

Rethinking Robustness Assessment: Adversarial Attacks on Learning-based Quadrupedal Locomotion Controllers

Fan Shi^{†1234}, Chong Zhang^{†1}, Takahiro Miki¹, Joonho Lee¹, Marco Hutter¹, Stelian Coros²

[†] Equal contribution. Video: <https://fanshi14.github.io/me/rss24.html>

¹ Robotic Systems Lab, ETH Zurich ² Computational Robotics Lab, ETH Zurich ³ ETH AI Center

⁴ National University of Singapore

Abstract—Legged locomotion has recently achieved remarkable success with the progress of machine learning techniques, especially deep reinforcement learning (RL). Controllers employing neural networks have demonstrated empirical and qualitative robustness against real-world uncertainties, including sensor noise and external perturbations. However, formally investigating the vulnerabilities of these locomotion controllers remains a challenge. This difficulty arises from the requirement to pinpoint vulnerabilities across a long-tailed distribution within a high-dimensional, temporally sequential space. As a first step towards quantitative verification, we propose a computational method that leverages sequential adversarial attacks to identify weaknesses in learned locomotion controllers. Our research demonstrates that, even state-of-the-art robust controllers can fail significantly under well-designed, low-magnitude adversarial sequence. Through experiments in simulation and on the real robot, we validate our approach’s effectiveness, and we illustrate how the results it generates can be used to robustify the original policy and offer valuable insights into the safety of these black-box policies.

I. INTRODUCTION

Legged robots, especially quadrupedal robots, are expected to free people from dull, dirty, and dangerous (3D) tasks, such as transportation, underground inspections, and rescue operations. However, real-world scenarios with complex terrain, sensor noises, and unexpected perturbations present significant hurdles. Robust locomotion controllers are essential in this context.

Recent advancements in machine learning, especially in reinforcement learning (RL), have significantly improved the performance of quadrupedal robots. [32, 67, 42]. Controllers based on neural networks (NN) have demonstrated remarkable robustness, contributing to the success of team CERBERUS in the DARPA Subterranean Challenge [62]. This competition is known for its demanding mobility tasks in real underground environments [12]. Unlike traditional controllers that rely on explicit dynamics models and online optimization [28, 20], neural network controllers reformulate the problem into offline optimization of the control policy through massive trials and errors in simulation.

Despite the rich literature on neural locomotion controllers, there’s still a limited understanding of how neural networks operate, making these controllers seem like a “black box” to researchers [37]. Recent studies have increasingly focused on



Fig. 1: The state-of-the-art robust locomotion policy [42] can be destabilized even on flat ground when subjected to a sequence of low-magnitude multi-modal adversarial attacks on observations, demonstrating the vulnerabilities in learning-based neural controllers.

the vulnerabilities of neural networks [15, 23], highlighting how small, adversarial modifications can significantly impact their performance. These subtle manipulations pose considerable risks in practical applications of neural network systems.

In the context of robot control, adversarial attacks can be formulated as introducing noises or disturbances that compromise the effectiveness of the controllers. For instance, Koren et al. [30] proposed to identify specific pedestrian movement patterns that could potentially cause an autonomous vehicle to crash [30]. Yet, within the domain of quadrupedal locomotion controllers, validation of such risks with temporally sequential attacks is still missing, highlighting the need of further exploration to understand and mitigate potential vulnerabilities.

Given the high-dimensional and multi-modal nature of the search space for sequential attacks, strategies such as introducing random noise through domain randomization [61] are insufficient for identifying the subtle vulnerabilities of the controllers. Moreover, policies which are designed to be resilient against such disturbances [42] present further chal-

lenges to discover effective sequential attacks using straightforward methods. Additionally, adversarial scenarios should be grounded in *realism*, implying they are scenarios that the robot might realistically encounter during its operational tasks. Highly improbable conditions, like disturbances significantly beyond normal environmental challenges, do not contribute effectively to our understanding of a controller’s nuanced vulnerabilities.

Our approach, therefore, is to generate adversaries in a way that is both methodical and informed by field robotics experience. This ensures that the scenarios we explore are not only plausible but also indicative of real-world conditions the robots might face. Moreover, the identification of the mildest successful adversarial sequence can also indicate the worst-case robustness, providing insights into the system’s safety. This comprehensive understanding will facilitate the future large-scale deployment of neural policies, ensuring they are equipped to handle a broad spectrum of operational challenges effectively. Furthermore, our findings reveal that the robustness of control policies can be notably increased by finetuning them with the adversarial scenarios we’ve identified, thereby enhancing their ability to withstand unexpected challenges.

Briefly, we identify our contribution as follows:

- 1) **A learning approach** that effectively discovers sequential adversarial attacks on neural quadrupedal locomotion controllers;
- 2) **Vulnerability analysis** of state-of-the-art quadrupedal locomotion controllers under multi-modal attacks and demonstrate on an actual robot;
- 3) **Closed-loop integration** with adversarial attacks to robustify and analyze the locomotion controllers, complemented by real-world validations.

II. RELATED WORK

A. Neural Controllers for Legged Robots

Neural controllers have shown impressive results on legged robots, achieving capabilities such as robust locomotion [26, 32, 67, 42, 7, 3, 41, 31, 36, 57, 56], mimicking animal behaviors [45, 46], interacting with objects [53, 59], and performing dynamic, athletic movements [21, 10]. The advantage of using neural policies lies in their ability to handle the control problem’s complexity and non-linearity efficiently, offering a significant reduction in computational demand during operation compared to traditional model-based optimization approaches [26]. Furthermore, by massive training with varied simulations that include random noises, disturbances, and dynamic changes [61, 66], model-free RL-based controllers can learn to adapt and stabilize the system under a wide range of conditions.

Despite their effectiveness, neural controllers often operate as a “black box” to developers, making it difficult to predict or comprehend their failure scenarios. This paper investigates how vulnerabilities of NNs affect the worst-case safety of quadrupedal locomotion controllers, and how they can be “patched” to robustify the controllers. Compared to another

line of recent works using distributional RL to optimize the rewards for the most extreme case with large perturbations during training [51, 35, 55], our work can even identify subtle vulnerabilities of controllers during deployment, requiring no transparency of the controller training.

B. Safety Validation on Locomotion Controller

To improve the safety of robot controllers, an effective safety validation process is necessary. However, this is nontrivial due to two primary challenges: the limited validation scenarios that may not fully represent the real world’s complexity, and the high computational cost associated with searching for rare failures.

Regarding the scenarios, some of existing works develop standard push-recovery testbeds by applying fixed external perturbations to the legged robot with different control policies [8, 64, 65]. Zhang and Yang [68] generate several terrain templates that is challenging and realistic to test the robustness of locomotion policies. However, these testbeds with fixed settings cover only a very limited range of scenarios. In contrast, real-world situations present more complex challenges.

Regarding the searching efficiency, as the control problems are time-sequential in a high-dimensional space, naive searching methods can be costly due to the ‘curse of dimensionality’ [6]. Hence, optimization-based methods, such as RL, are adopted to speed up the validation process in recent works. For example, in autonomous driving some use RL to learn successful adversarial attacks that uncover the hiding risks [30, 13, 18]. In the context of quadrupedal robots, a recent work involves using an evolutionary algorithm to identify adversarial attacks targeting joint torques [43]. However, scenarios involving extreme joint torque failures are uncommon in real-world settings, and this work’s analysis is based on a simplified “Ant” model, lacking validation with actual robots.

C. Robustify Control Policy with Adversarial Samples

Adversarial attacks can be introduced during the training phase to strengthen the resilience of control policies [44]. For instance, Hu et al. [22] shows that adversarial objects can be used to develop a more robust policy for grasping. Similarly, the robust adversarial RL technique [48] has been effective in enhancing robustness in competitive scenarios, such as active object tracking [71, 72, 16], quadruped object following [59], and simulated humanoid boxing [19].

However, most related studies concentrate on interactions between legged robots and other agents, treating these interactions as self-play [19, 59]. In contrast, our paper models sensor noises, command signals, and perturbation events on the quadrupedal robot as adversarial samples, which are further leveraged to enhance the controller’s robustness. We validate the efficacy of our approach both in simulation and on a actual robot.

In addition to employing well-grounded adversaries based on field experiences, our work designs distinct policy and adversarial loss function, inspired by [69, 49, 14, 46, 47,

63, 34, 33]. We direct the adversary to prioritize safety violations rather than the typical zero-sum formulation that seeks to minimize the policy’s training rewards. This focus is crucial for safety-critical systems, such as legged robots, where rewards are often dominated by performance objectives (e.g., velocity tracking accuracy), and minimizing these does not necessarily result in catastrophic failures.

