isler, "RAMP: Hierarchical Reactive Motion Planning for Manipulation Tasks Using Implicit Signed Distance Functions," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023.

[24] B. Sundaralingam, S. K. S. Hari, A. Fishman, C. Garrett, K. V. Wyk, V. Blukis, A. Millane, H. Oleynikova, A. Handa, F. Ramos, N. Ratliff, and D. Fox, "curobo: Parallelized collision-free minimum-jerk robot motion generation," 2023. [Online]. Available: https://arxiv.org/abs/2310.17274

[25] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017. [Online]. Available: https://arxiv.org/abs/1707.06347

[26] V. Makoviychuk, L. Wawrzyniak, Y. Guo, M. Lu, K. Storey, M. Macklin, D. Hoeller, N. Rudin, A. Allshire, A. Handa, and G. State, "Isaac gym: High performance gpu-based physics simulation for robot learning," 2021. [Online]. Available: https://arxiv.org/abs/2108.10470

[27] K. Jiang, Z. Fu, J. Guo, W. Zhang, and H. Chen, "Learning whole-body loco-manipulation for omni-directional task space pose tracking with a wheeled-quadrupedal-manipulator," 2024. [Online]. Available: https://arxiv.org/abs/2412.03012