Versatile Loco-Manipulation through Flexible Interlimb Coordination

Xinghao Zhu[*,1], **Yuxin Chen**[*,1,2], **Lingfeng Sun**[*,1], **Farzad Niroui**[1]
Simon Le Cleac'h[1], **Jiuguang Wang**[1], **Kuan Fang**[1,3]
[*]Denotes equal contribution
[1]*RAI Institute* [2]*University of California, Berkeley* [3]*Cornell University*

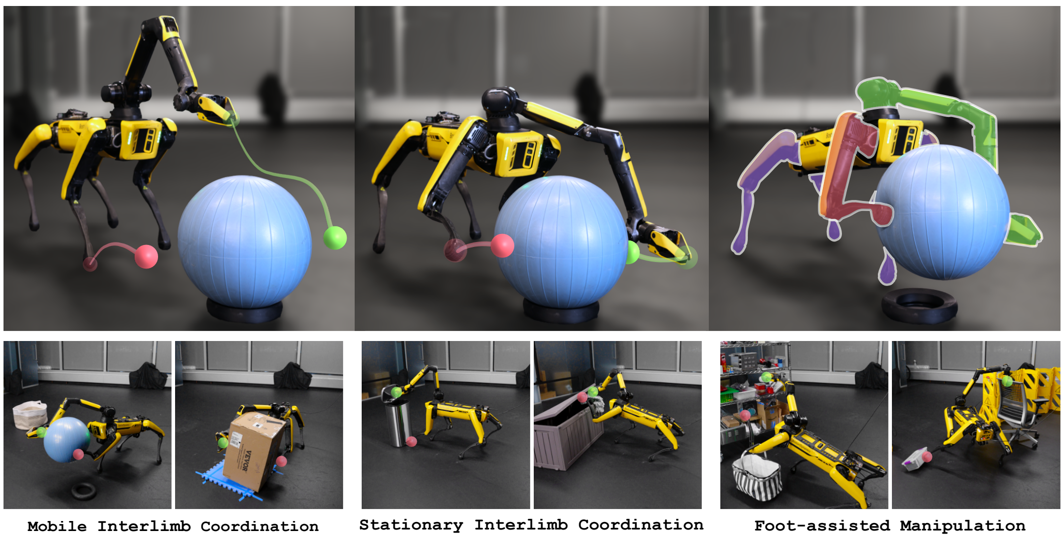

Figure 1: We present ReLIC, a learning-based approach that enables flexible interlimb coordination for versatile loco-manipulation in unstructured environments. ReLIC controls an arm-mounted quadruped robot to dynamically assign limb for either locomotion or manipulation during task execution, e.g., simultaneously using an arm (*green*) and a selected leg (*red*) to jointly reach manipulation targets while maintaining stable mobility with the remaining legs (*purple*). The effectiveness of our approach is demonstrated across diverse and complex tasks specified by various user input modalities, including direct target specification, contact points, and natural language instructions.

Abstract: The ability to flexibly leverage limbs for loco-manipulation is essential for enabling autonomous robots to operate in unstructured environments. Yet, prior work on loco-manipulation is often constrained to specific tasks or predetermined limb configurations. In this work, we present Reinforcement Learning for Interlimb Coordination (ReLIC), an approach that enables versatile loco-manipulation through flexible interlimb coordination. The key to our approach is an adaptive controller that seamlessly bridges the execution of manipulation motions and the generation of stable gaits based on task demands. Through the interplay between two controller modules, ReLIC dynamically assigns each limb for manipulation or locomotion and robustly coordinates them to achieve the task success. Using efficient reinforcement learning in simulation, ReLIC learns to perform stable gaits in accordance with the manipulation goals in the real world. To solve diverse and complex tasks, we further propose to interface the learned controller with different types of task specifications, including target trajectories, contact points, and natural language instructions. Evaluated on 12 real-world tasks that require diverse and complex coordination patterns, ReLIC demonstrates its versatility and robustness by achieving a success rate of 78.9% on average. Videos and code can be found at https://relic-locoman.rai-inst.com/.

Keywords: Loco-Manipulation, Whole-Body Control, Reinforcement Learning

1 Introduction

The ability to combine locomotion and manipulation, commonly referred to as *loco-manipulation*, is essential to achieving robot autonomy in unstructured environments [1]. As tasks grow in diversity and complexity, robots must perform versatile interactions with both the terrain and external objects, coordinating limb movements in concert. Despite significant advances in locomotion and manipulation individually, seamlessly integrating the two capabilities remains a major open challenge [2, 3]. Consider, for example, an arm-mounted quadruped robot tasked with transporting a large yoga ball across a room, as shown in Figure 1. Success of this task demands coordinating the arm and a leg to grasp, lift, and balance the object, while maintaining stable gaits to move around with the remaining support legs. Different tasks impose different contact and movement requirements, necessitating adaptive whole-body control strategies that can flexibly allocate actuation across limbs.

In light of this need, prior work has explored a range of decomposition and coordination strategies. A common approach partitions control among limb groups, typically treating arms and legs separately, based on task-specific heuristics [4, 5]. While effective in constrained settings such as mobile pick-and-place, these methods often depend heavily on manual design and struggle to generalize to tasks requiring dynamic assignment of limb functions. More recent efforts have pursued whole-body control, spanning both model-based and learning-based approaches [2, 6, 7]. Despite encouraging results, these methods are often confined to predefined task sets and fixed limb roles. For more complex settings, jointly optimizing for both locomotion and manipulation within a unified framework remains a fundamental challenge in practice.