III. PRELIMINARIES

A. Quadrupedal Neural Locomotion Policy

Neural controllers of quadrupedal locomotion map their observations to the action space which can be joint targets tracked by PD controllers [50] or parameters that are convertible to joint-level outputs [32, 42]. The observations typically come from onboard sensors, such as IMU, joint encoders, or height scan from lidars and depth cameras.

B. Reinforcement Learning

The robot discrete-time control problem could be formulated as a Markov Decision Process (MDP) to maximize the total task reward as Eq. 1. It is defined by (S, A, R, P, γ) , including states, actions, reward function, state transition probabilities, and discount factor. The policy π is a distribution over action given states as $\pi(a_t \in A | s_t \in S)$ and receive reward $r_t \in R$ through interaction with the environment, following the transition probability $P(s_{t+1} | s_t, a_t)$. In deep RL, the policy is parameterized by neural network parameters θ , which is optimized towards

$$\theta^* = \arg \max_{\theta} \mathbb{E}_{\pi_{\theta}} \left[\sum_{t=0}^{\infty} \gamma^t r_t \right]. \quad (1)$$

Considering the large number of samples required for learning, many works train control policies with the parallel simulation pipeline [25, 40] to speed up learning, avoid hardware damage during sample collection, or obtain privileged information.

C. Domain Randomization

Robots are expected to function in the real world. However, there is a large gap between simulation and the real world, necessitating techniques such as domain randomization (DR) [61] to bridge the gap. DR randomizes the environment settings, noises, and perturbations in simulation so that the learned policy can generalize to the real world. Despite the efficacy of DR for sim-to-real transfer, we find that the learned neural policy still exhibits vulnerabilities.

D. Definition of Robust Policy

In this paper, we focus on deliberately inducing failures in quadrupedal controllers, scenarios usually circumvented as termination conditions during the training of locomotion policies. Examples include the robot falling over or experiencing collisions at its base, where weight and sensors are heavily centralized. The robustness of the policy is defined by its capacity to prevent such failures in the face of adversarial attacks.

IV. LEARNING ADVERSARIAL ATTACKS ON QUADRUPEDAL LOCOMOTION

A. Adversarial Space

Selecting the appropriate attacks for a robot controller involves determining an adversarial space that effectively reveals the vulnerabilities of a locomotion policy. The initial step is to identify realistic adversarial scenarios that a robot is likely to face during actual deployment. We focus on sensor noise, external disturbances, and hazardous user commands as the main sources of failure. Consequently, we identify the observation space, the command space, and the perturbation space as potentially vulnerable to attacks.

For each adversarial space, it’s important to define appropriate ranges and rates of change. Extremely large attacks, such as external forces of 10,000 N on a 50-kg robot, or user commands changing at a rate of 100 m/s², are unrealistic and would not provide useful insights into a policy’s robustness under typical operational conditions. Hence, the selection of these ranges and their rates of change must be informed by practical experience in field robotics. In our experiments, we carefully adjust the boundaries and rates of change of the generated adversarial inputs to ensure they remain within realistic limits.

Regarding the command space, locomotion policies typically track velocity commands from the user or the navigation module, namely the linear velocity in the $x-y$ plane and the angular velocity around the z -axis. We attack the command space by giving malicious commands to be tracked.

Regarding the observation space, most of the existing locomotion policies use gathered base rotation and joint state data from the state estimator and joint encoders. Modern joint encoders exhibit negligible errors [1], while the pose estimation [4, 5] continues to demonstrate significant errors caused by the noisy IMU sensor and modelling mismatch, as shown in Fig. 5. Therefore, we ignore joint encoder errors in this paper, and limit the attacks to the rotation errors obtained from the robot’s state estimator.

Regarding the perturbation space, we attack the robot with external forces that are not observed by the controller. These forces can be applied on any body part, exemplified by Sec. V-A where forces target the base to simulate pushes and payload, and by Sec. V-B where forces target the feet to simulate stumbling and slippage. Note that we can only control the forces in simulation. Therefore, we do not apply perturbations during the real-world validation of the attack.

B. RL-based Attack Learning with Lipschitz Regularization

To efficiently search for failures that are time-sequential, we train adversarial RL policies that generate attacks based on the robot’s state. For an adversary policy $\pi_{\theta_{adv}}$ parameterized by θ_{adv} , we optimize the parameters towards

$$\theta_{adv}^* = \arg \max_{\theta_{adv}} \mathbb{E}_{\pi_{\theta_{adv}}} \left[\sum_{t=0}^{\infty} \gamma^t (-\mathbb{1}_{alive} + r_t^{adv}) \right] - \lambda \cdot \prod_i \|\theta_{adv}^i\|_{\infty}, \quad (2)$$

where $\mathbb{1}_{alive}$ constantly penalizes the adversarial policy if the robot doesn't fail, r_t^{aux} is the summation of auxiliary rewards that facilitate exploration, λ is the coefficient of Lipschitz regularization (the last term in the equation), and θ_{adv}^i is the i -th layer's weights of the adversarial policy.

RL solutions to our adversarial attack problem empirically adopt a "bang-bang" strategy to maximize the robot's instability (e.g., Fig. 2), but realistic adversarial attacks should be more smooth, like the real-world data in Fig. 5. Therefore, we regularize the infinity norms of the weights to constrain the Lipschitz constant of the adversary policy network, thereby making the outputs more smooth, inspired by [54] and [38]. Here we assume the adversary policy is a fully-connected network with 1-Lipschitz activation functions (such as ReLU), so the product of infinity norms make an upper bound of the Lipschitz constant [38]. In practice, we use the PPO algorithm [52] and add the regularization term to the loss.

The auxiliary rewards are

$$r_t^{aux} = c_{orient} \cdot g_z + c_{shake} \cdot (\omega_x^2 + \omega_y^2) + c_{torque} \cdot \sum_j \text{ReLU}\left(\frac{|\tau_j|}{\tau_{lim,j}} - 1\right), \quad (3)$$

where g_z is the z value of the normalized projected gravity (-1 for standing and 1 for lying), ω is the angular velocity in the base frame, τ is the joint torques, and τ_{lim} is the soft torque limits. These three terms respectively encourage the bad orientation, the shaking base, and the torques out of limit. They guide the learning of effective adversarial attacks with dense rewards.

C. Close the Loop: Finetuning with Adversarial Samples

Adversarial attacks not only expose the vulnerabilities of the neural controller, but can also be used for finetuning to further improve the control policy against the weaknesses. Finetuning can be done by simply substituting a portion of the robots' DR with adversarial samples.

Specifically, our finetuning process involves repeating the policy learning with warm-started actor and critic networks based on the previous checkpoint. During this phase, some of the rollouts continue to undergo domain randomization (DR) as in the initial training, while the remainder are subjected to adversarial attacks that are generated based on their states and actions. The optimization objective for both training and finetuning of the control policy remains the same, with the only difference being the state and action distributions due to adversarial attacks.

Given that adversarial samples can easily make robots fail, applying them to a high proportion of robots can lead to over-conservative behaviors. Therefore, it is important to balance the performance-robustness trade-off during finetuning.

V. CASE STUDIES

Our proposed method is not limited by the robot choice and the neural controller design. In this paper, we use the ANYmal robot [24] and train two control policies to exemplify our method. The first one is an end-to-end Multi-Layer

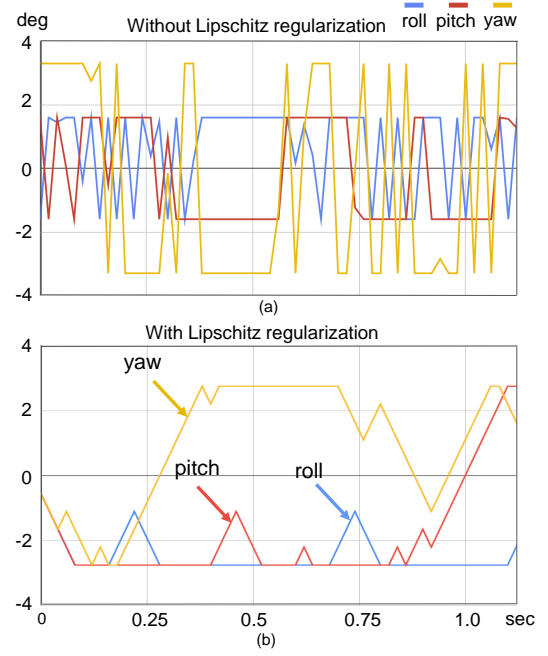


Fig. 2: The learned attack sequences under different methods, both able to cause the controller to fail in simulation. (a) is from the vanilla reinforcement learning method, whose adversarial output is in the bang-bang fashion and less practical; (b) is from the proposed method, whose adversarial output is smooth and more realistic for real-world scenarios. As shown in Fig. 5, real-world errors do not frequently oscillate between positive and negative values.

