IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

MULE - Multi-Terrain and Unknown Load Adaptation for Effective Quadrupedal Locomotion

Vamshi Kumar Kurva and Shishir Kolathaya

***Abstract*—Quadrupedal robots deployed for load-carrying applications must maintain stable locomotion across diverse terrains and varying payloads. Traditional approaches like Model Predictive Control (MPC) can handle such variations but often rely on predefined gait schedules and manually tuned trajectory planners, limiting adaptability in unstructured environments. To address this, we propose an adaptive reinforcement learning (RL) framework that enables quadrupedal robots to respond dynamically to terrain and payload changes without relying on contact force measurements or gait designs. The controller consists of a *nominal policy* that learns general locomotion across terrains and an *adaptive policy* that outputs corrective actions for handling dynamic variations due to payloads. We validate our approach through extensive simulations in Isaac Gym across payloads (2–10 kg) and terrains including flat ground, slopes, and stairs. Our method achieves higher success rates and lower height-tracking errors while maintaining the Cost of Transport (CoT) comparable to the best-performing baselines and to no-load (NL) operation. Real-world deployment on a Unitree Go1 confirms the approach's effectiveness under both static and dynamic payload changes, including freely moving masses. The policy also performs well on outdoor terrains such as grass, soil, and staircases. The adaptive policy modulates corrections based on payload changes, improving body stability and tracking without post-deployment fine-tuning.**

***Index Terms*—Reinforcement learning, legged robots, adaptive control.**

I. Introduction

THE load-carrying capability of quadrupedal robots is essential for enhancing their deployment across various domains, including logistics, search and rescue, military operations, and agriculture. Enabling these robots to transport substantial payloads can significantly improve operational efficiency, reducing the need for human intervention in hazardous or hard-to-reach environments. Although substantial advances have been made in legged locomotion, particularly in traversing uneven terrain and handling external disturbances, the challenge of adapting to unknown payloads remains relatively less explored.

Received 17 July 2025; accepted 22 October 2025. Date of publication 12 November 2025; date of current version 24 November 2025. This article was recommended for publication by Associate Editor T. Kiyokawa and Editor O. Stasse upon evaluation of the reviewers' comments. This work was supported by **ARTPARK**. *(Corresponding author: Vamshi Kumar Kurva.)*

Vamshi Kumar Kurva is with the Department of Computer Science and Automation, Indian Institute of Science, Bengaluru 560012, India (e-mail: vamshi@iisc.ac.in).

Shishir Kolathaya is with the Centre for Cyber-Physical Systems and the Department of Computer Science and Automation, Indian Institute of Science, Bengaluru 560012, India (e-mail: shishir@iisc.ac.in).

Project page with additional results, and documentation is available at https://www.stochlab.com/MULE/.

Digital Object Identifier 10.1109/LRA.2025.3632081

Several studies have attempted to address this issue. [1] employs an online recursive method that estimates the robot's inertial parameters and base center of mass (CoM) using contact forces and joint angles. However, this approach requires the robot to halt during payload detection, restricting its suitability for real-time applications. [2] circumvents direct parameter identification by learning a locally linear, time-varying residual model around the current trajectory. This approach enables real-time control and demonstrates effective payload handling with a 10 kg load on a 12 kg A1 robot. However, the experiments are limited to flat ground with a centrally placed payload, resulting in minimal CoM shift and reduced practical applicability.

To improve robustness and adaptability under more realistic conditions, other works have explored adaptive and robust control strategies.[3] integrates $\mathcal{L}_1$ adaptive control into a force control framework, enabling a Unitree A1 quadruped to stably transport a 6kg payload. Similarly, [4] introduces a robust min-max MPC strategy based on robust optimization to account for system uncertainties. [5] incorporates Control Lyapunov Function (CLF) constraints within an MPC framework to ensure stable and adaptive locomotion, with validation conducted on the ANYmal robot. Meanwhile, [6] integrates RL with MPC to achieve adaptive balancing and swing foot reflection, allowing quadrupedal robots to dynamically adjust to payload variations and external disturbances. Their framework successfully demonstrated payload handling of 7 kg on a Unitree Go1 robot on flat ground.

All the above-mentioned methods are model-based force controllers that regulate ground reaction forces (GRFs) at the stance feet based on desired height and payload variations. Also, all the above methods showed payload adaptation mostly on the flat terrain or smooth sloped terrains. These approaches typically model quadrupeds as a single rigid body (SRB) and compute the optimal GRFs to be applied at contact points, while a low-level PD controller tracks the swing leg trajectories. To enforce structured locomotion, they rely on gait or trajectory generators to predefine foot contact schedules based on gait and velocity, enforcing distinct control strategies for swing and stance legs. A switch-based controller applies force control to stance legs and PD control to swing legs, making the system sensitive to early or delayed contacts on unstructured terrains, which can induce instability. In contrast, RL-based approaches have demonstrated effective locomotion across unstructured terrains without relying on predefined gait schedules [7], [8], [9], [10], [11], [12]. These methods directly output desired joint positions, which are tracked using a PD controller, eliminating the need for phase-based switching. By learning policies that implicitly adapt to terrain variations and contact conditions, RL-based controllers achieve more robust and versatile locomotion compared to model-based methods.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

Fig. 1. Hardware deployment of our framework across diverse terrains, including dynamic payload variations on flat ground, slopes, and uneven surfaces, as well as moving payloads (e.g., rolling balls) on stairs. All results are obtained using a single policy, without sim-to-real changes or real-world fine-tuning.