To this end, we present **Re**inforcement **L**earning for **I**nterlimb **C**oordination (**ReLIC**), an approach for solving versatile loco-manipulation in unstructured environments. At the core of ReLIC is an adaptive controller that bridges manipulation success with locomotion stability under dynamic assignments of limb functions. Instead of solving both objectives holistically or relying on fixed decomposition heuristics, our method decouples loco-manipulation into two interconnected subproblems, robustly generating manipulation behaviors and maintaining stable gaits based on task demands. We train the locomotion controller entirely in simulation using an efficient reinforcement learning pipeline and then transfer it to the real world through motor calibration. Built on top of this adaptive controller, our approach supports flexible task specification via high-level task interfaces, including direct targets, contact points, and free-form language instructions. As illustrated in Figure 1, ReLIC enables a broad range of loco-manipulation tasks requiring seamless interlimb coordination. Deployed on an arm-mounted Boston Dynamics Spot, ReLIC demonstrates robust performance across 12 diverse tasks involving mobile interlimb manipulation, stationary interlimb coordination, and foot-assisted manipulation. We achieve an overall success rate of 78.9% across the three different types of task specifications.

2 Related Work

Joint locomotion and manipulation has been widely studied in robotics, with applications across diverse hardware platforms including wheeled mobile manipulators [8–16], humanoid robots [17–20], and legged systems [4, 6, 7, 21–29]. Model-based approaches typically rely on whole-body control frameworks and trajectory optimization, leveraging accurate dynamics models and structured task representations [6, 21–23]. Assuming detailed knowledge of the scene, object geometry, and contact conditions [4, 24–26], most of these approaches operate in controlled settings and often struggle to scale to unstructured environments. On the other hand, learning-based methods offer improved adaptability and reduced engineering overhead by leveraging large-scale, high-quality data [7, 27–29]. However, most are limited to specific tasks or assume fixed limb roles, due to the complex nature of learning such capabilities and the limited data [30, 31]. Our approach follows a task-oriented design that decouples the loco-manipulation problem solved through an integration of a model-based controller and trained RL policy. This separation enables more flexible reuse of learned behaviors across tasks while supporting dynamic reallocation of limb roles in response to task demands.

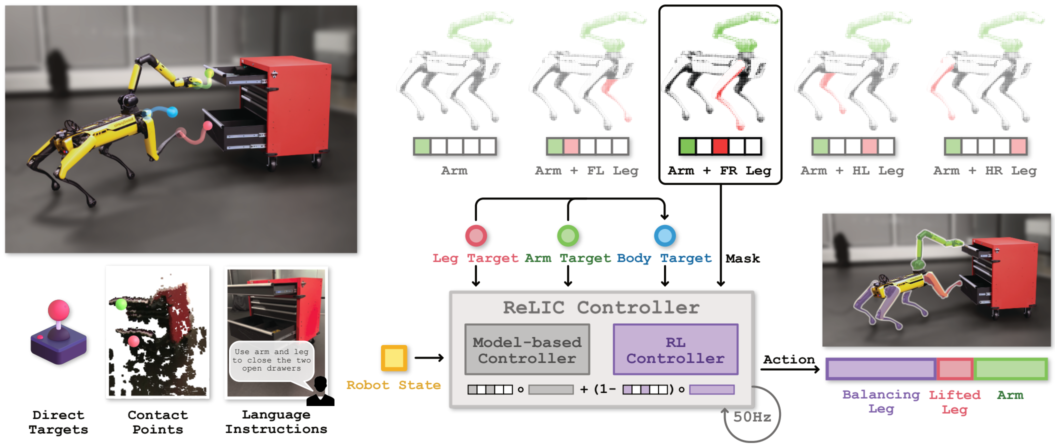

Figure 2: **Overview of ReLIC.** Based on various types of task specifications, ReLIC enables the robot to perform versatile loco-manipulation. Conditioned on the dynamic assignment of limb functions, the ReLIC controller generates the actions through the interplay between a model-based module that prioritizes task success, and an RL policy that robustly maintain stable gaits in accordance with the manipulation behaviors. This design decouples the two challenging subproblems without relying on rigid heuristics or sacrificing inter-module coordination.

Interlimb coordination is a hallmark of natural locomotion and dexterous behavior in animals and humans [32–34]. In robotics, coordinated use of multiple limbs has been explored individually for multi-legged locomotion [35] and bimanual manipulation [36, 37]. Recent work has investigated using legs as manipulators to extend task versatility [38–42]. However, these systems often rely on predefined contact sequences with explicit mode switching, sometimes requiring additional actuation hardware. Notably, Sleiman et al. [2, 30] propose whole-body planning for contact-rich tasks using pre-modeled scenes, discovering feasible contact modes offline. While effective in pre-defined settings, these methods usually assume static assignments of limb functions and known objects, limiting their applicability in new tasks or unseen environments. In contrast, our approach enables flexible interlimb coordination, allowing each limb to dynamically alternate between locomotion and manipulation based on the evolving requirement for diverse and complex tasks.

3 Method

Our goal is to control a legged manipulator to flexibly utilize its limbs to solve manipulation tasks while stably performing locomotion across diverse scenarios. While our approach is designed to generalize to robots with different arm–leg configurations, our experiments and examples focus on a quadrupedal robot equipped with a single arm. Concretely, the set of available limbs is denoted as $\Lambda = \{Arm, FL\text{-}Leg, FR\text{-}Leg, HL\text{-}Leg, HR\text{-}Leg\}$, where F, H, L, and R indicate front, hind, left, and right respectively. In contrast to prior work [2], our approach enables each limb to dynamically switch roles between locomotion and manipulation in response to the task demand.

