Pedipulate: Enabling Manipulation Skills using a Quadruped Robot's Leg

Conference Paper

Author(s):
Arm, Philip; Mittal, Mayank (ID); Kolvenbach, Hendrik (ID); Hutter, Marco (ID)

Pedipulate: Enabling Manipulation Skills using a Quadruped Robot's Leg

Philip Arm, Mayank Mittal, Hendrik Kolvenbach, and Marco Hutter

Abstract— **Legged robots have the potential to become vital in maintenance, home support, and exploration scenarios. In order to interact with and manipulate their environments, most legged robots are equipped with a dedicated robot arm, which means additional mass and mechanical complexity compared to standard legged robots. In this work, we explore pedipulation - using the legs of a legged robot for manipulation. By training a reinforcement learning policy that tracks position targets for one foot, we enable a dedicated pedipulation controller that is robust to disturbances, has a large workspace through whole-body behaviors, and can reach far-away targets with gait emergence, enabling loco-pedipulation. By deploying our controller on a quadrupedal robot using teleoperation, we demonstrate various real-world tasks such as door opening, sample collection, and pushing obstacles. We demonstrate load carrying of more than $2.0\,\mathrm{kg}$ at the foot. Additionally, the controller is robust to interaction forces at the foot, disturbances at the base, and slippery contact surfaces. Videos of the experiments are available at https://sites.google.com/leggedrobotics.com/pedipulate.**

I. INTRODUCTION

In recent years, the locomotion capabilities of legged robots have greatly improved. Specifically, quadrupedal robots are nowadays capable of traversing various industrial and natural environments [1]–[4], facilitating their deployment in inspection, exploration, and search-and-rescue missions [5]–[10]. Although legged robots have reached industrial maturity, their application range currently remains limited to inspection tasks that require minimal interaction with the environment. We need to improve and robustify their manipulation capabilities to deploy legged robots in a broader range of real-world scenarios, such as maintenance, home support, and sample collection.

Recent works in legged mobile manipulation have mainly focused on systems with a dedicated robotic arm for interactive tasks such as door opening, fetching objects, and opening valves [11]–[15]. However, such an additional arm increases the robot's mechanical complexity and power consumption. Taking inspiration from quadrupedal animals [16], we hypothesize that many manipulation tasks do not require the additional dexterity and complexity of robotic arms and hands but can be solved using the legs of quadrupedal robots. Using the same limbs for locomotion and manipulation can reduce

Fig. 1: Our foot target tracking controller enables a variety of real-world manipulation tasks such as opening doors (A) and fridges (B), object transport (C), pressing a button (D), pushing obstacles out of the way (E), and collecting rock samples (F).

a robotic system's mechanical complexity and cost, which is especially beneficial in mass-constrained applications such as space exploration.

We, therefore, seek to enable a broad range of manipulation tasks using the legs of quadrupedal robots - so-called *pedipulation*. By developing a versatile low-level controller that can track foot target points, we investigate how we can use pedipulation to enable real-world skills. To this end, we use a deep reinforcement learning setup to train a robust controller for foot target point tracking. We additionally seek to enable the controller to approach far-away targets with the foot in the air while using a tripod gait, facilitating object transport. We investigate the controller's tracking performance and disturbance rejection capabilities in simulation and hardware experiments and deploy it in various real-world tasks through teleoperation. Specifically, our key contributions are as follows:

- We design a reinforcement learning-based pedipulation controller by training a policy that tracks foot position targets.
- We investigate the achievable workspace of our controller while standing and show that it reaches a large local workspace through inherent whole-body motions.
- We enable the controller to adapt the robot's stance and locomote towards far-away targets via curriculum-based command sampling.
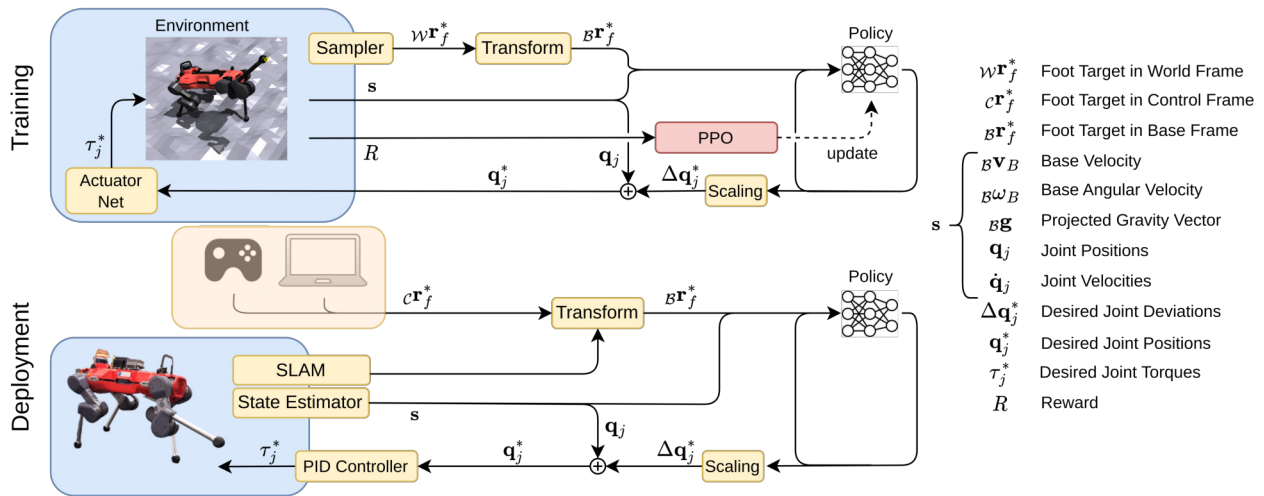
Fig. 2: Overview of our training and deployment setup. We first specify the commands in an inertial frame, namely the world frame in training and the control frame during deployment. To make the policy agnostic to the used inertial frame, we transform the commands to the base frame before adding them to the observations. The policy's actions are interpreted as deviations from the current joint position.

- We demonstrate that the controller is robust to external disturbances, such as interaction forces on the base and foot, and slippery terrains.
- We show that through the above features, our learned pedipulation controller enables numerous real-world manipulation skills such as door opening, rock sample collection, and pushing obstacles without task-specific adaptions (Fig. 1).

II. RELATED WORK

Several works investigated manipulation using robotic legs - or, more precisely, pedipulation - in recent years. We split these works into three categories:

- *Manipulation via locomotion*: The robot moves objects towards a goal by pushing them with its body while walking.
- *Non-prehensile pedipulation*: The robot manipulates objects with its legs without the capability to exert pulling forces onto the objects.
- *Prehensile pedipulation*: The robot has a specialized tool on one or multiple legs to grasp and manipulate objects with pushing and pulling forces.

A. Manipulation via Locomotion

Manipulation via locomotion can enable robots to push objects close to their body weight [17]. Scaling this strategy to a multi-robot approach allows manipulating even larger objects [18], [19]. In these works, the authors trained a reinforcement learning policy to move large-scale objects to a desired 2D pose. This approach can, for example, be used to remove obstacles. However, it does not generalize to dexterous tasks such as opening doors or pushing small buttons, which we aim to tackle in this paper.

B. Non-prehensile Pedipulation

Non-prehensile pedipulation facilitates moving obstacles out of the way [20], kicking or balancing balls [21], [22],

probing the environment [6], [23], or opening doors [24]. Some of the above approaches track pre-defined trajectories with a tracking controller. Lu et al. [20] used an inverse kinematics controller to track a heuristically defined Bézier Curve to push objects on the ground. Kolvenbach et al. [6], [23] executed a pre-defined trajectory with FreeGait, an API for whole-body control [25] to haptically inspect the terrain. Others trained a reinforcement learning policy on a single task, for example, kicking balls [21] or balancing balls [22]. While many of these works show promising results in their respective domain, our goal is to have a general controller that can be used for multiple manipulation tasks.

Cheng et al. [26] showed a non-prehensile pedipulation controller capable of generalizing to multiple tasks. In this work, the authors used separately learned low-level manipulation and locomotion policies and a high-level behavior tree to synthesize skills such as kicking a ball or pushing buttons. While the paper shows an impressive range of skills, the low-level manipulation and locomotion policies are separated, which implies that combined locomotion and pedipulation maneuvers (loco-pedipulation) are not feasible. Furthermore, the manipulation policy only tracks half-sine trajectories near the robot's nominal foot position. This form of command sampling probably does not generalize to arbitrary trajectories in the robot's workspace. Both these limitations prevent the application of the approach in more complex scenarios, as considered in this work, like door opening or object transport.