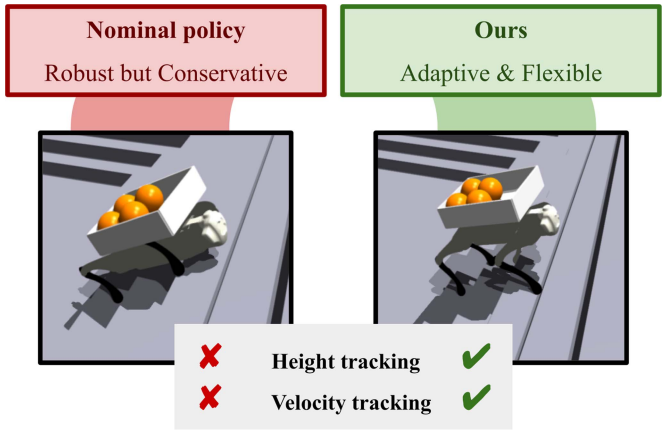

Fig. 2. Nominal policy vs. Adaptive framework: While nominal policy is robust but conservative, our adaptive approach enables flexible locomotion with improved height and velocity tracking across challenging terrains.

Building upon the strengths of RL-based methods for unstructured terrain locomotion, we propose an Adaptive RL framework (Fig. 2) that enables load-carrying capability across a variety of terrains. Unlike traditional model-based methods that rely on explicit model adjustments for varying loads, our approach allows the quadruped to dynamically adjust its locomotion strategy based on perceived changes in payload. This eliminates the reliance on predefined gait schedules, offering greater robustness and versatility in handling both terrain and payload variations. To this end, our key contributions are as follows:

- We introduce an **Adaptive RL framework** for locomotion under varying payload by augmenting a nominal policy with an adaptive corrective policy.
- We conduct extensive ablation studies comparing our method with baseline variants on flat ground and stairs under multiple payload configurations.
- We demonstrate that this adaptive framework significantly improves success rate, especially on stairs with added payloads, while maintaining energy efficiency comparable to the best performing baseline and to no-load operation.
- The proposed framework is validated in both simulation and hardware (Fig. 1), showing notable improvements compared to the baseline approach.

II. Background

A. Preliminaries

Reinforcement Learning (RL) provides a framework for training autonomous agents to maximize cumulative rewards in an environment by interacting with it through trial and error. In the context of quadruped locomotion, RL formulates the control problem as a Markov Decision Process (MDP), where the robot learns an optimal policy to achieve stable and efficient movement across diverse terrains. The MDP is defined by a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, r, \gamma)$, where: $\mathcal{S}$ represents the state space, $\mathcal{A}$ defines the action space, $\mathcal{P}$ denotes the transition dynamics, which model how the quadruped's state evolves based on applied actions and environmental interactions, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function and $\gamma \in (0, 1]$ is the discount factor that balances immediate and long-term rewards.

The goal of RL is to learn a policy $\pi_\theta(a_t \mid s_t)$, parameterized by θ, which defines the probability of selecting action $a_t \in \mathcal{A}$ given the current state $s_t \in \mathcal{S}$. The agent interacts with the environment over discrete time steps $t = 0, 1, 2, \ldots$, receiving a reward $r(s_t, a_t)$ at each step based on the current state and action. The objective of RL is to learn a policy that maximizes the expected cumulative discounted reward:

$$J(\pi) = \lim_{T \to \infty} \mathbb{E}\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)\right] \tag{1}$$

B. RL in Quadrupedal Locomotion

Recent advances in quadrupedal locomotion have leveraged high-throughput simulators like Isaac Gym to enable scalable RL. A seminal framework from ETH Zurich [10] introduced a teacher-student architecture, where a privileged-information-based teacher policy trains a student to infer latent dynamics from observation history, facilitating hardware deployment. Building on this, RMA [7] proposed real-time adaptation without domain knowledge or reference trajectories, enabling robust deployment across terrain variations. Subsequent work, Walk These Ways [8], demonstrated behavior diversity through reward shaping within a single policy. DreamWaQ [9] further extended these ideas using an asymmetric actor-critic framework, where the actor receives only partial observations while the critic accesses full state information during training. This setup allows the policy to benefit from privileged signals to learn more robust locomotion strategies while remaining deployable under limited sensory input. DreamWaQ demonstrated generalizable performance across diverse terrains, including stairs and slopes, highlighting the strength of asymmetry and structured information flow during training.