We present Reinforcement Learning for Interlimb Coordination (ReLIC), an approach for versatile loco-manipulation through flexible interlimb coordination (Figure 2). In this section, we first provide an overview of our framework for versatile loco-manipulation (Section 3.1). Next, we propose an adaptive controller by leveraging reinforcement learning (RL) to bridge locomotion and manipulation (Section 3.2). We then explain how to effectively learn a robust and transferable policy from simulation for our proposed controller (Section 3.3). Lastly, we describe how the ReLIC controller can be interfaced with various types of user specifications to perform diverse and complex loco-manipulation tasks (Section 3.4).

3.1 Overview

We propose to solve versatile loco-manipulation in a hierarchical framework consisting of a *task level* and a *command level*. The former aims to describe diverse and complex loco-manipulation tasks, as shown in Figure 3, using a unified target-driven representation, while the latter controls the robot to achieve the task-specific targets while maintaining stable mobility through flexible coordination of limbs. Below, we elaborate on the design and interface for each level.

Task level. Based on the user description, we assume the task solution can be represented as end-effector targets for designated limbs over time. At each time step, we use a binary mask $m \in \{0, 1\}^{|\Lambda|}$ to indicate the role for each limb $\lambda \in \Lambda$, with manipulation indicated by 1 and locomotion by 0. Given the assignment mask m, the target pose for each limb end-effector designated for manipulation is specified as τ^{λ}, leaving the remaining limbs for locomotion with unspecified targets. Accordingly, the desired torso target τ^{torso} is determined by the manipulation limb targets via whole-body IK [43], all together denoted as τ. Now, the task of the horizon T can be represented as the sequence of $\{\tau_t, m_t\}$ for $t = 0, ..., T-1$. In Section 3.4, we will explain how such task representation can be computed from the user descriptions of various modalities.

Command level. Given the representation from the task level, we design the controller to generate motor commands for the robot to perform versatile loco-manipulation as shown in Figure 2. On the Spot robot, low-level commands include the desired position for each joint. Next, we will explain how to obtain the controller generating commands based on $\{\tau_t, m_t\}$ at each time step.

3.2 Adaptive Control for Flexible Interlimb Coordination

To tackle the substantial challenge of solving whole-body control under dynamic limb assignments, we design the ReLIC controller to seamlessly bridge locomotion and manipulation. Unlike prior work that learns a monolithic policy end-to-end [7, 30, 39, 41], our approach generates actions through the interaction of two dedicated controller modules: a manipulation module that prioritizes task success, and a locomotion module that maintains stable gaits in accordance with the manipulation behaviors. The two modules in the ReLIC controller communicate through the up-to-date whole-body robot state s_t and the limb assignment m to jointly predict the motor commands as the action a. This design decouples the two challenging subproblems without relying on rigid heuristics or sacrificing inter-module coordination.

According to the different needs for manipulation and locomotion, we adopt different design options for the two controller modules. Given the target-driven task representation, the manipulation task can be directly solved by a model-based (MB) controller conditioned on s, m, and τ. Depending on the task requirements, this can be anything from a standard inverse kinematics solver to a meticulously tuned impedance controller. As for locomotion, which involves dynamic behaviors with significantly higher demands for robustness and adaptability, we train a policy $\pi(\cdot|s, m)$ through reinforcement learning (RL) [44]. Concretely, we denote the action yielded by the two modules as a_{MB} and a_{RL}, both of which have the same dimensionality with the final action a with only the dimensions for the assigned limbs to be valid. To this end, the final action is computed as $a = m \circ a_{\text{MB}} + (1 - m) \circ a_{\text{RL}}$.

To ensure stable locomotion across different limb coordination patterns, we exert gait regularization during training by leveraging contact-time metrics among feet [44, 45]. Specifically, we enforce trotting gait for quadrupedal locomotion and a three-phase bouncing gait for tripedal locomotion, as visualized in Figure 4. Compared to phase-based gait regularization [46], the contact-time-based approach is simpler to implement and more stable during training, as it avoids sampling from a state-dependent phase variable. Implementation details are provided in the Appendix.

3.3 Learning Transferrable Policy in Simulation

We leverage IsaacLab [44] to scale up reinforcement learning for improving the robustness of the policy π. To facilitate sim-to-real transfer, we implement comprehensive domain randomization during training, covering variations in robot dynamics, terrain properties, and external disturbances.

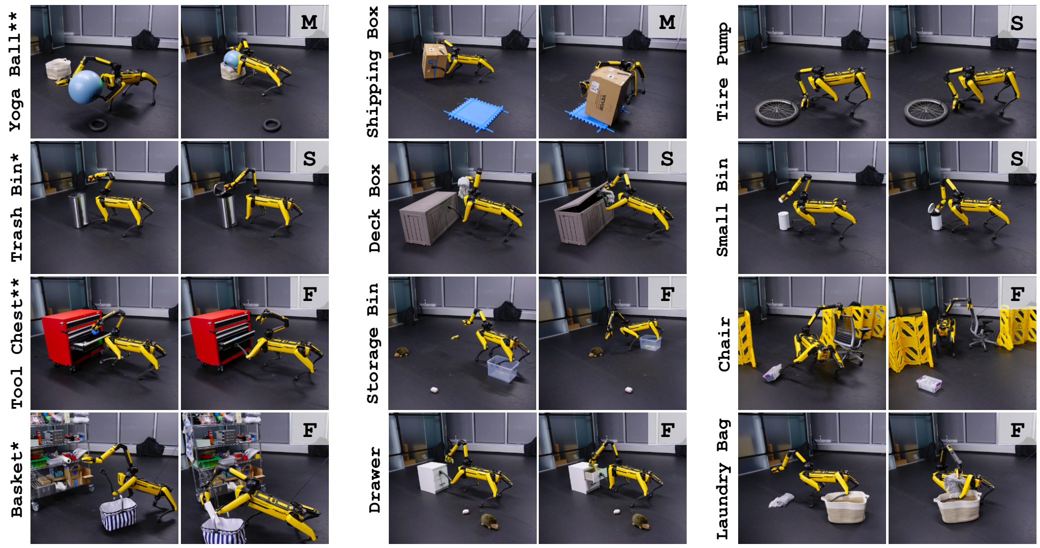

Figure 3: **Loco-Manipulation Tasks with Interlimb Coordination.** ReLIC is evaluated on 12 real-world tasks designed to test diverse and complex interlimb coordination. The task suite spans three categories: mobile interlimb coordination (**M**), stationary interlimb coordination (**S**), and foot-assisted manipulation (**F**). All tasks can be specified using direct target inputs, with a subset also supports specification via contact points (*) and language instructions (**).