TABLE I: Observations of Attacked Locomotion Policies

Observation type	Input	Dims
Proprioception ^[B,M]	command	3
	gravity vector	3
	body velocity	6
	joint position	12
	joint velocity	12
	previous joint target	12
Exteroception ^[M]	height scan samples	208
History and others ^[M]	joint position history (3 time steps)	36
	joint velocity history (2 time steps)	24
	joint target history	12
	CPG phase information	13

Note [B] denotes the blind *didactic policy* in Sec.V-A, [M] denotes the perceptive *Miki policy* in Sec.V-B.

Perceptron (MLP) policy proposed by Rudin et al. [50] for blind locomotion, used as a didactic toy example because of its near-minimal implementation. The second one is the state-of-the-art robust locomotion policy which has gated recurrent units (GRU) [11] in the network, uses central pattern generator (CPG) parameters as the action [27], used in a mission of the DARPA Subterranean (SubT) Challenge [42, 62].

A. Didactic Example: Adversaries on Minimal Blind Policy

1) *Locomotion Policy*: We use the open-source implementation of [50] to train the blind locomotion policy (also called the *didactic policy*), which takes only single-frame proprioception as inputs (see also Table I). The policy network

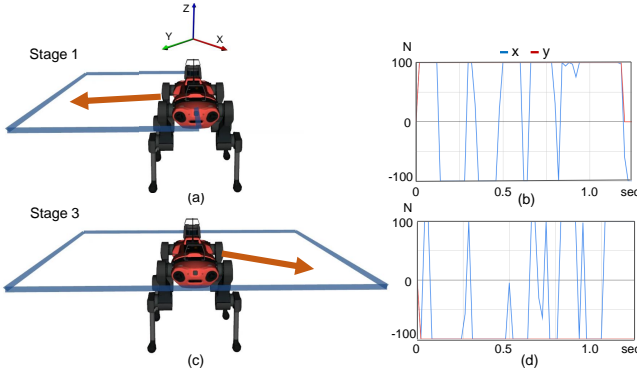


Fig. 3: Didactic example to show the proposed method being effective. (a)-(b): Stage 1: we limit the y values of the adversarial forces to be positive (c)-(d): Stage 3: we allow the y values of the adversarial forces to be either positive or negative when attacking the finetuned controller. During reattacking, the learned adversary effectively finds the unpatched weakness by leveraging the previously unused negative y forces.

of the controller is an MLP with 2 hidden layers, and outputs joint targets to be tracked by a PD controller.

2) *Adversary Settings*: Regarding the adversarial policy, we use the same observations as that of the locomotion policy, and an MLP with 2 hidden layers as the policy network. The outputs are the pushing forces on the base in the $x-y$ plane updated at the same frequency of the locomotion policy with the maximum force 100 N on each axis.

3) *Attack, robustification, and reattack*: To show that our adversarial attacks can effectively cause failures, we conduct a simple controlled experiment in simulation:

- 0) We command the robot to walk forward on the flat terrain with 0.4 m/s.
- 1) In Stage 1, we purposely set the adversarial force to have only x and positive y values as Fig. 3(a). The learned adversarial attacks can make the robot fall over.
- 2) In Stage 2, we finetune the original control policy with the learned adversarial attacks in Stage 1 to improve its robustness against the biased perturbations.
- 3) In Stage 3, we train a new adversary to attack the robustified policy, and allow the force to have both positive and negative y values as Fig. 3(c). The attack succeeds again and tends to leverage perturbations from the negative y axis.

This case presents a basic pipeline of attack-defense-reattack, and will be reused for further discussion in Sec. VI.

4) *Baselines*: To demonstrate the necessity and efficacy of the learning-based adversarial attack, we also conduct experiments for the following standard test (ST) settings as baselines:

- ST 1: apply a random constant pushing force [32];
- ST 2: apply random pushing forces at 0.5 Hz [65];
- ST 3: apply maximum pushing forces in random directions at 0.5 Hz;
- ST 4: apply a random impact [8, 64], here we adjust it

TABLE II: Adversarial Space and Range Values against *Miki Policy*

Adversarial space	Range values
Twist command	x/y : $(-0.5, 0.5)$ m/s yaw: $(-0.5, 0.5)$ rad/s
Orientation error	Roll/pitch/yaw: $(-3.0^\circ, 3.0^\circ)$
Perturbation force	End effectors: $(-15.0, 15.0)$ N

by applying the perturbation force for 0.2 s to ensure a controlled comparison.

The attack range remains consistent with Stage 1, where perturbations are applied to the robot’s base, utilizing the *didactic policy* as the locomotion controller.

Our simulation experiments show that, *none* of the four STs above can induce any failure through 1000 trials, whereas the learned adversary can fail the controller with a 100% probability. Although these STs are useful for evaluating the effects of large perturbations, they prove incapable of uncovering the subtle perturbations that are the focus of our study.

B. Attacks on DARPA-SubT-Winning Locomotion Policy

1) *Locomotion Policy*: In this case, the locomotion policy is a robust perceptive policy proposed by Miki et al. [42] (also called the *Miki policy* in this paper) trained for challenging scenarios like subterranean caves and snowy mountains. The policy is empirically robust against observation noises and external perturbations [62], and the failure cases are rare in the real world.

2) *Adversary Settings*: Regarding the adversary, the policy network is an MLP with 3 hidden layers, and the observations are the same as that of the locomotion policy (see Table I). The attacks can be applied to the observation space, the command space, and the perturbation space, updated at the same frequency of the locomotion policy. The attacked values and their ranges are listed in Table II, with the change rate limited to $0.1 \times$ maximum value per timestep (0.02 s).

In the following experimental results, we employ two settings. One (Sec. V-B3) attacks only the observation space and the command space to verify consistency between simulation and the real world, given that perturbations cannot be accurately replicated in the real world. The other (Sec. V-B4) attacks all of the three spaces, and the learned adversary is used in simulation to further robustify the locomotion policy.

3) *Learned Adversary with Real-World Validation*: Our method can effectively learn adversarial attacks to fail the locomotion policy with only the observation space and the command space. To verify that the simulation results align with the real-world performance, we conduct the same attacking test in simulation and the real world. We also try rescaled orientation attacks to identify the minimal range to cause the failure. The motion of the robot under attack is depicted in Fig. 1.

The test results presented in Fig. 6 demonstrate the sim-to-real transferability of our learned adversary, and the corresponding attack sequence is visualized in Fig. 4 with compar-

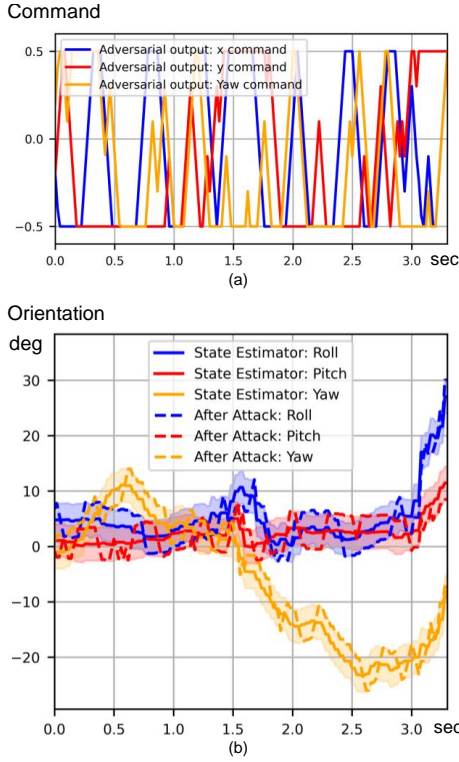


Fig. 4: Plots of adversarial output to fall over the real robot. (a) is the adversarial attack sequence in the command space; (b) is the attack sequence in the observation space, which outputs the orientation errors within 3° . Note that to avoid the damage on the physical hardware, early-stop is triggered before the robot falls down.

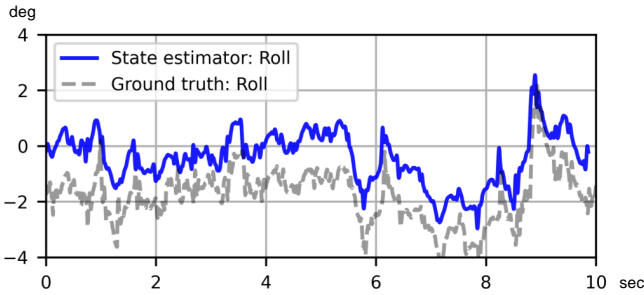


Fig. 5: Orientation data from the state estimator and ground truth (via motion capture) when the robot traverses flat indoor terrain, with peak errors reaching almost 3° . These estimation errors tend to significantly increase on uneven or slippery outdoor surfaces, or when the robot is subjected to perturbations [4].

ison to a typical state estimation error curve in Fig. 5. These results indicate that, vulnerabilities identified in simulation do reflect the controller’s weaknesses in the real world, and the risk of encountering such failures during operation is non-negligible.