C. Prehensile Pedipulation

Integrating a gripper on the robot's foot enables applications such as fetching objects or collecting samples [27]–[30]. Both the Lauron robot [27], [28] and Mantis [29] are hexapod robots where the integrated grippers can fetch objects. These works show capable gripper concepts. However, they provide little detail on their manipulation control

approach and do not demonstrate robustness to external disturbances. Tsvetkov and Ramamoorthy [30] introduced a dactylus-style gripper for single-leg and two-leg manipulation. In this work, the gripper concept is remarkable and facilitates manipulating multiple objects and tools. However, the operator can only command the robot's target position in joint space. The target position is converted into a trajectory and tracked in an open-loop fashion. This approach requires cumbersome command tuning and limits the robustness to disturbances. In our work, we aim to overcome these issues and develop a versatile controller which is useful for both prehensile and non-prehensile pedipulation scenarios.

III. METHOD

We hypothesize that a hierarchical structure with a task-specific high-level planner and a generic low-level controller generalizes well to numerous pedipulation tasks. Here, we focus on the required low-level controller that effectively tracks foot position commands and is robust to disturbances. Due to the recent success of reinforcement learning [1]–[4], [31], we choose to use deep reinforcement learning to train a neural network policy (Fig. 2). We train the policy to track foot target points of the right front foot with minimal additional rewards. During deployment, the user provides the command in an inertial frame, and we expect high-level planners to do the same. However, we want the policy to be agnostic to the used inertial frame. Therefore, we internally transform the command to the robot's base frame before adding it to the observation vector.

We use Isaac Gym as a simulation environment [32], [33] and PPO as a deep reinforcement learning algorithm [34]. We use MLPs for the actor and critic networks as in previous work on locomotion [32] and rely on similar hyperparameters. We randomize simulation friction parameters, add random pushes to the robot's base in simulation, and train on irregular terrain to facilitate the sim-to-real transfer. We deployed our controller on the quadrupedal robot ANYmal D by ANYbotics [35].

A. Observation and Action Space

The policy receives proprioceptive information about the robot's state and the foot position commands as observations (Table I). The scaled actions are interpreted as desired deviations from the current joint positions. We tested two action spaces: deviations from current joint positions and nominal joint positions. However, we did not notice a relevant difference in the achieved rewards and behaviors.

B. Rewards

We use two types of rewards; a task reward R_e representing the main objective, and penalties R_n, which render the behavior safe and physically feasible on the real system. The total reward is $R = R_e + R_n$.

1) Task Reward: We provide a single dense task reward to encourage the foot to be close to the target point. The reward function is defined as

$$R_e = w_1 \cdot e^{-\left(\frac{|| _{\mathcal{B}}\boldsymbol{r}_f - _{\mathcal{B}}\boldsymbol{r}_f^*||}{\sigma}\right)}, \tag{1}$$
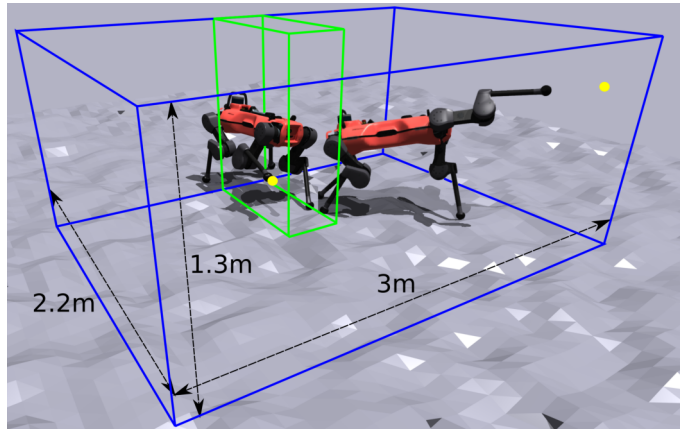

Fig. 3: Simulation setup in Isaac Gym. The initial and final sampling space of the foot position commands are visualized in green and blue, respectively. The command curriculum enables combined pedipulation and tripod locomotion in a single policy. The figure shows one robot tracking a close target point (left) and a robot approaching a far-range target point using the tripod gait (right). The target points are visualized in yellow.

Observations		
base linear velocity	$_{\mathcal{B}}\boldsymbol{v}_B$	$\mathbb{R}^3$
base angular velocity	$_{\mathcal{B}}\boldsymbol{\omega}_B$	$\mathbb{R}^3$
projected gravity vector	$_{\mathcal{B}}\boldsymbol{g}$	$\mathbb{R}^3$
joint positions	$\boldsymbol{q}_j$	$\mathbb{R}^{12}$
joint velocities	$\dot{\boldsymbol{q}}_j$	$\mathbb{R}^{12}$
foot position command	$_{\mathcal{B}}\boldsymbol{r}_f^*$	$\mathbb{R}^3$
last actions	$\boldsymbol{a}_{t-1}$	$\mathbb{R}^{12}$
Actions		
desired deviation from last joint positions	$\Delta\boldsymbol{q}_j^*$	$\mathbb{R}^{12}$

TABLE I: Observations and actions we use for our foot target tracking policy. The actor and critic receive the same set of observations.

where $_{\mathcal{B}}\boldsymbol{r}_f$ and $_{\mathcal{B}}\boldsymbol{r}_f^*$ denote the measured and desired foot position in the base frame, respectively. We use $w_1 = 15$ and $\sigma = 0.8\,\mathrm{m}^{-1}$.