Despite these efforts, the dynamic behaviors required for flexible interlimb coordination, particularly during three-legged locomotion, reveal a significant sim-to-real deployment gap which primarily stems from unmodeled variations in motor parameters [47] such as time-dependent torque limits.

To bridge this gap, we leverage a motor calibration procedure by utilizing rollouts from the real world. After the initial training in simulation with uncalibrated parameters, the policy is deployed on the real-world robot to collect extensive calibration data, including joint positions, velocities, commanded torques, and actual torque measurements. Using CMA-ES [48], the torque limits are optimized as functions of joint state from collected real-world data. The policy is subsequently fine-tuned in simulation using these calibrated parameters.

3.4 Task Interface for Versatile Loco-Manipulation

Leveraging the flexibility of our formulation and the adaptability of the ReLIC controller, we solve diverse and complex loco-manipulation tasks based on user commands. We interface with user input at the task level through three modalities:

- **Direct targets:** The most straightforward approach is to manually specify target trajectories for designated limbs, such as through teleoperation. These trajectories can be directly converted into the task representation described in Section 3.1 with minimal post-processing.
- **Contact points.** Many loco-manipulation tasks can be described by specifying key contact points and associated motions. Given these, target trajectories are generated via motion planning algorithms.
- **Language instructions.** Free-form language provides a flexible way to describe complex tasks. While our controller is not directly conditioned on language, contact points and trajectories can be inferred from RGBD observations using vision-language models (VLMs) [49], which provide strong semantic reasoning capabilities.

These different modalities reflect varying levels of user specification and prior knowledge, resulting in different levels of difficulty for equivalent tasks expressed via different modalities. Additional implementation details for each interface modality are provided in the Appendix.

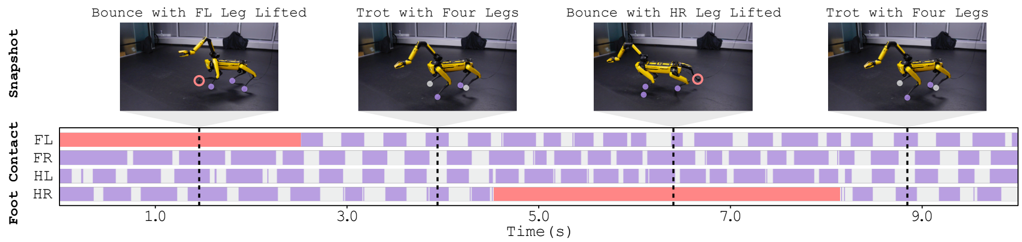

Figure 4: **Flexible Gait Transitions.** With ReLIC, the robot can execute a range of gaits, including four-legged trotting and three-legged bouncing with a designated limb lifted (FL: front-left, FR: front-right, HL: hind-left, HR: hind-right). The controller enables seamless transitions between these gait modes in real-world operation, without requiring the robot to pause or reset its stance.

4 Experiments

We conduct a series of real-world experiments to investigate the following questions: **Q1.** Can the ReLIC controller robustly perform interlimb coordination? (Section 4.1) **Q2.** Can ReLIC successfully perform diverse and complex loco-manipulation tasks on command? (Section 4.2) **Q3.** What are the primary sources of failure within the control stack? (Section 4.3)

Experiments are run on a Boston Dynamics *Spot* (quadrupedal, 12 actuators) equipped with *Spot Arm* and a gripper (7 actuators total). The onboard stereo cameras and IMU supply state estimates and RGB-D images for the task interface. All computation executes on an external PC connected over Ethernet. Additional hardware details appear in the Appendix.

4.1 Quantitative Analysis on Interlimb Coordination

We first conduct quantitative analysis on the ReLIC controller's performance on interlimb coordination. Specifically, we examine the robot's ability to (i) switch gaits on demand under varying arm and leg configurations and (ii) perform target-driven end-effector motions with multiple limbs.

Gait transitions. We command the robot to switch between four-legged trotting and three-legged bouncing, while randomly changing the end-effector targets during execution. As shown in Figure 4, the robot adapts its gait dynamically to maintain balance throughout the transitions. All gait switches occur instantly, without requiring the robot to stop or transition through a predefined mode. We observe that the robot can immediately reassign the lifted limb from FL to HR during walking without interruption. In contrast, prior work [38–40, 42] for leg manipulation rely on predefined state machines or pre-trained gait policies often requiring the robot to pause before switching limbs.

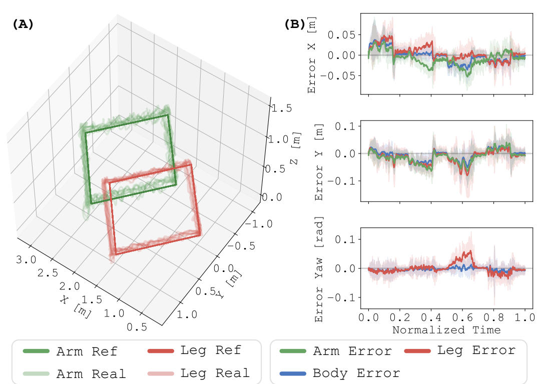

Figure 5: **End-Effector Motions.** The robot tracks independent rhombus-shaped trajectories using the arm and the lifted front-left (FL) leg while walking with three supporting limbs. **(A)** Overlay of reference and executed trajectories for both effectors. **(B)** Linear tracking errors in x and y directions and rotational error in yaw.