To further enhance robustness across real-world variations, such methods often incorporate domain randomization, which introduces small variations in robot parameters during training. This helps develop policies that are resilient to disturbances. However, when the range of parameter variations is too large, the resulting policies tend to be overly conservative [13], [14], prioritizing robustness at the cost of optimal performance. These limitations highlight the need for adaptive policies that can dynamically adjust to varying conditions, such as payload changes, rather than relying on a one-size-fits-all approach.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

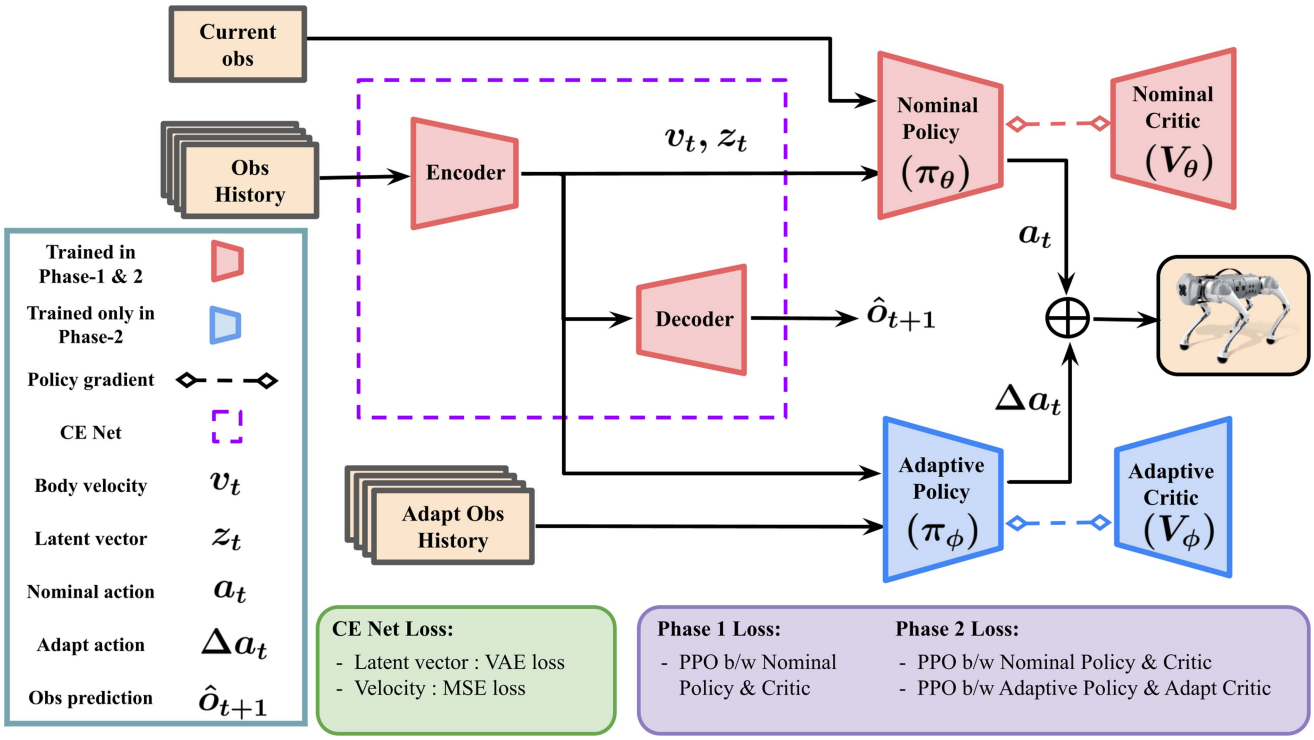

Fig. 3. Overview of the proposed framework - History of observations is encoded to get a latent vector and body velocity using Context-aided Estimator Network (CE Net). The nominal policy and critic are trained in Phase 1, while the adaptive policy and adaptive critic are introduced in Phase 2 to enhance adaptation to payload variations. The combined action enables robust locomotion across a wide range of payloads.

III. Methodology

When a payload is added or removed from the robot, it causes significant changes in system parameters such as mass, center of mass (CoM), and inertia, altering the dynamics of the system. Explicitly estimating these parameters in real time can be challenging and error prone. Instead, we propose an adaptive framework where a corrective action is learned to compensate for these changes.

Our proposed **Adaptive RL framework** (Fig. 3) is inspired by adaptive control methods from classical control and recent RL works [15], [16], [17], [18]. It consists of two phases of training:

- *Phase 1:* We train a nominal policy under normal conditions (without payload). The nominal policy is responsible for basic locomotion and command tracking.
- *Phase 2:* We train an adaptive policy to provide corrective actions under payload changes, treating the added payload as an external disturbance.

We denote the **Nominal Policy** as π_θ, which maps the observation o_t to action a_t, and the **Adaptive Policy** as π_ϕ, which takes a history of adaptive observations, denoted $\tilde{o}_t$, and outputs a corrective action Δa_t.

A. Phase 1: Nominal Policy Training Under No Load

The objective of Phase 1 is to train the nominal policy for robust locomotion across diverse terrains.