2) Normalization Rewards: To reduce jerky motions and enable safe behaviors on the real robot, we add penalties on quantities such as joint torques, joint velocities, and collisions between robot links. The normalization reward is defined as

$$R_n = w_2||\dot{\boldsymbol{q}}_j||^2 + w_3||\ddot{\boldsymbol{q}}_j||^2 + w_4||\boldsymbol{\tau}||^2 \\ + w_5||\boldsymbol{a}_t - \boldsymbol{a}_{t-1}||^2 + w_6 \cdot n_c + w_7 \cdot k, \tag{2}$$

where $\dot{\boldsymbol{q}}_j$ and $\ddot{\boldsymbol{q}}_j$ denote the joint velocities and accelerations, $\boldsymbol{\tau}$ denotes the joint torques, $\boldsymbol{a}_t$ the action at time t and w_i the scaling factors. Additionally, we penalize collisions between shanks and thighs, expressed by the number of collisions at the current time step n_c. Lastly, we add a termination penalty, which is applied to the environments where a base collision is registered, denoted by $k \in \{0, 1\}$, where $k = 1$ represents a termination at the current time step. We use the manually tuned reward scales

$$w_2 = -5 \cdot 10^{-2} \qquad w_3 = -5 \cdot 10^{-6}$$
$$w_4 = -2 \cdot 10^{-5} \qquad w_5 = -1 \cdot 10^{-2}$$
$$w_6 = -2 \qquad w_7 = -80.$$

Fig. 4: Our controller enables numerous real-world manipulation tasks: (A) The robot opens a push door. (B) The robot opens a fridge. (C) The robot lifts a backpack and transports it to a box using a tripod-hopping gait. (D) The large workspace of the controller allows pressing a button far above the robot's base. (E) The controller can be used to push obstacles out of the way. (F) With an additional gripper, the robot can collect rock samples.

C. Terrain

We observed that training on flat terrain leads to behaviors where stance feet slide across the ground to adapt the stance. On the real robot, such policies are prone to stumbling on terrain irregularities. To robustify the behavior, we train on irregular terrain (Fig. 3). We uniformly sample a height field between $-0.08\,\mathrm{m}$ and $0.08\,\mathrm{m}$ on a grid with $0.2\,\mathrm{m}$ resolution to create the terrain.

D. Command Curriculum for Loco-Pedipulation

To enable tracking far-away targets that require the robot to adapt the stance legs or walk on three legs, we implemented a curriculum for adaptive command sampling. Initially, commands are sampled uniformly in a box that is reachable for the foot without moving the stance feet (green box in Fig. 3). Once the agent learns to track the commands with an average error of less than $0.06\,\mathrm{m}$, the command sampling box is extended by $0.2\,\mathrm{m}$ in the robot's heading direction and $0.2\,\mathrm{m}$ in both lateral directions. We limit the box to $2.2\,\mathrm{m}$ x $3.0\,\mathrm{m}$ x $1.3\,\mathrm{m}$ (blue box in Fig. 3) and let the training converge with this command range. The command range does not limit the range of loco-pedipulation maneuvers since we can still give a sequence of target points to reach a further target.

E. Sim-to-Real Transfer, Training, and Deployment

We randomize the friction parameters in the simulation and add noise to the observations. Additionally, we exert random pushes onto the robot's base every $3.0\,\mathrm{s}$, which we simulate by setting a random twist to the base link. Each twist component is sampled in $[-0.6, 0.6]\,\mathrm{m\,s^{-1}}$ or $[-0.6, 0.6]\,\mathrm{rad\,s^{-1}}$. Furthermore, we add a disturbance force at the feet of up to $12\,\mathrm{N}$ that is constant over an episode. Finally, we use an actuator network to simulate the actuator dynamics as in [36]. We train with an episode length of $12\,\mathrm{s}$ and resample commands every $6\,\mathrm{s}$. Training takes $5.5\,\mathrm{h}$ on an NVIDIA GeForce RTX 2080 Ti for 8000 iterations.