End-effector motions. While walking with three support legs, the robot is asked to use the arm and the lifted FL leg to independently track rhombus-shaped trajectories ten times in succession. Figure 5(A) overlays the reference and executed paths, and Figure 5(B) reports the linear tracking errors in x, y, and the rotational error for yaw. The low mean Cartesian error across both effectors confirms that the controller maintains precise interlimb coordination during dynamic locomotion.

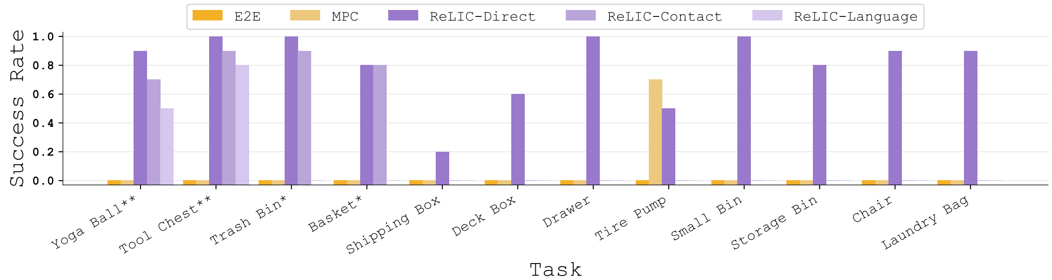

Figure 6: **Loco-Manipulation Task Performances.** We evaluate ReLIC on 12 real-world loco-manipulation tasks, reporting the success rate over 10 trials per task. Our three ReLIC variants consistently outperform the baselines, demonstrating the versatility and robustness of our controller. In contrast, the end-to-end RL (E2E) and MPC baselines fail on most tasks, highlighting the challenges of dynamic interlimb coordination and the advantage of our approach.

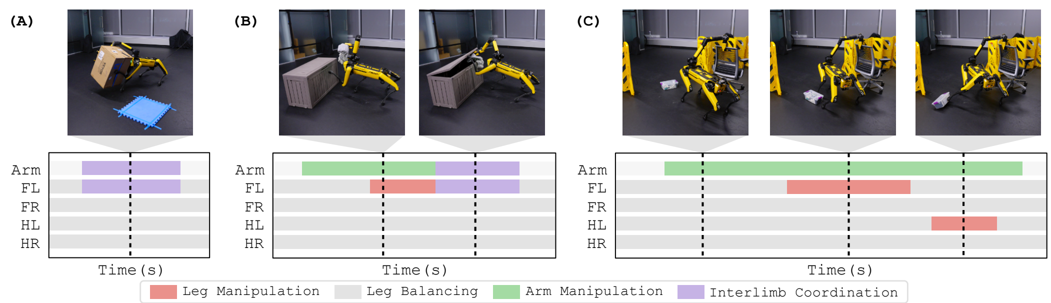

Figure 7: **Flexible Interlimb Coordination.** ReLIC enables dynamic assignments of limbs between manipulation and locomotion during task execution. This seamless role-switching allows the robot to adapt efficiently to varying task demands. Here, we demonstrate the different interlimb coordination patterns during the execution of *Shipping Box* **(A)**, *Deck Box* **(B)**, and *Chair* **(C)** tasks.

4.2 Loco-Manipulation Task Solving

Tasks. As shown in Figure 3, we devise 12 tasks in the real world to demonstrate the capability of ReLIC and conduct comprehensive evaluation, examining the robot's capability involving diverse and complex interlimb coordination patterns (detailed task setup can be found in Appendix):

- **Mobile interlimb coordination:** *Yoga Ball* and *Shipping Box* test the robot's ability to manipulate large objects using its arm and one leg while navigating with the remaining three legs.
- **Stationary interlimb coordination:** In the *Tire Pump*, *Trash Bin*, *Deck Box*, and *Small Bin* tasks, the robot coordinates its arm with one designated leg for object manipulation while maintaining balance through static support from the remaining three legs.
- **Foot-assisted manipulation:** While the *Tool Chest*, *Storage Bin*, *Chair*, *Basket*, *Drawer*, and *Laundry Bag* tasks can be completed using only the arm, incorporating an additional leg as a manipulator demonstrates measurable performance improvements in stability and task execution.

Model variants and baselines. We evaluate three variants of our method based on different user input modalities: direct targets (**ReLIC-Direct**), contact points (**ReLIC-Contact**), and language instructions (**ReLIC-Language**). ReLIC-Contact and ReLIC-Language are evaluated on a subset of tasks due to input modality constraints. All variants uses the same trained ReLIC controller. As baselines, we compare against an end-to-end reinforcement learning policy [7] (**E2E**) and a model predictive control (**MPC**) policy conditioned on direct targets.

Results. We evaluate ReLIC on the 12 loco-manipulation tasks in unstructured real-world environments. Each model variant and baseline is evaluated over 10 randomized trials per task. The average success rate for each settings is reported in Figure 6.