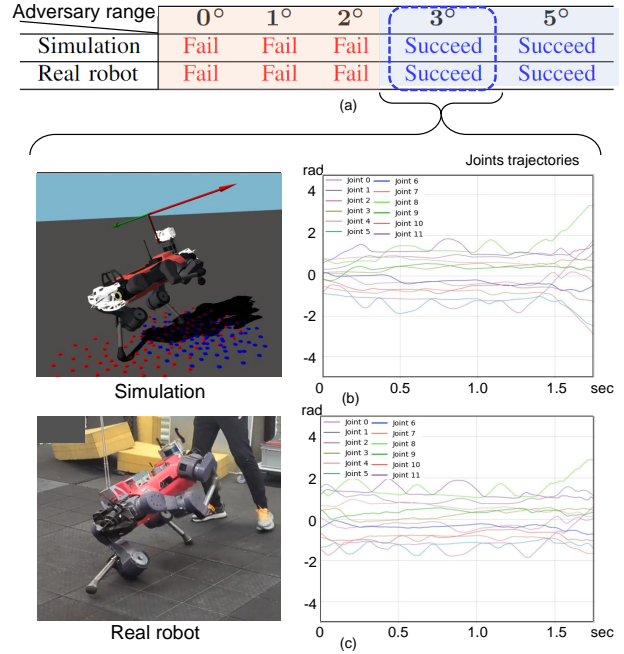


Fig. 6: Sim-to-real comparison under adversarial attacks. (a) shows the adversary cannot successfully attack the controller when the orientation error is rescaled to be no larger than 2° . This applies to both simulation and the real robot. (b) is falling-over snapshots and the joint trajectories in simulation when the orientation error is bounded by 3° . (c) is falling-over snapshots and the joint trajectories in the real world when the orientation error is bounded by 3° .

4) *Robustification*: We train an adversary with all of the three adversarial spaces, and finetune the policy in simulation with a 5% probability of encountering the learned adversary instead of randomized perturbations. We expect that the finetuned policy is more robust against perturbations and state estimation errors. This is verified not only by a failed trial of reattack learning, but also by the real-world robust performances indoors (Fig. 7) and outdoors (Fig. 8).

To further assess the robustness of the finetuned policy, we conduct a re-attack at a larger scale, with the results presented in Fig. 9. During the real-world test, the operator also applied additional perturbations. Despite these challenging conditions, the robot maintained its robustness and resisted all attacks.

To quantitatively examine the performance change of the new controller, we compare the joint torques and command tracking accuracy before and after robustification, as presented in Table III. Despite the improved robustness, we find no significant change in the tracking performance.

VI. ANALYSES AND EXTENSIVE STUDIES

A. Is DR sufficient for controller robustness?

Domain randomization (DR) randomizes the properties of the environments, with the expectation that the policy will work across all these varied settings. However, we *hypothesize* that naive randomization methods (e.g., time-invariant

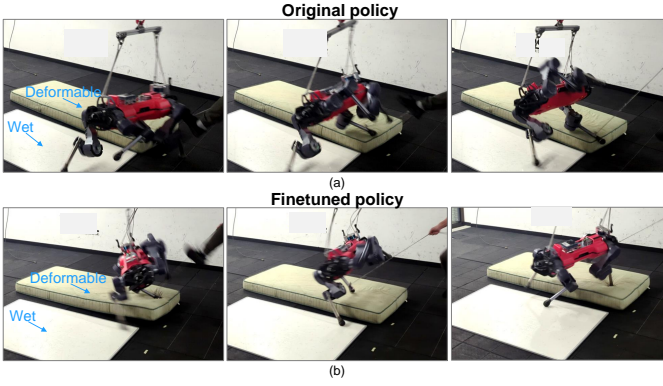


Fig. 7: Challenging indoor perturbation experiments on a slippery wet whiteboard and a deformable mattress. (a) shows the performance of the original *Miki policy*, which quickly loses balance; (b) is the robustified policy by our proposed method, showcasing more robust reactive behaviors.



Fig. 8: Outdoor experiments in the challenging terrains, including rough slopes, slippery ditches, deformable sand, and steep mountain roads with small rocks. The finetuned policy maintains the traversability while being more robust against adversarial scenarios.

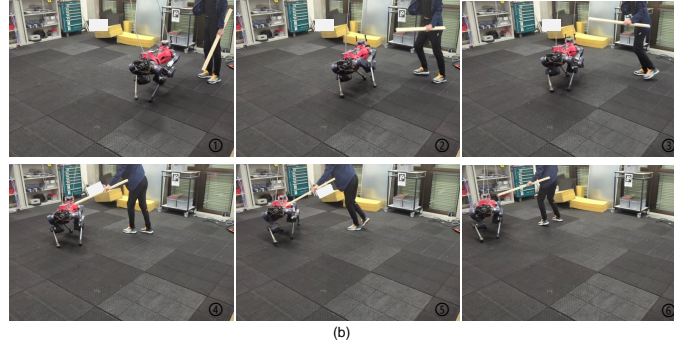
uniform distributions or Gaussian distributions) may hardly cover threatening time-sequential adversaries in the high-dimensional adversarial space. Consequently, DR may be insufficient in guarding the controller against adversarial attacks.

To verify this, we train a locomotion policy similar to the *didactic policy* and with exteroception observations. During training, we add randomized perturbation forces up to 100 N and varies at 5 Hz. We then train an adversary policy with up to 100 N perturbation forces to attack this policy. As shown by Fig. 10 (“policy-DR”), the robot still fails under the learned adversarial attacks, though it can survive larger pushing forces compared to the policy without DR.

Our conclusions are: 1) DR is not sufficient to ensure the ro-

Re-trained adversary range	0.1°	1°	2°	3°	5°
Simulation	Fail	Fail	Fail	Fail	Fail
Real robot	Fail	Fail	Fail	Fail	Fail

(a)



(b)

Fig. 9: Re-trained adversaries fail to attack the finetuned controller. (a) Compared to the controller before finetuning in Fig. 6, the finetuned controller can withstand adversaries at a larger scale. The orientation error is bounded by 5° , informed by expert knowledge for realistic attack scenarios. (b) In real-world conditions, the finetuned controller maintains its robustness, effectively resisting both re-attacks and additional perturbations applied by the operator.

TABLE III: Comparison of Torques and Tracking Performance Before and After Robustification of *Miki Policy* in Different Terrains

Terrain	Avg. joint torque	Avg. tracking error
Flat terrain	+0.42 Nm	+0.002 m/s
Rough terrain	-1.76 Nm	-0.006 m/s
Standard stairs	+0.26 Nm	+0.008 m/s
Random stones	-0.26 Nm	+0.006 m/s

bustness against adversarial attacks; meanwhile, 2) adversarial attacks can be complementary to DR.

B. Attacks as robustness indicators

Quantitatively assessing the robustness of control policies remains a challenge, as standard tests are inefficient in identifying worst cases. As an initial exploration, we claim that adversarial attacks, as the effective tools to search the worst cases, can be leveraged to indicate the robustness of a control policy.

we conduct the following experiments to demonstrate this:

- 1) We train a locomotion policy with the same random perturbation setting as VI-A (called “policy-DR”). We train another one without DR (called “policy”).
- 2) We add the perturbation ranges and exteroception to the adversary’s observation space. We also randomize the perturbation ranges during training so that the agent is aware of the maximum perturbation magnitude.
- 3) We train adversaries to respectively attack the two locomotion policies. We then finetune “policy-DR” under adversaries. This refined version of the policy, now

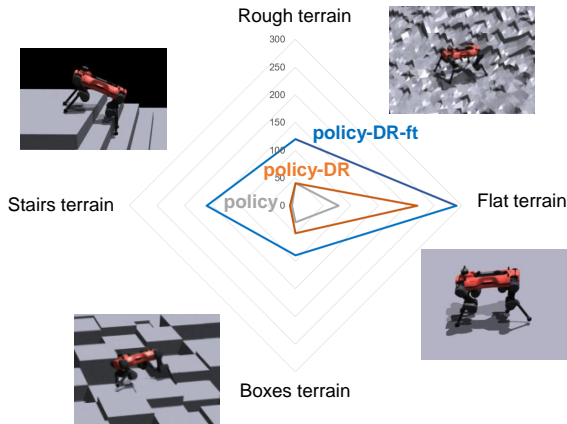


Fig. 10: Minimum perturbation force ranges that cause the controller to fall over vary across different terrains, highlighting that finetuning with learned adversarial attacks can significantly diminish its vulnerabilities on challenging, uneven terrains.