Observations - The observation is a 45 dimensional vector consisting of the following

$$o_t = \begin{bmatrix} \omega_t & g_t & c_t & q_t & \dot{q}_t & a_{t-1} \end{bmatrix}^T$$

where w_t, g_t are the body angular velocity and gravity vectors, c_t is the body velocity commands, $(q_t, \dot{q}_t)$ are the joint angle positions and velocities, a_{t-1} is the previous action.

Actions - The action is 12 dimensional vector which represents the desired joint positions relative to a fixed standing pose q_{stand}, i.e.

$$q_{des} = q_{stand} + a_t$$

The desired joint angles are tracked using a PD controller.

Rewards - Total reward at any given time-step is given by

$$r_{nominal} = \sum_i r_i w_i$$

where i is the index of the reward component and w_i is the weight as shown in Table I

Encoder - The encoder is a part of the Context Estimator Network (CE Net) that encodes the history of observations o_t^H into a latent vector z_t and body velocity v_t. A decoder is used to reconstruct the next observations from the encoding. β-VAE is used for this auto-encoding task. CE Net is optimized using a hybrid loss function, defined as follows:

$$\mathcal{L}_{CE} = \mathcal{L}_{est} + \mathcal{L}_{VAE} \tag{2}$$

These losses are taken directly from [9].

Training - We train only the nominal policy π_θ using Proximal Policy Optimization (PPO) [19], while the adaptive policy π_ϕ remains inactive (i.e., $\nabla_\phi = 0$). The final action applied to the environment is a_t. The PPO objective for the nominal policy is:

$$\mathcal{L}^\theta_{\text{PPO}} = \mathbb{E}\left[\min\left(\rho_t(\theta)\hat{A}^\theta_t,\ \text{clip}\left(\rho_t(\theta), 1-\epsilon, 1+\epsilon\right)\hat{A}^\theta_t\right)\right],$$

TABLE I
REWARD WEIGHTS FOR NOMINAL AND ADAPTIVE POLICIES: THE NOMINAL POLICY PRIORITIZES VELOCITY TRACKING, WHILE THE ADAPTIVE POLICY FOCUSES ON GRF TRACKING AND BODY HEIGHT STABILIZATION

Reward	**Nominal weights** (w_i)	**Adaptive weights** (α_i)
Linear velocity tracking	1.0	0.0
Angular velocity tracking	0.5	0.0
Linear velocity (z)	-2.0	-2.0
Angular velocity (xy)	-0.05	-0.05
Orientation	-0.2	-0.2
Joint accelerations	-2.5×10^{-7}	-2.5×10^{-7}
Joint power	-2.0×10^{-5}	0.0
Base height	-1.0	-2.0
Foot clearance	-0.01	-0.01
Action rate	-0.001	-0.01
Smoothness	-0.01	-0.01
GRF tracking	0.0	2.0

where $\rho_t(\theta)$ is the probability ratio between the current and old policies:

$$\rho_t(\theta) = \frac{\pi_\theta(a_t \mid o_t)}{\pi_{\theta_{\text{old}}}(a_t \mid o_t)},$$

and $\hat{A}_t^\theta$ is the advantage estimate for the nominal policy.

B. Phase 2: Policy Adaptation Under Varying Loads

In Phase 2, while the nominal policy retains the reward structure from Phase 1, the adaptive policy is trained with a separate reward that emphasizes stability, base height maintenance, and responsiveness to payload variations.

The primary objective of the adaptive policy is to maintain the robot's desired base height under varying payloads. When the base height drops below the target due to an increase in payload, the policy needs to apply greater forces at the stance feet to restore it. Estimating end-effector forces (foot forces) is crucial for achieving this corrective behavior. We estimate these forces at each foot using the Jacobian relationship between the applied joint torques and the resulting forces:

$$\tau = J(q)^T f \implies f = \left(J(q)^T\right)^\dagger \tau \tag{3}$$

where J is the Jacobian matrix that depends on the joint configuration q, τ is the vector of applied joint torques, f represents the estimated end-effector forces, and $(J(q)^T)^\dagger$ is the Moore-Penrose pseudoinverse of $J(q)^T$.

To incorporate this information into the adaptive policy, we augment its observation space with the estimated foot forces:

$$\text{Adapt observation } \tilde{o}_t = (o_t, f) \tag{4}$$

We introduce a GRF based height tracking reward to encourage the adaptive policy to generate higher GRFs when the base height falls below the desired target.

$$r_{\text{GRF}} = 0.75 \cdot \mathbb{1}\{h > h_{\text{cmd}}\} + 0.50 \cdot \mathbb{1}\{h < h_{\text{cmd}}\} \cdot \mathbb{1}\left\{\sum_{i=1}^{4} |f_i| > (m_r + m_p)g\right\}$$

where h is the current base height, h_{cmd} is the desired base height, f_i is the GRF at leg i, and m_r and m_p are the masses of the robot and payload, respectively. To encourage coordination between the two policies and keep the adaptive actions small, we modify the adaptation reward as follows:

$$r_{\text{adapt}} += 0.1 \cdot r_{\text{nominal}} \tag{5}$$

$$r_{\text{adapt}} += 0.2 \cdot \exp\left(-\|\Delta a_t\|^2\right) \tag{6}$$

Training - We introduce dynamic payload variations to train the robot to adapt to changing payload. A lightweight tray (250 g) is mounted on the robot's base, and at the start of each episode, four spherical objects (balls) are placed inside. The initial mass of each ball is sampled from a uniform distribution $[0, 0.5]$ kg, resulting in a total payload of up to 2 kg. The mass of each ball is re-sampled every 4 seconds from a uniform distribution $[0, 2.5]$ kg.