ReLIC-Direct achieves the highest success rates across all but one task. In 9 out of 12 tasks, it succeeds in more than 8 out of 10 trials, covering all three task categories. Most tasks involve long-horizon execution with multiple stages, requiring diverse capabilities such as object picking (e.g., *Basket*, *Small Bin*), displacement (e.g., *Storage Bin*, *Drawer*), locomotion (e.g., *Yoga Ball*, *Laundry Bag*), and maintaining forceful contact (e.g., *Trash Bin*, *Chair*). The consistently high performance demonstrates the robustness and reliability of the ReLIC controller. On tasks with additional challenges of repetitive motion or fine balance, like *Deck Box* and *Tire Pump*, ReLIC-Direct maintains a non-negligible success rate, though performance is reduced. Among all tasks, *Shipping Box* proves the most difficult due to the large size and rigid-body dynamics of the object. Despite receiving less direct guidance, both ReLIC-Contact and ReLIC-Language achieve comparable results. In these settings, the controller reliably executes targets inferred user-specified contact points or language instructions, validating the effectiveness of our hierarchical framework.

Baseline methods underperform across the board due to their lack of robust interlimb coordination. The off-the-shelf MPC baseline lacks support for interlimb manipulation or three-leg locomotion. As a result, it fails on all tasks except for *Tire Pump*. This success occurs only by chance when the robot happens to step on the pump and actuate it with its arm, without coordinated intent. The end-to-end RL baseline fails across all tasks due to unstable gait generation and inaccurate end-effector tracking, underscoring the need for more structured and adaptive control strategies.

4.3 Failure Analysis

With ReLIC-Contact as a test case, we analyze failure modes with a breakdown illustrated in Figure 8. The first source of failure for ReLIC-Contact comes from perception error, primarily due to inaccuracies in the 3D point cloud or state estimation. Even when perception is accurate, task failures may still arise for tracking performed by the ReLIC controller, particularly under extreme or unstable body configurations. For example, when the robot attempts to reach an arm target while one foot is stepping on the trash bin pedal, the unpredictable external force from the pedal poses significant challenges for the policy to maintain balance. The third category of failures involves unintended manipulation errors with challenging contacts. A common example for this category occurred in the *Yoga Ball* task, where the robot either lost grip or unintentionally kicked the ball, resulting in slippage. Such failures are especially challenging to avoid or recover from, as a viable solution would require real-time feedback and fine-grained contact reasoning. Despite these challenges, ReLIC achieves an overall success rate of 82.5% across the four most contact-intensive loco-manipulation tasks.

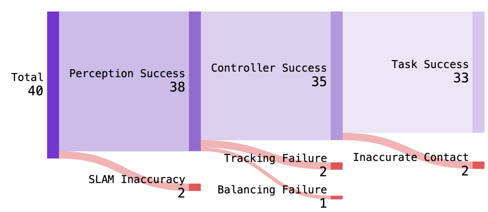

Figure 8: **Failure Breakdown.** Analysis of failure modes for the ReLIC-Contact variant across multiple tasks. Failures are categorized into SLAM errors, tracking errors, balance loss, and inaccurate contact.

5 Conclusion and Discussion

We introduced ReLIC, an approach that enables flexible interlimb coordination for versatile loco-manipulation. By dynamically assigning limb roles and decoupling locomotion and manipulation into two coordinated modules, ReLIC learns to achieve robust whole-body control. Based on task specifications of various modalities, the ReLIC controller can be employed to solve diverse and complex tasks on command. Our system demonstrates strong real-world performance on 12 challenging scenarios, spanning mobile interlimb coordination, stationary interlimb coordination, and foot-assisted manipulation. We hope this work inspires further research at the intersection of reinforcement learning and whole-body control, and highlights the critical role of interlimb coordination in advancing loco-manipulation capabilities.

6 Limitations

While ReLIC demonstrates versatile whole-body control across a diverse set of loco-manipulation tasks, several limitations remain.

First, the high-level task interface based on contact points and language instructions generates targets in an open-loop fashion. Although effective in the evaluated scenarios, this approach can be brittle in tasks requiring real-time adaptation or fine-grained feedback. A promising direction is to collect large-scale demonstrations and train policies to predict task-level actions through imitation learning.

Second, the manipulation controller currently relies on a standard inverse kinematics solver. For more complex tasks involving dynamic behaviors or collision avoidance, more expressive or learned controllers may be needed. Integrating such capabilities with the RL locomotion policy presents an interesting challenge for future work.

Finally, our experiments in this paper focus on repurposing legs for manipulation in an arm-mounted quadruped. Extending ReLIC to broader forms of interlimb coordination, such as using arms to support locomotion or generalizing to other robot morphologies, offers exciting opportunities for expanding whole-body autonomy.

References

[1] B. U. Rehman, M. Focchi, J. Lee, H. Dallali, D. G. Caldwell, and C. Semini. Towards a multi-legged mobile manipulator. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 3618–3624, 2016. doi:10.1109/ICRA.2016.7487545.

[2] J.-P. Sleiman, F. Farshidian, and M. Hutter. Versatile multicontact planning and control for legged loco-manipulation. *Science Robotics*, 8(81):eadg5014, 2023.

[3] Z. Gu, J. Li, W. Shen, W. Yu, Z. Xie, S. McCrory, X. Cheng, A. Shamsah, R. Griffin, C. K. Liu, A. Kheddar, X. B. Peng, Y. Zhu, G. Shi, Q. Nguyen, G. Cheng, H. Gao, and Y. Zhao. Humanoid locomotion and manipulation: Current progress and challenges in control, planning, and learning, 2025. URL https://arxiv.org/abs/2501.02116.

[4] S. Zimmermann, R. Poranne, and S. Coros. Go fetch!-dynamic grasps using boston dynamics spot with external robotic arm. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4488–4494. IEEE, 2021.

[5] J. Hooks, M. S. Ahn, J. Yu, X. Zhang, T. Zhu, H. Chae, and D. Hong. Alphred: A multi-modal operations quadruped robot for package delivery applications. *IEEE Robotics and Automation Letters*, 5(4):5409–5416, 2020. doi:10.1109/LRA.2020.3007482.