During deployment, we define the command in a fixed local control frame ($_C r_F^*$). When our pedipulation controller starts, it locks the control frame at the current base position, and our SLAM solution [37] continuously updates the transforms between the map, the control, and the base frame. We define the command by the joystick or the GUI in the control frame and transform it into the base frame ($_B r_F^*$) before adding it to the policy observations. The control frame allows the user to intuitively control the foot in a fixed frame rather than having the behavior influenced by the current base pose. The user can use the joystick to incrementally move the target point or a GUI interface to directly command a single far-away target point (Fig. 2).The policy and the PID controller run at $50\,\mathrm{Hz}$ and $400\,\mathrm{Hz}$, respectively.

IV. RESULTS AND DISCUSSION

A. Real-World Manipulation Skills

Tracking foot target points in a large workspace via whole-body motions allows for solving various manipulation tasks (Fig. 4). This section presents multiple manipulation skills we achieved on the real system using our designed pedipulation controller and teleoperation interface. The operator had a direct line of sight to the robot in all experiments.

1) Door Opening: Door opening requires a controller that is robust to interaction forces with the environment. Additionally, reaching the door handle with the robot's foot requires a large workspace. Lastly, passing the door when it is open requires a combined loco-pedipulation behavior.
The robot was able to open a push door with a resistive spring-damper element at the hinge (Fig. 4A). We used the joystick interface to push down the handle and then set a target point behind the door using the GUI to move the robot through the door using the tripod gait while simultaneously pushing the door open with the pedipulating foot.
Similarly, we used our controller to open a fridge door (Fig. 4B). The friction on the foot was sufficient to open the pull door on the fridge. The interaction force at the door did not destabilize the controller.

2) Transporting a Backpack: In this experiment, we collected a $2.0\,\mathrm{kg}$ backpack by sliding the foot into one of the shoulder straps and raising the foot using the joystick interface (Fig. 4C). We then commanded a single target point above a box using the GUI interface and lowered the foot to release the backpack into the box with the joystick. This experiment demonstrates the ability of our controller to transport loads with the foot in the air thanks to the combined loco-pedipulation capability.

3) Pressing a Hard-to-Reach Button: The robot can exploit the large workspace of our pedipulation controller to reach and press a button of $0.03\,\mathrm{m}$ x $0.05\,\mathrm{m}$ at a height of $1.25\,\mathrm{m}$ (Fig. 4D). The operator can move the target point into the wall such that the foot exerts a small force on the button without destabilizing the controller.

4) Moving an Obstacle: A common problem in legged locomotion is moving obstacles out of the way in confined spaces (Fig. 4C). Our pedipulation controller allows the user to move obstacles out of the robot's way to continue walking
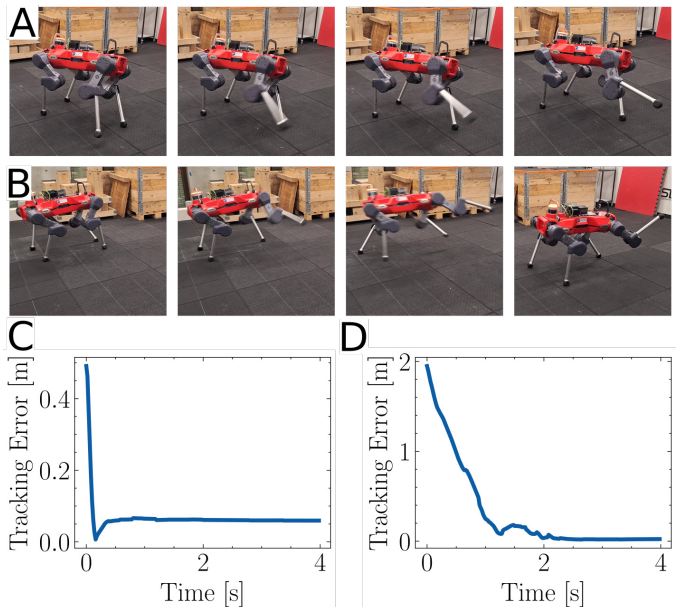

Fig. 5: For close-range targets, the policy does not require a stance adaptation (A) and the tracking error quickly converges to a steady state value (C). For far-range targets, a tripod gait emerges (B), and the tracking error oscillates in the slow-down maneuver before converging (D).

through a confined space. Again, robustness to unmodelled forces is required. In this experiment, we used a box with a mass of $2.6\,\mathrm{kg}$.