TABLE IV: Comparison of Different Combinations in Adversarial Space

	C	O	P	C+O	C+P	O+P	C+O+P
Adversaries results	F	F	F	S	S	S	S

C/O/P denotes the command/observation/perturbation space, S/F denotes the success/failure to cause robots to fall over with the corresponding adversarial space.

termed "policy-DR-ft", is then re-examined under newly learned adversarial attacks.

- 4) We execute the three adversarial policies on the corresponding locomotion policies. During execution for each, we gradually reduce the perturbation range until the locomotion policy can withstand its adversary.
- 5) We record the minimal ranges to make the control policies fail in different terrains. The values are reported in Fig. 10.

As we can see, Fig. 10 clearly shows "policy-DR" is better than "policy", while "policy-DR-ft" shows the best safety. This example shows how adversarial attacks can be potentially used to assess the robustness of neural controllers.

C. Multi-modality is essential

Previous works in legged robots only investigate a single adversarial modality [8, 64, 65, 59]. Here we claim that, *multi-modal* adversaries are more effective in searching for the worst cases.

Using the *Miki policy* in Sec. V-B as the example, we train adversaries with different adversarial spaces, and the results are presented in Table IV.

Based on the results, we conclude that multiple modalities are essential in finding effective and subtle adversaries. Combining any two or all three of the adversarial subspaces can achieve successful attacks despite larger search spaces, while no unimodal attacks can succeed as we limit the attack

TABLE V: Ablative Analysis of Various Adversarial Reward Combinations

Adversarial reward setting	Convergence iter.	Falling over
Eq. 2 (Proposed)	240	Succeed
Zero-sum fashion	320	Fail
Set $\mathbb{1}_{alive} = 0$	280	Fail
Set $c_{orien}/c_{shake} = 0$	650	Succeed
Set $c_{torque} = 0$	720	Succeed

The first setting is our proposed adversarial reward; in the zero-sum setting, the adversarial reward is set to be the negative of the locomotion reward in [50]; for the other approaches, parts of the auxiliary rewards in Eq. 2 and Eq. 3 are set to 0.

ranges. Multi-modal adversaries also better reflect the real world where failures of a well-developed controller are often due to the combination of multiple factors.

D. Ablating reward function design

A conventional approach in adversarial reinforcement learning sets the adversarial reward as the negative of the policy reward, thus formulating the problem within a zero-sum context. However, the complexity of various robot control challenges motivates the development of distinct policy and adversarial rewards [46, 47, 63, 34, 33], raising the question of which approach is more effective for our problem addressing complex locomotion reward structures in real-world applications. Furthermore, we aim to validate the necessity of our auxiliary terms as delineated in Equations 2 and 3.

To illustrate the need for distinct adversarial rewards, we conducted an ablation study. Initially, we adopted the zero-sum formulation, setting the adversarial reward as the negative of the original policy's reward. This configuration was ineffective for achieving our objective of inducing the robot to fall over. To more comprehensively evaluate the effectiveness of our proposed adversarial reward structure, we conducted further ablation analyses by systematically removing each component; the outcomes are detailed in Table V. Additionally, the auxiliary reward facilitated greater exploration in the adversarial space, enabling the generation of diverse failure scenarios [14, 39]. More details are provided in Section VI-G.

We conclude that in real-world applications, many components of the original control policy are specifically designed to shape complex behaviors, including metrics such as velocity tracking, foot clearance, and torque penalties. Simply inverting the policy reward directs the adversary towards minimizing task performance (evidenced by poor velocity tracking and enlarged torques) rather than inducing catastrophic failures. This inefficacy arises in our problem because a fall produces a reward of 0, significantly lower than the penalties from suboptimal walking. Similar challenges are observed in systems using cross-entropy loss [14]. While recent theoretical advances have begun to more thoroughly investigate non-zero-sum adversarial training, employing multi-objective optimization [2] or integrating the adversarial objective as a constraint [49], the domain still demands extensive exploration to fully develop and clarify the underlying theories.

TABLE VI: Comparison of successful human hackers and proposed adversarial method

	Winner 1	Winner 2	Winner 3	Our Adversary
Avg. survival time (s)	4.0	5.8	11.5	1.38
Avg. falling roll rate (rad/s)	1.77	1.53	2.80	5.04

E. Can humans be good hackers?

To improve robustness, control policies are often iteratively refined by humans through intuitive testing of perturbations and sensor noise on the robot. The question arises: can humans efficiently identify the controller’s weaknesses, and is there a real need for the computational method proposed in this paper?

To verify this, despite the difficulty in formally evaluating human performance (e.g., inviting a human champion against machine intelligence [58] [29]), we make a first attempt by organizing a small *AI safety challenge* on locomotion controller hacking.

In the challenge, we provide the *didactic policy* in simulation and ask participants to attack the robot. The robot is consistently commanded to walk on flat ground with a fixed forward velocity of 0.4 m/s. Participants are allowed to use joysticks to apply 2D pushing forces (up to 100 N on x, y axes) to the robot base and to overwrite the lateral and yaw velocity commands respectively within 0.5 m/s and 0.5 rad/s.

Regarding the machine intelligence counterpart, for fair comparison, we train an adversarial agent without joint state observations as humans have only limited observations visually displayed in the simulator’s GUI. Besides, we limit the adversary’s output frequency to 5 Hz as humans cannot move the joysticks at high frequencies.

We successfully invited 100 volunteer participants with a background in robotics to join the challenge. Due to resource constraints, we gave each participant 5 minutes to attempt their attacks after they learned how to operate the system. During the whole challenge, only 3 of the participants successfully made the robot fall. We compare the performances of the successful participants and our learned adversary in Table VI. Compared to humans, our learned adversary can effectively attack the controller in *less time* and cause more severe damage through *more violent falls*.

We conclude that, identifying weaknesses in RL-based locomotion controllers through human experience is challenging, even when there is no perturbation randomization for robustness during training. On the other hand, the computational method proposed in this paper is crucial to be a complement and help uncover vulnerabilities in black-box neural controllers.

F. Attacks that leverage terrains

Although our real-world attack experiments are conducted on flat ground for safety reasons, uneven terrains could potentially present more challenging scenarios.

Indeed, we find in simulation that the learned adversary against the *Miki policy* in Sec. V-B, with exteroceptive obser-

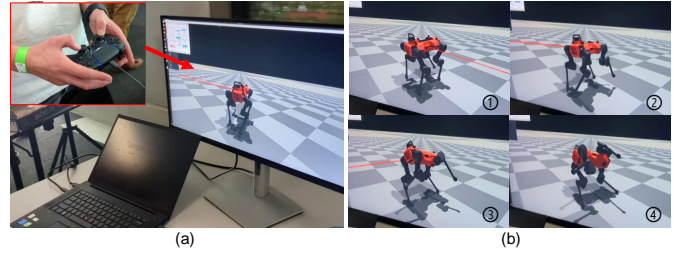


Fig. 11: *AI safety challenge* in hacking robot controller. (a): Human can use remote controller to apply perturbation force and send twist command to the robot in the simulation; (b): Snapshots of the Winner 2 in falling over the controller.

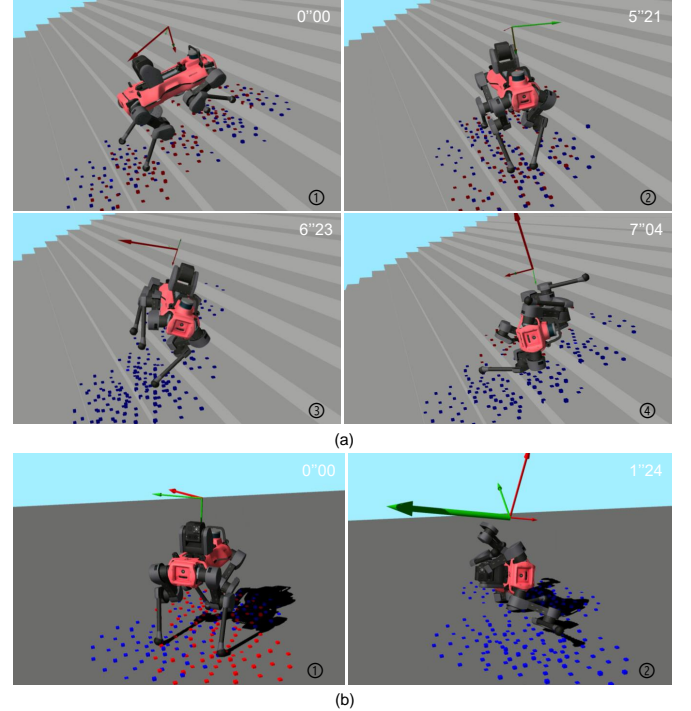


Fig. 12: Learned adversarial agent with exteroceptive observations leverages the terrain for effective attacks. (a): Stairs are used strategically, where the adversary first rotates the robot to position it advantageously before initiating a successful fall-over attack. (b): On flat terrain, conversely, the adversary launches the attack directly without the rotation maneuver observed on the stairs.

variations, can actively exploit uneven terrains to enhance the attack. Fig. 12(a) illustrates an instance where the learned adversary first directs the robot towards the edge of stairs and then utilizes the stairs to induce a fall. In contrast, such behavior is not observed on flat ground, as depicted in Fig. 12(b).