In this phase, the final action applied to the environment is $a_t + \Delta a_t$. The PPO objective for each policy is given by:

$$\mathcal{L}_{\text{PPO}}^\psi = \mathbb{E}\left[\min\left(\rho_t(\psi)\hat{A}_t^\psi,\ \text{clip}\left(\rho_t(\psi), 1-\epsilon, 1+\epsilon\right)\hat{A}_t^\psi\right)\right],$$

where $\rho_t(\psi)$ is the probability ratio between the current and old policies ($\psi \in \{\theta, \phi\}$), and $\hat{A}_t^\psi$ is the corresponding advantage estimate. The gradient updates for Phase 2 are given by:

$$\theta \leftarrow \theta - \eta \nabla_\theta \mathcal{L}_{\text{PPO}}^\theta \tag{7}$$

$$\phi \leftarrow \phi - \eta \nabla_\phi \mathcal{L}_{\text{PPO}}^\phi \tag{8}$$

IV. RESULTS

A. Simulation

1) Implementation Details: We used the Isaac Gym simulator to validate our controller. The policy was trained using 4096 agents with a history size of $H = 5$ on a Nvidia RTX A6000 GPU. All actor and critic networks consist of three hidden layers with 512, 256, and 128 units, respectively. The encoder network has two hidden layers with 128 and 64 units, while the decoder network also contains two hidden layers with 64 and 128 units, respectively. During Phase 1, only the nominal policy was trained for 1000 iterations. In Phase 2, the nominal policy weights from Phase 1 were restored, and both policies were trained simultaneously for 2000 iterations. Joint angles in simulation were tracked using an actuator network pre-trained for the Unitree Go1, ensuring realistic actuator behavior [8], [20].

2) Baselines and Ablations: Since our proposed method builds upon and improves DreamwaQ, we compare its performance against several variants and ablations.

In the context of quadrupedal locomotion, domain randomization typically refers to randomizing robot parameters (e.g., link lengths, masses) at the start of each episode, fixed thereafter. In the context of payload variation, we consider two domain randomization schemes: **SP** (Static Payload), where the payload is sampled at initialization and held constant throughout the episode, and **DP** (Dynamic Payload), where the payload is resampled every 4 seconds. We evaluate the following configurations:

- *DWQ+SP:* DreamwaQ with static payload randomization in [0, 10] kg
- *HL+SP:* HimLoco [11] with static payload randomization
- *DWQ+DP:* DreamwaQ with dynamic payload variation

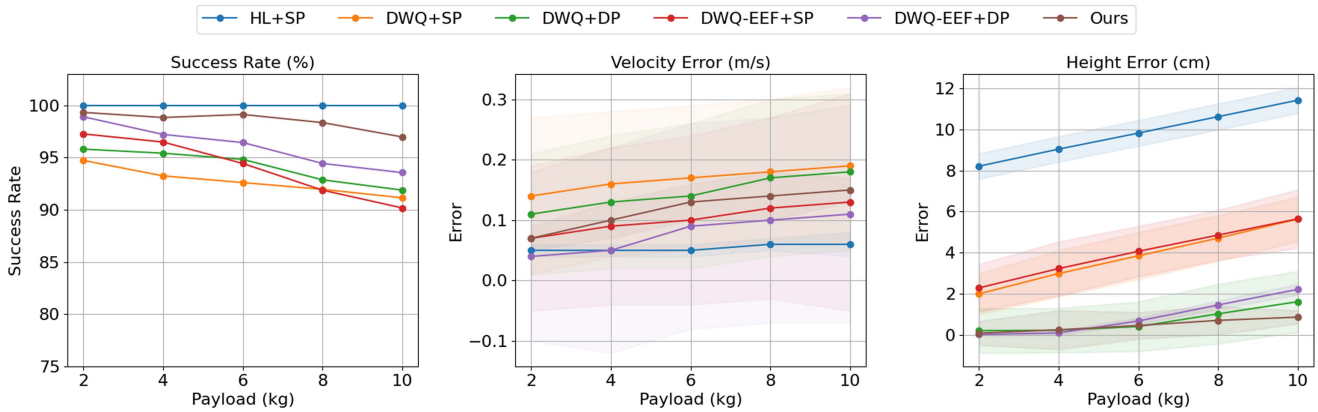

Fig. 4. Comparison of flat ground locomotion performance under different payloads. From left to right: success rate (%), velocity tracking error, and height tracking error.