5) Rock Sample Collection: By adding a gripper to the foot, the robot can conduct prehensile pedipulation tasks like collecting rock samples (Fig. 4D). This task is especially relevant in scenarios like planetary exploration, which require capable robots with minimal mass and mechanical complexity. We successfully collected rock samples of a size of $0.08\,\mathrm{m}$ and a mass of roughly $300\,\mathrm{g}$. The controller proved robust to the additional mass of the gripper and the rock.

B. Reachable Workspace and Tracking Performance

We investigated the reachable workspace of our pedipulation controller and benchmarked the tracking performance. In these experiments, we distinguish between two behaviors: Close-range tracking and far-range tracking. Close-range tracking refers to target points the robot can reach without adapting its stance or walking to reach it. Far-range tracking refers to target points the policy reaches via locomotion or stance adaptation. Fig. 5 shows the respective behaviors. Notably, the transition between the two is seamless. If the target point moves outside the reachable range, the robot automatically starts stepping to follow the target point.

1) Close-Range Target Tracking: To evaluate the workspace our controller covers without locomotion, we deployed it in Isaac Gym and uniformly sampled commands in the robot's vicinity. After each episode, we discarded targets if the robot broke contact with one of the designated stance legs or if the tracking error was higher than $0.05\,\mathrm{m}$. As shown in Fig. 6, the whole-body behavior of the controller allows the robot to reach a large workspace, even without adapting its stance. Note that this workspace does

not coincide with the theoretical limits reachable through whole-body motions in stance. The robot stretches almost to the kinematic limits but starts stepping early to reach a more favorable configuration. These samples are discarded in this evaluation.

Additionally, we evaluated the tracking performance in the close-range workspace by sampling points within a box of $0.3\,\text{m}$ x $0.4\,\text{m}$ x $0.2\,\text{m}$ both in simulation and on the real robot. We compared the foot target position with the ground truth position in the simulation and the estimated position from the state estimator on the real robot. In simulation, the controller reached an average error of $0.037\,\text{m}$, while on the real robot, the average error was $0.057\,\text{m}$. We use a fixed-base inverse kinematics controller as a baseline, which reached an average error of $0.041\,\text{m}$ in simulation on the reachable range.

2) Far-Range Target Tracking: To evaluate the tracking performance of the controller on target points the policy reaches by stepping or walking, we evaluated the tracking error on the full training range ($2.2\,\text{m}$ x $2.7\,\text{m}$ x $1.3\,\text{m}$) in Isaac Gym. We uniformly sampled commands on this range and logged the tracking error at the end of each episode. Our controller reaches an average tracking error of $0.043\,\text{m}$.

C. Robustness against Disturbances

To evaluate the controller's ability to generalize to different environmental conditions, we deployed it on ANYmal D standing on a wet whiteboard with extremely low friction. Although the stance legs slip during operation, the robot does not fall. Even when applying disturbance forces at the foot, which causes the stance legs to slip, the robot remains standing (Fig. 7A).

Additionally, we disturbed the robot's base while the controller was tracking a constant target point while standing on a flat floor. Thanks to the whole-body behavior of the controller, it can compensate for disturbances on the base. We applied disturbances of up to $132\,\text{N}$. We estimated the force on the base by summing the contact forces on the stance legs we obtained from the state estimator via the joint torque measurements. The disturbance force resulted in a base displacement of up to $0.1\,\text{m}$ along the heading axis. Despite this disturbance, the tracking error only increased by $0.06\,\text{m}$ (Fig. 7C). Interestingly, the controller can compensate for disturbances in the negative heading direction better than for those in the positive heading direction.

If we disturb the base further, interesting behaviors emerge: If required, the stance feet adapt, and the robot steps to remain stable. The accompanying video shows this behavior.

V. CONCLUSION AND FUTURE WORK

In this work, we developed a deep reinforcement learning policy to track end-effector target points with a quadrupedal robot's foot. We showed that such a controller enables solving numerous real-world tasks via pedipulation: Using our controller, we could teleoperate the robot to successfully open doors, transport objects, press buttons, push obstacles, and collect rock samples. These skills are enabled by
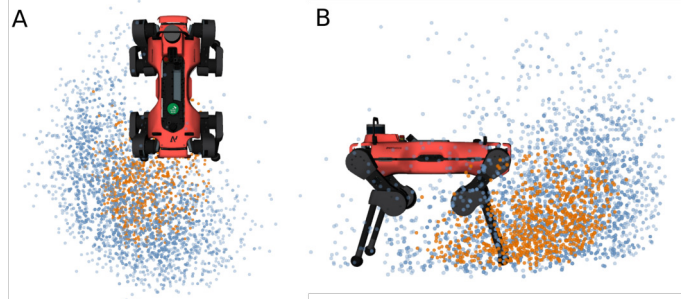