G. Can we uncover diverse weaknesses?

The proposed RL-based method empirically converges to a specific attack strategy in each run, while the controller’s weaknesses may be non-unique. Hence, it is preferable to

TABLE VII: Diverse Adversarial Attacks and Corresponding Weights

	Falling over	Self collision	Torque over limits
c_{orient}	0.1	0	0
c_{shake}	0.05	0	0
c_{torque}	5.0	0	20.0
$\mathbb{1}_{alive}$	1.0	1.0	1.0

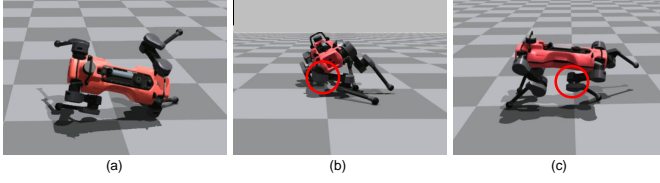


Fig. 13: Results of learned diverse failure scenarios that could cause damage or prompt termination in the actual robot: (a) Falling over; (b) Self-collision with the base hitting a front leg; (c) Knee motor exceeding the torque limit due to the configuration and a single foot touchdown.

uncover a variety of weaknesses rather than a single one.

The problem can be addressed in two ways. One is to train different adversaries with different random seeds for neural network initialization [70], which lacks controllability. The other is to guide the adversaries with diverse reward settings [60] as shown here.

To be specific, we adjust the weights of different reward terms in Eq. 3, as listed in Table VII. With different reward weights, we can learn different adversary behaviors, as shown in Fig. 13 where we attacked the command space and the observation space.

H. Illustrating effects of robustification: saving the unguarded

We designed an experiment to illustrate the efficacy of our proposed methodology through multiple rounds of attacking and finetuning. To this end, we use the *didactic policy* but deliberately bias the domain randomization during initial training: the external forces can only be sampled within 45° from the x -axis in the robot’s sagittal plane (both forward and backward), as illustrated in Fig. 14(a).

As shown in Fig. 14(b), the initial policy is predictably vulnerable to lateral pushes according to our push tests. We trained the first adversary, which successfully identified and exploited weaknesses when pushing from the left, causing the robot to fall as Fig. 14(c). After finetuning with the first adversary, the controller became more robust against leftward pushes but remained vulnerable to rightward pushes. Subsequently, a second adversary was trained, which pinpointed and exploited weaknesses from the right, revealing further vulnerabilities in the controller as Fig. 14(d). By finetuning with the second adversary, the policy was significantly enhanced, showing improved resistance to pushes from both sides.

This example underscores the utility of adversarial attacks, particularly when the domain randomization employed fails to sufficiently cover the range of scenarios necessary for robust robot operation.

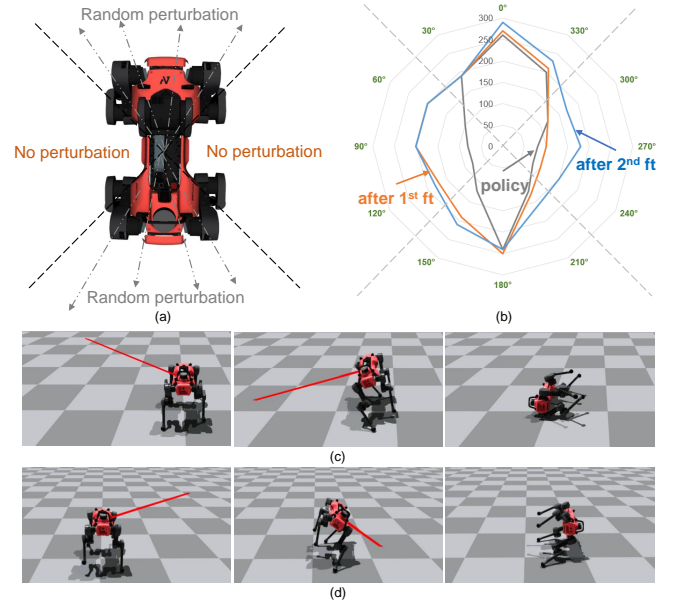


Fig. 14: Example to illustrate the effects of robustification through multiple rounds of attacking and finetuning (ft). (a) Initial setup: The domain randomization is deliberately biased to exclude lateral forces. (b) Minimal constant force perturbations required to induce failure: Comparisons among the initial policy (gray), the policy finetuned after the first adversarial attack (orange), and the policy further refined after both adversarial attacks (blue). (c) First attack findings: Reveals control policy weaknesses when the robot is pushed leftward. (d) Second attack outcomes: Exposes vulnerabilities when the robot is pushed rightward.

I. Bonus: attacking MPC

Principally, our proposed method is not limited to the specific RL-based controller or any robot type. Here, we demonstrate that the model predictive controller (MPC) can also be subjected to learning-based attacks which are time-variant and state-dependent. To this end, we conducted successful attacks on an open-source MPC [17] for the Unitree A1 robot¹, as shown in Fig. 15

VII. CONCLUSION AND LIMITATIONS

In this paper, we present a learning-based adversarial attack framework targeting neural network based quadrupedal locomotion controllers. The learned adversaries can effectively uncover the vulnerabilities of the controllers, including the state-of-the-art DARPA-winning locomotion policy. By formulating the problem into a Markov Decision Process, a reinforcement learning method with Lipschitz regularization is employed to efficiently learn realistic sequential attacks. The efficacy of our method is validated both in simulation and on the actual robot, with consistent behaviors between the simulation and the real world.

¹The controller was developed by Di Carlo et al. [17] for mini cheetah, and we use the adapted implementation for A1 by Chen and Nguyen [9].

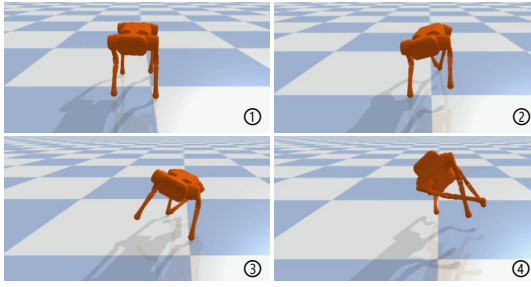


Fig. 15: Results of learned adversarial attacks on a MPC controller in the simulation. Our attack is in command and observation space with the same range and reward setting used for the *Miki policy* attacking.

Furthermore, we demonstrate how learned adversarial attacks can be utilized to enhance the robustness of the controllers and potentially act as an analytical tool for robustness assessment. Besides, we provide extensive analyses and insights on robustness. Some key takeaways are:

- 1) Domain randomization is not sufficient to ensure the robustness against learned attacks;
- 2) Multi-modal attacks are stronger than unimodal ones;
- 3) Adversaries with exteroception can leverage terrains;
- 4) Human intelligence on attacks does not diminish the significance of our learning-based approach.

We also study how to diversify the learned attacks and whether our proposed method is applicable to other controllers such as model predictive controllers.

We hope this paper can raise awareness among legged robot researchers regarding the vulnerability in neural controllers and the critical need for comprehensive safety verification. By implementing the effective verification process, the safety and trustworthiness of robot controllers can be improved, paving the way for their secure large-scale deployment.

A. Limitations

Regarding the observation space, this paper attacks solely on the proprioception. However, for perceptive locomotion policies, the exteroceptive observations are also likely to be vulnerable, given their high dimensionality (e.g., 208-dim for the *Miki policy*). The proposed RL-based method might not be the most effective approach to output such high-dimensional sequences, and we will explore generative methods in the future.

Another limitation is the absence of theoretical guarantees to cover all of the weaknesses. Although we showcase how we can learn diverse attacks in Sec. VI-G, investigating new techniques that maintain diversity from the machine learning community can be valuable.

Furthermore, our proposed method can serve as a tool to compare different frameworks and methods. However, ensuring fairness in these comparisons necessitates substantial effort to formally verify that the policies assessed are 'optimal' for their respective reinforcement learning algorithms and robotic configurations. Currently, the diversity of robotic platforms

and sensor settings complicates direct comparisons. We plan to tackle these challenges through more rigorous comparisons in our future work.