- *DWQ-EEF+SP:* DreamwaQ with end-effector forces as inputs, and static payload randomization
- *DWQ-EEF+DP:* DreamwaQ with end-effector forces as inputs, GRF-tracking reward, and dynamic payload variation
- *Ours:* Our proposed approach (DreamWaQ augmented with an adaptive policy and evaluated under DP).

Each of the above methods is evaluated on flat ground and stairs under multiple payload configurations. For stair environments, we consider stair heights of 12 cm, 14 cm, and 16 cm, and evaluate performance under payloads of 2 kg, 4 kg, 6 kg, 8 kg, and 10 kg. All methods are evaluated using the same metrics across terrain and payload settings, with results aggregated over 1024 robots. Each episode runs for a maximum of 10 seconds (500 control time steps). We report three metrics: (i) **Success rate**, defined as the percentage of robots that traverse at least 75% of the target distance (computed as commanded velocity multiplied by episode duration), (ii) **Velocity tracking error**, measured as the mean and standard deviation of absolute velocity error per robot over its actual episode length, and (iii) **Height tracking error**, computed analogously. All metrics are computed per robot and averaged across the population.

Flat Terrain - On flat ground, all methods maintain high success rates across payloads, with a slight decline as payload increases. Our method achieves success rates that are consistently close to the best-performing baseline (HL+SP), with a margin of less than 3% across all payloads (see Fig. 4).

In terms of height tracking, our approach demonstrates strong performance, achieving the lowest height error at higher payloads (8 kg and 10 kg), and remaining among the top two methods at lower payloads (4 kg and 6 kg). This indicates that our method maintains stable body height, especially as payloads increase.

While HL+SP achieves the lowest velocity tracking errors (as low as 0.05–0.06 m/s), it also exhibits significantly higher height errors (8.21–11.42 cm). This shows that, on flat terrain, accurate forward velocity tracking can still be achieved even when body height control is poor. However, such behavior may not generalize to more complex scenarios like stairs or uneven slopes, where accurate height tracking becomes critical for success. In contrast, our method maintains low height errors across most settings while also keeping velocity tracking errors consistently low (within 0.07–0.15 m/s across all payloads). This demonstrates that our framework achieves a more balanced trade-off between height and velocity control.

Stairs - We observe that static-payload (SP) methods such as HL+SP, DWQ+SP, and DWQ-EEF+SP exhibit severe degradation in performance as either the stair height or payload mass increases. Their success rates decline sharply, often approaching zero for 10 kg payloads or 16 cm stairs (see Fig. 5). This drop occurs because, although payloads are randomized across episodes, each robot experiences a fixed payload within an episode, limiting the policy's ability to adapt to dynamic variations. Additionally, these methods exhibit significantly higher height tracking errors under large payloads, often exceeding 5–7 cm at 8–10 kg, which suggests a lowered base height, reducing the robot's ability to lift its legs and clear taller steps.

On the other hand, methods trained with dynamic payload switching, such as DWQ+DP and DWQ-EEF+DP, show more resilience to increasing difficulty. These policies benefit from intermittent mass variation during training, leading to better adaptation to changing payloads. This is reflected in consistently higher success rates and improved tracking metrics across moderate stair heights and payloads. Furthermore, the addition of end-effector forces (EEF variants) appears to offer further gains, particularly for taller stairs likely because it helps the policy infer aspects of the underlying dynamics such as payload variation. For instance, DWQ-EEF+DP generally outperforms DWQ+DP across payloads at 14 cm and 16 cm stairs.

Our method achieves the highest success rates across all stair heights and payloads, maintaining robust performance even in the most challenging settings. In terms of height tracking, it consistently yields the lowest error across all configurations, typically under 2 cm even as stair height and payload increase. While velocity tracking errors are also lower for our method in most settings, we do not emphasize this metric in our evaluation, since this metric can be misleading in low-success regimes. Policies that collapse early may still show small average velocity errors simply because they track well until failure, but do not sustain locomotion.

3) Analysis of Adaptation Policy: To understand how our approach achieves improved success rate, we analyze the role of the adaptive module. As shown in Fig. 6, increasing payloads lead the adaptive policy to produce higher corrective outputs, resulting in increased GRFs. This enables the robot to maintain target height and follow velocity commands, highlighting the