[6] C. D. Bellicoso, K. Krämer, M. Stäuble, D. Sako, F. Jenelten, M. Bjelonic, and M. Hutter. Alma-articulated locomotion and manipulation for a torque-controllable robot. In *2019 International conference on robotics and automation (ICRA)*, pages 8477–8483. IEEE, 2019.

[7] Z. Fu, X. Cheng, and D. Pathak. Deep whole-body control: learning a unified policy for manipulation and locomotion. In *Conference on Robot Learning*, pages 138–149. PMLR, 2023.

[8] Y. Zhang, T. Liang, Z. Chen, Y. Ze, and H. Xu. Catch it! learning to catch in flight with mobile dexterous hands. *arXiv preprint arXiv:2409.10319*, 2024.

[9] M. Bajracharya, J. Borders, R. Cheng, D. Helmick, L. Kaul, D. Kruse, J. Leichty, J. Ma, C. Matl, F. Michel, et al. Demonstrating mobile manipulation in the wild: A metrics-driven approach. *arXiv preprint arXiv:2401.01474*, 2024.

[10] Z. Fu, T. Z. Zhao, and C. Finn. Mobile aloha: Learning bimanual mobile manipulation using low-cost whole-body teleoperation. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 4066–4083. PMLR, 06–09 Nov 2025.

[11] J. Hu, P. Stone, and R. Martín-Martín. Causal policy gradient for whole-body mobile manipulation. In *Proceedings of Robotics: Science and Systems*, 2022.

[12] J. Kindle, F. Furrer, T. Novkovic, J. J. Chung, R. Siegwart, and J. Nieto. Whole-body control of a mobile manipulator using end-to-end reinforcement learning. *arXiv preprint arXiv:2003.02637*, 2020.

[13] T. Lew, S. Singh, M. Prats, J. Bingham, J. Weisz, B. Holson, X. Zhang, V. Sindhwani, Y. Lu, F. Xia, et al. Robotic table wiping via reinforcement learning and whole-body trajectory optimization. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7184–7190. IEEE, 2023.

[14] C. Sun, J. Orbik, C. M. Devin, B. H. Yang, A. Gupta, G. Berseth, and S. Levine. Fully autonomous real-world reinforcement learning with applications to mobile manipulation. In *Conference on Robot Learning*, pages 308–319. PMLR, 2022.

[15] R. Yang, Y. Kim, R. Hendrix, A. Kembhavi, X. Wang, and K. Ehsani. Harmonic mobile manipulation. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3658–3665. IEEE, 2024.

[16] J. Wu, R. Antonova, A. Kan, M. Lepert, A. Zeng, S. Song, J. Bohg, S. Rusinkiewicz, and T. Funkhouser. Tidybot: Personalized robot assistance with large language models. *Autonomous Robots*, 47(8):1087–1102, 2023.

[17] T. He, Z. Luo, X. He, W. Xiao, C. Zhang, W. Zhang, K. M. Kitani, C. Liu, and G. Shi. Omnih2o: Universal and dexterous human-to-humanoid whole-body teleoperation and learning. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 1516–1540. PMLR, 06–09 Nov 2025.

[18] Z. Fu, Q. Zhao, Q. Wu, G. Wetzstein, and C. Finn. Humanplus: Humanoid shadowing and imitation from humans. In *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pages 2828–2844. PMLR, 06–09 Nov 2025.

[19] Y. Ze, Z. Chen, W. Wang, T. Chen, X. He, Y. Yuan, X. B. Peng, and J. Wu. Generalizable humanoid manipulation with improved 3d diffusion policies. *arXiv preprint arXiv:2410.10803*, 2024.

[20] C. Lu, X. Cheng, J. Li, S. Yang, M. Ji, C. Yuan, G. Yang, S. Yi, and X. Wang. Mobile-television: Predictive motion priors for humanoid whole-body control. *arXiv preprint arXiv:2412.07773*, 2024.

[21] Y. Ma, F. Farshidian, T. Miki, J. Lee, and M. Hutter. Combining learning-based locomotion policy with model-based manipulation for legged mobile manipulators. *IEEE Robotics and Automation Letters*, 7(2):2377–2384, 2022.

[22] P. Ewen, J.-P. Sleiman, Y. Chen, W.-C. Lu, M. Hutter, and R. Vasudevan. Generating continuous motion and force plans in real-time for legged mobile manipulation. In *2021 IEEE international conference on robotics and automation (ICRA)*, pages 4933–4939. IEEE, 2021.

[23] N. Yokoyama, A. Clegg, J. Truong, E. Undersander, T.-Y. Yang, S. Arnaud, S. Ha, D. Batra, and A. Rai. Asc: Adaptive skill coordination for robotic mobile manipulation. *IEEE Robotics and Automation Letters*, 9(1):779–786, 2023.

[24] H. Ferrolho, V. Ivan, W. Merkt, I. Havoutis, and S. Vijayakumar. Roloma: Robust loco-manipulation for quadruped robots with arms. *Autonomous Robots*, 47(8):1463–1481, 2023.

[25] E. Arcari, M. V. Minniti, A. Scampicchio, A. Carron, F. Farshidian, M. Hutter, and M. N. Zeilinger. Bayesian multi-task learning mpc for robotic mobile manipulation. *IEEE Robotics and Automation Letters*, 8(6):3222–3229, 2023.

[26] J. Zhang, N. Gireesh, J. Wang, X. Fang, C. Xu, W. Chen, L. Dai, and H. Wang. Gamma: Graspability-aware mobile manipulation policy learning based on online grasping pose fusion. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1399–1405. IEEE, 2024.