VIII. ACKNOWLEDGMENT

The authors would like to thank Prof. Kaiqing Zhang and Julian Nubert for the insightful discussion.

This work was supported by the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation program (grant agreement No. 866480, 780883). This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 101016970.

This research was partially supported by the ETH AI Center.

REFERENCES

- [1] Dynapar Encoders. URL https://www.dynapar.com/knowledge/encoder_resolution_encoder_accuracy_repeatability.
- [2] Isabela Albuquerque, Joao Monteiro, Thang Doan, Brendan Considine, Tiago Falk, and Ioannis Mitliagkas. Multi-objective training of generative adversarial networks with multiple discriminators. In *International Conference on Machine Learning*, pages 202–211. PMLR, 2019.
- [3] Guillaume Bellegarda, Yiyu Chen, Zhuochen Liu, and Quan Nguyen. Robust high-speed running for quadruped robots via deep reinforcement learning. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10364–10370. IEEE, 2022.
- [4] Michael Bloesch, Christian Gehring, Péter Fankhauser, Marco Hutter, Mark A Hoepflinger, and Roland Siegwart. State estimation for legged robots on unstable and slippery terrain. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 6058–6064. IEEE, 2013.
- [5] Michael Bloesch, Michael Burri, Hannes Sommer, Roland Siegwart, and Marco Hutter. The two-state implicit filter recursive estimation for mobile robots. *IEEE Robotics and Automation Letters*, 3(1):573–580, 2017.
- [6] Miles Brundage, Shahar Avin, Jasmine Wang, Haydn Belfield, Gretchen Krueger, Gillian Hadfield, Heidy Khlaaf, Jingying Yang, Helen Toner, Ruth Fong, et al. Toward trustworthy ai development: mechanisms for supporting verifiable claims. *arXiv preprint arXiv:2004.07213*, 2020.
- [7] Jan Carius, Farbod Farshidian, and Marco Hutter. Mpcnet: A first principles guided policy search. *IEEE Robotics and Automation Letters*, 5(2):2897–2904, 2020.
- [8] Hua Chen, Zejun Hong, Shunpeng Yang, Patrick M Wensing, and Wei Zhang. Quadruped capturability and push recovery via a switched-systems characterization of dynamic balance. *IEEE Transactions on Robotics*, 2023.

- [9] Yiyu Chen and Quan Nguyen. Learning agile locomotion and adaptive behaviors via rl-augmented mpc. *arXiv preprint arXiv:2310.09442*, 2023.
- [10] Xuxin Cheng, Kexin Shi, Ananye Agarwal, and Deepak Pathak. Extreme parkour with legged robots. *arXiv preprint arXiv:2309.14341*, 2023.
- [11] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.
- [12] Timothy H Chung, Viktor Orekhov, and Angela Maio. Into the robotic depths: Analysis and insights from the darpa subterranean challenge. *Annual Review of Control, Robotics, and Autonomous Systems*, 6:477–502, 2023.
- [13] Anthony Corso, Peter Du, Katherine Driggs-Campbell, and Mykel J Kochenderfer. Adaptive stress testing with reward augmentation for autonomous vehicle validation. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pages 163–168. IEEE, 2019.
- [14] Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In *International conference on machine learning*, pages 2206–2216. PMLR, 2020.
- [15] Francesco Croce, Maksym Andriushchenko, Vikash Sehwag, Edoardo Debenedetti, Nicolas Flammarion, Mung Chiang, Prateek Mittal, and Matthias Hein. Robustbench: a standardized adversarial robustness benchmark. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.
- [16] Alessandro Devo, Alberto Dionigi, and Gabriele Costante. Enhancing continuous control of mobile robots for end-to-end visual active tracking. *Robotics and Autonomous Systems*, 142:103799, 2021.
- [17] Jared Di Carlo, Patrick M Wensing, Benjamin Katz, Gerardo Bledt, and Sangbae Kim. Dynamic locomotion in the mit cheetah 3 through convex model-predictive control. In *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 1–9. IEEE, 2018.
- [18] Shuo Feng, Haowei Sun, Xintao Yan, Haojie Zhu, Zhengxia Zou, Shengyin Shen, and Henry X Liu. Dense reinforcement learning for safety validation of autonomous vehicles. *Nature*, 615(7953):620–627, 2023.
- [19] Adam Gleave, Michael Dennis, Cody Wild, Neel Kant, Sergey Levine, and Stuart Russell. Adversarial policies: Attacking deep reinforcement learning. *arXiv preprint arXiv:1905.10615*, 2019.
- [20] Ruben Grandia, Fabian Jenelten, Shaohui Yang, Farbod Farshidian, and Marco Hutter. Perceptive locomotion through nonlinear model-predictive control. *IEEE Transactions on Robotics*, 2023.
- [21] David Hoeller, Nikita Rudin, Dhionis Sako, and Marco Hutter. Anymal parkour: Learning agile navigation for quadrupedal robots. *Science Robotics*, 9(88):eadi7566, 2024.
- [22] Zhe Hu, Yu Zheng, and Jia Pan. Grasping living objects with adversarial behaviors using inverse reinforcement learning. *IEEE Transactions on Robotics*, 39(2):1151–1163, 2023.
- [23] Sandy Huang, Nicolas Papernot, Ian Goodfellow, Yan Duan, and Pieter Abbeel. Adversarial attacks on neural network policies. *arXiv preprint arXiv:1702.02284*, 2017.
- [24] Marco Hutter, Christian Gehring, Dominic Jud, Andreas Lauber, C Dario Bellicoso, Vassilios Tsounis, Jemin Hwangbo, Karen Bodie, Peter Fankhauser, Michael Bloesch, et al. Anymal-a highly mobile and dynamic quadrupedal robot. In *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 38–44. IEEE, 2016.
- [25] Jemin Hwangbo, Joonho Lee, and Marco Hutter. Per-contact iteration method for solving contact dynamics. *IEEE Robotics and Automation Letters*, 3(2):895–902, 2018.
- [26] Jemin Hwangbo, Joonho Lee, Alexey Dosovitskiy, Dario Bellicoso, Vassilios Tsounis, Vladlen Koltun, and Marco Hutter. Learning agile and dynamic motor skills for legged robots. *Science Robotics*, 4(26):eaau5872, 2019.
- [27] Auke Jan Ijspeert. Central pattern generators for locomotion control in animals and robots: a review. *Neural networks*, 21(4):642–653, 2008.
- [28] Fabian Jenelten, Ruben Grandia, Farbod Farshidian, and Marco Hutter. Tamols: Terrain-aware motion optimization for legged systems. *IEEE Transactions on Robotics*, 38(6):3395–3413, 2022.
- [29] Elia Kaufmann, Leonard Bauersfeld, Antonio Loquercio, Matthias Müller, Vladlen Koltun, and Davide Scaramuzza. Champion-level drone racing using deep reinforcement learning. *Nature*, 620(7976):982–987, 2023.
- [30] Mark Koren, Saud Alsaif, Ritchie Lee, and Mykel J Kochenderfer. Adaptive stress testing for autonomous vehicles. In *2018 IEEE Intelligent Vehicles Symposium (IV)*, pages 1–7. IEEE, 2018.
- [31] Ashish Kumar, Zipeng Fu, Deepak Pathak, and Jitendra Malik. Rma: Rapid motor adaptation for legged robots. *arXiv preprint arXiv:2107.04034*, 2021.
- [32] Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning quadrupedal locomotion over challenging terrain. *Science robotics*, 5(47):eabc5986, 2020.
- [33] Chenhao Li, Sebastian Blaes, Pavel Kolev, Marin Vlastelica, Jonas Frey, and Georg Martius. Versatile skill control via self-supervised adversarial imitation of unlabeled mixed motions. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2944–2950. IEEE, 2023.
- [34] Chenhao Li, Marin Vlastelica, Sebastian Blaes, Jonas Frey, Felix Grimminger, and Georg Martius. Learning agile skills via adversarial imitation of rough partial demonstrations. In *Conference on Robot Learning*, pages