Fig. 6: Thanks to the whole-body behavior, the robot reaches a large workspace even without the need to adapt its stance. The blue dots represent sampled commands the controller tracks without changing the stance configuration and with a tracking error below $0.05\,\text{m}$, shown in top view (A) and side view (B). For reference, the orange dots are reachable samples for a fixed-base inverse kinematics controller with the same sampling space.
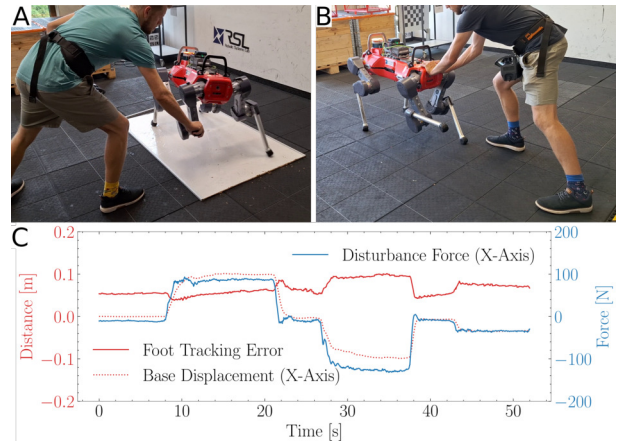

Fig. 7: (A): The controller can react to disturbances, even on slippery surfaces like whiteboards. (B): When the robot's base is disturbed, the controller compensates for the base motion. (C): Evolution of the tracking error over time when the base is disturbed.

the controller's large workspace, achieved through inherent whole-body behaviors, its loco-pedipulation capability, and its robustness to disturbances.

While the controller is helpful in teleoperation scenarios, future work should move towards autonomous pedipulation. For such autonomy, a hierarchical approach, successfully used for other tasks [17], could be a valid strategy. A hierarchical controller could contain our pedipulation policy as a low-level policy and combine it with task-specific perceptive high-level policies that output the foot targets. Additionally, we did not explicitly model interaction forces. Tracking interaction forces could extend the range of admissible manipulation tasks, such as haptic interaction or heavy payload transportation.

Our work shows that numerous manipulation tasks can be solved by only doing pedipulation with quadrupedal robots. This insight will be relevant for future works on the design and control of legged mobile manipulators.

REFERENCES

[1] J. Lee, *et al.*, "Learning quadrupedal locomotion over challenging terrain," *Science Robotics*, vol. 5, no. 47, p. eabc5986, 2020.

[2] T. Miki, *et al.*, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Science Robotics*, vol. 7, no. 62, p. eabk2822, 2022.

[3] S. Choi, *et al.*, "Learning quadrupedal locomotion on deformable terrain," *Science Robotics*, vol. 8, no. 74, p. eade2256, 2023.

[4] D. Hoeller, *et al.*, "Anymal parkour: Learning agile navigation for quadrupedal robots," *arXiv preprint arXiv:2306.14874*, 2023.

[5] M. Hutter, *et al.*, "Towards a generic solution for inspection of industrial sites," in *Field and Service Robotics: Results of the 11th International Conference*. Springer, 2018, pp. 575–589.

[6] H. Kolvenbach, *et al.*, "Towards autonomous inspection of concrete deterioration in sewers with legged robots," *Journal of Field Robotics*, vol. 37, no. 8, pp. 1314–1327, 2020.

[7] M. Stachowiak, *et al.*, "Procedures of detecting damage to a conveyor belt with use of an inspection legged robot for deep mine infrastructure," *Minerals*, vol. 11, no. 10, p. 1040, 2021.

[8] M. Tranzatto, *et al.*, "Cerberus in the darpa subterranean challenge," *Science Robotics*, vol. 7, no. 66, p. eabp9742, 2022.

[9] P. Arm, *et al.*, "Scientific exploration of challenging planetary analog environments with a team of legged robots," *Science robotics*, vol. 8, no. 80, p. eade9548, 2023.

[10] B. Lindqvist, *et al.*, "Multimodality robotic systems: Integrated combined legged-aerial mobility for subterranean search-and-rescue," *Robotics and Autonomous Systems*, vol. 154, p. 104134, 2022.