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

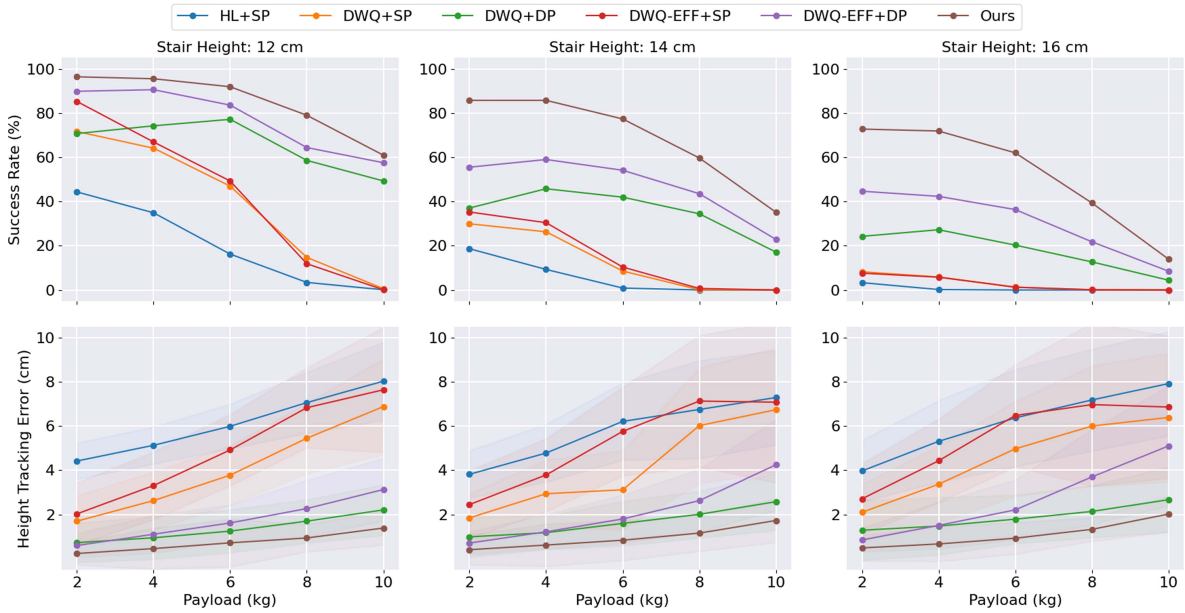

Fig. 5. Comparison of locomotion performance across stair heights (12 cm, 14 cm, and 16 cm) and payloads (2–10 kg). Top row shows the success rate (%) of different controllers as payload increases. Bottom row shows corresponding height tracking errors (cm), with shaded regions indicating standard deviation. Our method maintains the highest success rates and lowest height tracking errors across all payloads and stair configurations.

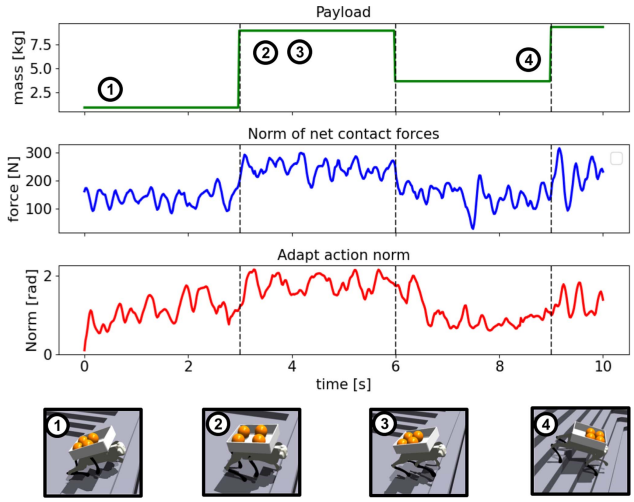

Fig. 6. Adaptation of the quadruped robot to varying payload on stairs. (Top) The payload mass profile with 4 phases indicating mass transitions. (Middle) Norm of net contact forces over time. (Bottom) Norm of adaptive actions, demonstrating the controller's response to mass changes and terrain transitions. Snapshots (1-4) depict representative instances during the locomotion sequence. (2) and (3) show how the robot has recovered from heavy payload change by generating higher GRFs.

adaptive policy's role in compensating for payload-induced dynamics. To quantify the contribution of the adaptive policy, we compare the full framework (nominal + adaptive) against the nominal policy alone, as shown in Fig. 7. The adaptive module consistently improves performance across all payloads and stair heights, with the gap widening under more challenging conditions (e.g., heavier payloads or steeper stairs). This indicates that the nominal policy alone is insufficient to generalize to significant dynamics shifts caused by varying payloads. In contrast, the adaptive policy enables the framework to compensate for such

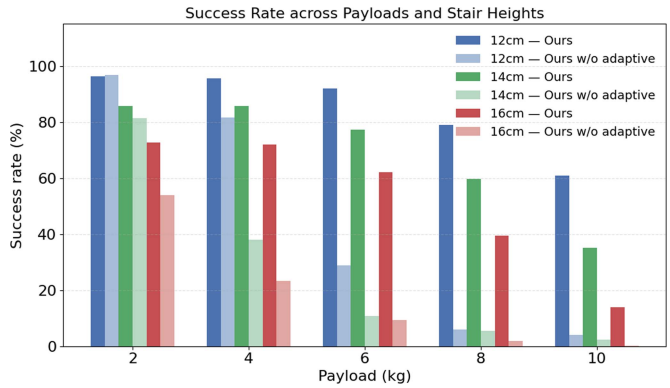

Fig. 7. Success rate comparison across payloads and stair heights for our framework with and without the adaptive policy. The darker bars correspond to the full framework (nominal + adaptive), while the lighter bars show performance using only the nominal policy without corrective adaptation.