- 342–352. PMLR, 2023.
- [35] Sicen Li, Yiming Pang, Panju Bai, Jiawei Li, Zhaojin Liu, Shihao Hu, Liquan Wang, and Gang Wang. Learning locomotion for quadruped robots via distributional ensemble actor-critic. *IEEE Robotics and Automation Letters*, 2024.
 - [36] Zhongyu Li, Xuxin Cheng, Xue Bin Peng, Pieter Abbeel, Sergey Levine, Glen Berseth, and Koushil Sreenath. Reinforcement learning for robust parameterized locomotion control of bipedal robots. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2811–2817. IEEE, 2021.
 - [37] Zachary C Lipton. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.
 - [38] Hsueh-Ti Derek Liu, Francis Williams, Alec Jacobson, Sanja Fidler, and Or Litany. Learning smooth neural functions via lipschitz regularization. In *ACM SIGGRAPH 2022 Conference Proceedings*, pages 1–13, 2022.
 - [39] Weilin Liu, Ye Mu, Chao Yu, Xuefei Ning, Zhong Cao, Yi Wu, Shuang Liang, Huazhong Yang, and Yu Wang. Multi-agent vulnerability discovery for autonomous driving with hazard arbitration reward. *arXiv preprint arXiv:2112.06185*, 2021.
 - [40] Viktor Makoviychuk, Lukasz Wawrzyniak, Yunrong Guo, Michelle Lu, Kier Storey, Miles Macklin, David Hoeller, Nikita Rudin, Arthur Allshire, Ankur Handa, et al. Isaac gym: High performance gpu based physics simulation for robot learning. In *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)*, 2021.
 - [41] Gabriel B Margolis, Ge Yang, Kartik Paigwar, Tao Chen, and Pulkit Agrawal. Rapid locomotion via reinforcement learning. *arXiv preprint arXiv:2205.02824*, 2022.
 - [42] Takahiro Miki, Joonho Lee, Jemin Hwangbo, Lorenz Wellhausen, Vladlen Koltun, and Marco Hutter. Learning robust perceptive locomotion for quadrupedal robots in the wild. *Science Robotics*, 7(62):eabk2822, 2022.
 - [43] Takuto Otomo, Hiroshi Kera, and Kazuhiko Kawamoto. Adversarial joint attacks on legged robots. In *2022 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pages 676–681. IEEE, 2022.
 - [44] Anay Pattanaik, Zhenyi Tang, Shuijing Liu, Gautham Bommannan, and Girish Chowdhary. Robust deep reinforcement learning with adversarial attacks. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pages 2040–2042, 2018.
 - [45] Xue Bin Peng, Erwin Coumans, Tingnan Zhang, Tsang-Wei Lee, Jie Tan, and Sergey Levine. Learning agile robotic locomotion skills by imitating animals. *arXiv preprint arXiv:2004.00784*, 2020.
 - [46] Xue Bin Peng, Ze Ma, Pieter Abbeel, Sergey Levine, and Angjoo Kanazawa. Amp: Adversarial motion priors for stylized physics-based character control. *ACM Transactions on Graphics (ToG)*, 40(4):1–20, 2021.
 - [47] Xue Bin Peng, Yunrong Guo, Lina Halper, Sergey Levine, and Sanja Fidler. Ase: Large-scale reusable adversarial skill embeddings for physically simulated characters. *ACM Transactions On Graphics (TOG)*, 41(4):1–17, 2022.
 - [48] Lerrel Pinto, James Davidson, Rahul Sukthankar, and Abhinav Gupta. Robust adversarial reinforcement learning. In *International Conference on Machine Learning*, pages 2817–2826. PMLR, 2017.
 - [49] Alexander Robey, Fabian Latorre, George J. Pappas, Hamed Hassani, and Volkan Cevher. Adversarial training should be cast as a non-zero-sum game. In *The Twelfth International Conference on Learning Representations*, 2024. URL <https://openreview.net/forum?id=XJ9vjEAqbx>.
 - [50] Nikita Rudin, David Hoeller, Philipp Reist, and Marco Hutter. Learning to walk in minutes using massively parallel deep reinforcement learning. In *Conference on Robot Learning*, pages 91–100. PMLR, 2022.
 - [51] Lukas Schneider, Jonas Frey, Takahiro Miki, and Marco Hutter. Learning risk-aware quadrupedal locomotion using distributional reinforcement learning. *arXiv preprint arXiv:2309.14246*, 2023.
 - [52] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.
 - [53] Fan Shi, Timon Homberger, Joonho Lee, Takahiro Miki, Moju Zhao, Farbod Farshidian, Kei Okada, Masayuki Inaba, and Marco Hutter. Circus anymal: A quadruped learning dexterous manipulation with its limbs. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2316–2323. IEEE, 2021.
 - [54] Guanya Shi, Xichen Shi, Michael O’Connell, Rose Yu, Kamyar Azizzadenesheli, Animashree Anandkumar, Yisong Yue, and Soon-Jo Chung. Neural lander: Stable drone landing control using learned dynamics. In *2019 international conference on robotics and automation (icra)*, pages 9784–9790. IEEE, 2019.
 - [55] Jiyuan Shi, Chenjia Bai, Haoran He, Lei Han, Dong Wang, Bin Zhao, Xiu Li, and Xuelong Li. Robust quadrupedal locomotion via risk-averse policy learning. *arXiv preprint arXiv:2308.09405*, 2023.
 - [56] Jonah Siekmann, Yesh Godse, Alan Fern, and Jonathan 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. IEEE, 2021.
 - [57] Jonah Siekmann, Kevin Green, John Warila, Alan Fern, and Jonathan W. Hurst. Blind bipedal stair traversal via sim-to-real reinforcement learning. In *Robotics: Science and Systems*, 2021.
 - [58] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529

- (7587):484–489, 2016.
- [59] Yujin Tang, Jie Tan, and Tatsuya Harada. Learning agile locomotion via adversarial training. In *2020 IEEE/RSJ International Conference On Intelligent Robots And Systems (IROS)*, pages 6098–6105. IEEE, 2020.
 - [60] Zhenggang Tang, Chao Yu, Boyuan Chen, Huazhe Xu, Xiaolong Wang, Fei Fang, Simon Shaolei Du, Yu Wang, and Yi Wu. Discovering diverse multi-agent strategic behavior via reward randomization. In *International Conference on Learning Representations*, 2020.
 - [61] Josh Tobin, Rachel Fong, Alex Ray, Jonas Schneider, Wojciech Zaremba, and Pieter Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 23–30. IEEE, 2017.
 - [62] Marco Tranzatto, Takahiro Miki, Mihir Dharmadhikari, Lukas Bernreiter, Mihir Kulkarni, Frank Mascarich, Olov Andersson, Shehryar Khattak, Marco Hutter, Roland Siegwart, et al. Cerberus in the darpa subterranean challenge. *Science Robotics*, 7(66):eabp9742, 2022.
 - [63] Eric Vollenweider, Marko Bjelonic, Victor Klemm, Nikita Rudin, Joonho Lee, and Marco Hutter. Advanced skills through multiple adversarial motion priors in reinforcement learning. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5120–5126, 2023. doi: 10.1109/ICRA48891.2023.10160751.
 - [64] Bowen Weng, Guillermo A Castillo, Yun-Seok Kang, and Ayonga Hereid. Towards standardized disturbance rejection testing of legged robot locomotion with linear impactor: A preliminary study, observations, and implications. *arXiv preprint arXiv:2308.14636*, 2023.
 - [65] Bowen Weng, Guillermo A Castillo, Wei Zhang, and Ayonga Hereid. On the comparability and optimal aggressiveness of the adversarial scenario-based safety testing of robots. *IEEE Transactions on Robotics*, 2023.
 - [66] Zhaoming Xie, Xingye Da, Michiel Van de Panne, Buck Babich, and Animesh Garg. Dynamics randomization revisited: A case study for quadrupedal locomotion. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4955–4961. IEEE, 2021.
 - [67] Chuanyu Yang, Kai Yuan, Qiuguo Zhu, Wanming Yu, and Zhibin Li. Multi-expert learning of adaptive legged locomotion. *Science Robotics*, 5(49):eabb2174, 2020.
 - [68] Chong Zhang and Lizhi Yang. Generating a terrain-robustness benchmark for legged locomotion: A prototype via terrain authoring and active learning. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*, pages 9190–9196. IEEE, 2023.
 - [69] Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric Xing, Laurent El Ghaoui, and Michael Jordan. Theoretically principled trade-off between robustness and accuracy. In *International conference on machine learning*, pages 7472–7482. PMLR, 2019.
 - [70] Shaofeng Zhang, Meng Liu, and Junchi Yan. The diversified ensemble neural network. *Advances in Neural Information Processing Systems*, 33:16001–16011, 2020.
 - [71] Fangwei Zhong, Peng Sun, Wenhan Luo, Tingyun Yan, and Yizhou Wang. Ad-vat+: An asymmetric dueling mechanism for learning and understanding visual active tracking. *IEEE transactions on pattern analysis and machine intelligence*, 43(5):1467–1482, 2019.
 - [72] Fangwei Zhong, Peng Sun, Wenhan Luo, Tingyun Yan, and Yizhou Wang. Towards distraction-robust active visual tracking. In *International Conference on Machine Learning*, pages 12782–12792. PMLR, 2021.