[11] C. D. Bellicoso, *et al.*, "Alma-articulated locomotion and manipulation for a torque-controllable robot," in *2019 International conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 8477–8483.

[12] S. Zimmermann, *et al.*, "Go fetch!-dynamic grasps using boston dynamics spot with external robotic arm," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 4488–4494.

[13] M. Mittal, *et al.*, "Articulated object interaction in unknown scenes with whole-body mobile manipulation," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 1647–1654.

[14] H. Ferrolho, *et al.*, "Roloma: Robust loco-manipulation for quadruped robots with arms," *arXiv preprint arXiv:2203.01446*, 2022.

[15] J.-P. Sleiman, *et al.*, "Versatile multicontact planning and control for legged loco-manipulation," *Science Robotics*, vol. 8, no. 81, p. eadg5014, 2023.

[16] A. Pets, "Dogs and cats opening doors," https://www.youtube.com/watch?v=MGGFNO3TVpg, online; accessed 06-September-2023.

[17] S. Jeon, *et al.*, "Learning whole-body manipulation for quadrupedal robot," *arXiv preprint arXiv:2308.16820*, 2023.

[18] O. Nachum, *et al.*, "Multi-agent manipulation via locomotion using hierarchical sim2real," *arXiv preprint arXiv:1908.05224*, 2019.

[19] M. J. Mataric, *et al.*, "Cooperative multi-robot box-pushing," in *Proceedings 1995 IEEE/RSJ International Conference on Intelligent Robots and Systems. Human Robot Interaction and Cooperative Robots*, vol. 3. IEEE, 1995, pp. 556–561.

[20] B. Lu, *et al.*, "Autonomous obstacle legipulation with a hexapod robot," *arXiv preprint arXiv:2011.06227*, 2020.

[21] Y. Ji, *et al.*, "Hierarchical reinforcement learning for precise soccer shooting skills using a quadrupedal robot," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 1479–1486.

[22] F. Shi, *et al.*, "Circus anymal: A quadruped learning dexterous manipulation with its limbs," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 2316–2323.

[23] H. Kolvenbach, *et al.*, "Haptic inspection of planetary soils with legged robots," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1626–1632, 2019.

[24] T. T. Topping, *et al.*, "Quasi-static and dynamic mismatch for door opening and stair climbing with a legged robot," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 1080–1087.

[25] P. Fankhauser, *et al.*, "Free gait—an architecture for the versatile control of legged robots," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 1052–1058.

[26] X. Cheng, *et al.*, "Legs as manipulator: Pushing quadrupedal agility beyond locomotion," *arXiv preprint arXiv:2303.11330*, 2023.

[27] A. Roennau, *et al.*, "Lauron v: A versatile six-legged walking robot with advanced maneuverability," in *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*. IEEE, 2014, pp. 82–87.

[28] G. Heppner, *et al.*, "Versatile-high power gripper for a six legged walking robot," in *Mobile Service Robotics*. World Scientific, 2014, pp. 461–468.

[29] W. Brinkmann, *et al.*, "Enhancement of the six-legged robot mantis for assembly and construction tasks in lunar mission scenarios within a multi-robot team," in *Proceedings: International Symposium on Artificial Intelligence, Robotics and Automation in Space*, 2020.

[30] Y. Tsvetkov and S. Ramamoorthy, "A novel design and evaluation of a dactylus-equipped quadruped robot for mobile manipulation," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 1633–1638.

[31] A. Agarwal, *et al.*, "Legged locomotion in challenging terrains using egocentric vision," in *Conference on Robot Learning*. PMLR, 2023, pp. 403–415.

[32] N. Rudin, *et al.*, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Conference on Robot Learning*. PMLR, 2022, pp. 91–100.

[33] V. Makoviychuk, *et al.*, "Isaac gym: High performance gpu-based physics simulation for robot learning," *arXiv preprint arXiv:2108.10470*, 2021.

[34] J. Schulman, *et al.*, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[35] ANYbotics, "Anymal specifications," https://www.anybotics.com/anymal-autonomous-legged-robot/, online; accessed 27-January-2023.

[36] J. Hwangbo, *et al.*, "Learning agile and dynamic motor skills for legged robots," *Science Robotics*, vol. 4, no. 26, p. eaau5872, 2019.

[37] S. Khattak, *et al.*, "Complementary multi–modal sensor fusion for resilient robot pose estimation in subterranean environments," in *2020 International Conference on Unmanned Aircraft Systems (ICUAS)*. IEEE, 2020, pp. 1024–1029.