[27] S. Jeon, M. Jung, S. Choi, B. Kim, and J. Hwangbo. Learning whole-body manipulation for quadrupedal robot. *IEEE Robotics and Automation Letters*, 9(1):699–706, 2023.

[28] H. Ha, Y. Gao, Z. Fu, J. Tan, and S. Song. UMI on legs: Making manipulation policies mobile with manipulation-centric whole-body controllers. In *Proceedings of the 2024 Conference on Robot Learning*, 2024.

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

[30] J.-P. Sleiman, M. Mittal, and M. Hutter. Guided reinforcement learning for robust multi-contact loco-manipulation. In *8th Annual Conference on Robot Learning (CoRL 2024)*, 2024.

[31] G. Pan, Q. Ben, Z. Yuan, G. Jiang, Y. Ji, J. Pang, H. Liu, and H. Xu. Roboduet: A framework affording mobile-manipulation and crossembodiment. *arXiv preprint arXiv:2403.17367*, 6, 2024.

[32] A. Fuchs and J. Kelso. A theoretical note on models of interlimb coordination. *Journal of Experimental Psychology: Human Perception and Performance*, 20(5):1088, 1994.

[33] T. Wannier, C. Bastiaanse, G. Colombo, and V. Dietz. Arm to leg coordination in humans during walking, creeping and swimming activities. *Experimental brain research*, 141:375–379, 2001.

[34] S. F. Donker, P. J. Beek, R. Wagenaar, and T. Mulder. Coordination between arm and leg movements during locomotion. *Journal of motor behavior*, 33(1):86–102, 2001.

[35] S. Aoi, P. Manoonpong, Y. Ambe, F. Matsuno, and F. Wörgötter. Adaptive control strategies for interlimb coordination in legged robots: a review. *Frontiers in neurorobotics*, 11:39, 2017.

[36] T. Tarn, A. Bejczy, and X. Yun. Coordinated control of two robot arms. In *Proceedings. 1986 IEEE International Conference on Robotics and Automation*, volume 3, pages 1193–1202. IEEE, 1986.

[37] T. Lin, Z.-H. Yin, H. Qi, P. Abbeel, and J. Malik. Twisting lids off with two hands. In *8th Annual Conference on Robot Learning*, 2024.

[38] W. J. Wolfslag, C. McGreavy, G. Xin, C. Tiseo, S. Vijayakumar, and Z. Li. Optimisation of body-ground contact for augmenting the whole-body loco-manipulation of quadruped robots. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 3694–3701. IEEE, 2020.

[39] C. Lin, X. Liu, Y. Yang, Y. Niu, W. Yu, T. Zhang, J. Tan, B. Boots, and D. Zhao. Locoman: Advancing versatile quadrupedal dexterity with lightweight loco-manipulators. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 6877–6884. IEEE, 2024.

[40] P. Arm, M. Mittal, H. Kolvenbach, and M. Hutter. Pedipulate: Enabling manipulation skills using a quadruped robot's leg. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5717–5723. IEEE, 2024.

[41] Z. He, K. Lei, Y. Ze, K. Sreenath, Z. Li, and H. Xu. Learning visual quadrupedal loco-manipulation from demonstrations. In *2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9102–9109. IEEE, 2024.

[42] X. Cheng, A. Kumar, and D. Pathak. Legs as manipulator: Pushing quadrupedal agility beyond locomotion. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5106–5112. IEEE, 2023.

[43] K. Zakka. Mink: Python inverse kinematics based on mujoco, July 2024. URL `https://github.com/kevinzakka/mink`. License: Apache-2.0.

[44] M. Mittal, C. Yu, Q. Yu, J. Liu, N. Rudin, D. Hoeller, J. L. Yuan, R. Singh, Y. Guo, H. Mazhar, A. Mandlekar, B. Babich, G. State, M. Hutter, and A. Garg. Orbit: A unified simulation framework for interactive robot learning environments. *IEEE Robotics and Automation Letters*, 8(6):3740–3747, 2023. doi:10.1109/LRA.2023.3270034.

[45] A. Miller, F. Yu, M. Brauckmann, and F. Farshidian. High-performance reinforcement learning on spot: Optimizing simulation parameters with distributional measures. *arXiv preprint arXiv:2403.16967*, 2025.

[46] J. Siekmann, Y. Godse, A. Fern, and J. Hurst. Sim-to-real learning of all common bipedal gaits via periodic reward composition. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7309–7315, 2021. doi:10.1109/ICRA48506.2021.9561814.

[47] S. Miller, M. Fritz, T. Darrell, and P. Abbeel. Parametrized shape models for clothing. In *2011 IEEE International Conference on Robotics and Automation*, pages 4861–4868. IEEE, 2011.

[48] M. Nomura and M. Shibata. cmaes : A simple yet practical python library for cma-es. *arXiv preprint arXiv:2402.01373*, 2024.

[49] OpenAI. Gpt-4o system card, 2024. URL `https://arxiv.org/abs/2410.21276`.

[50] Boston Dynamics. Spot. `https://bostondynamics.com/products/spot/`, .

[51] Boston Dynamics. Spot sdk. `https://dev.bostondynamics.com/`, .

[52] O. R. developers. Onnx runtime. `https://onnxruntime.ai/`, 2021.

[53] N. Rudin, D. Hoeller, P. Reist, and M. Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on Robot Learning*, pages 91–100. PMLR, 2022.

[54] N. Ravi, V. Gabeur, Y.-T. Hu, R. Hu, C. Ryali, T. Ma, H. Khedr, R. Rädle, C. Rolland, L. Gustafson, E. Mintun, J. Pan, K. V. Alwala, N. Carion, C.-Y. Wu, R. Girshick, P. Dollár, and C. Feichtenhofer. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024. URL `https://arxiv.org/abs/2408.00714`.