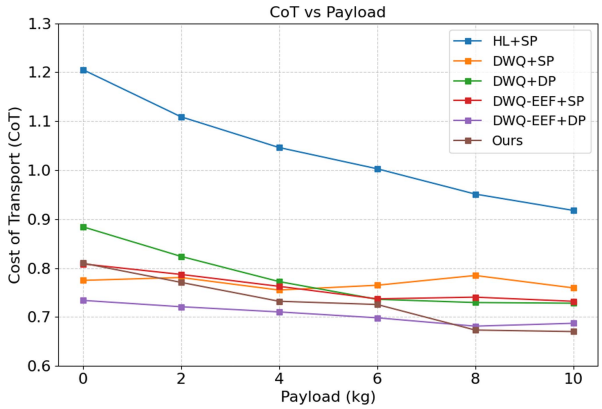

Fig. 8. Cost of transport (CoT) versus payload for all controllers. Our method maintains consistently low CoT across the tested payload range, showing energy efficiency comparable to the best-performing baselines.

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

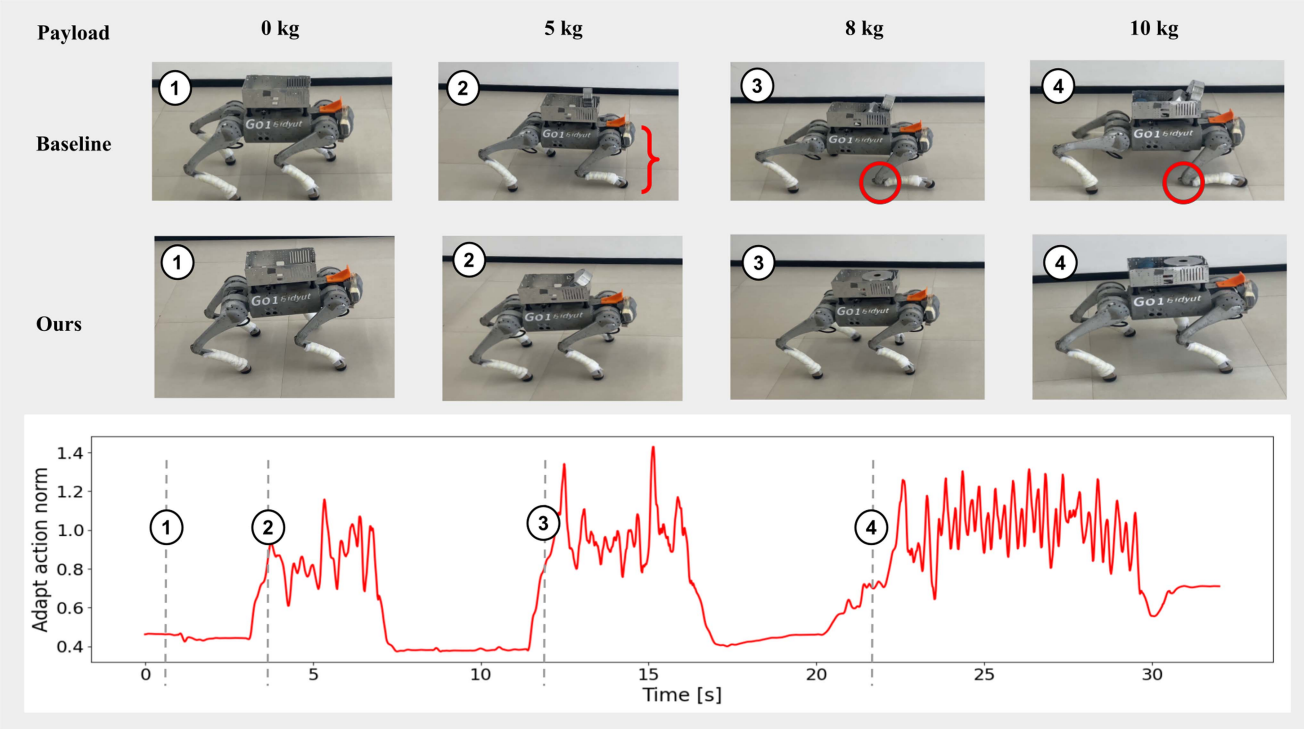

Fig. 9. Comparison of quadruped locomotion performance under progressively increasing payloads (0–10 kg) using the baseline controller (top row) and the proposed adaptive controller (middle row). The bottom plot shows the adapt action norm over time, capturing the adaptive controller's response to payload changes. Each numbered marker specifically denotes the moment additional payload was added to bring the total payload to the indicated value. Between two consecutive markers, the payload remains constant at the indicated value. The flat segments where the adapt action norm is close to zero correspond to brief halts where the robot was stopped to place or adjust the payload.

variations during deployment, leading to substantially higher success rates.

We attribute this success to the structure of our framework, which decomposes the problem into two complementary components. This division simplifies the learning problem compared to previous approaches, where a single policy must infer both terrain characteristics and dynamic shifts simultaneously. The clear drop in performance when the adaptive module is removed highlights its critical role in achieving robust locomotion in various scenarios. Next, we evaluate whether these performance gains affect energy efficiency.

4) Energy Efficiency: We assess the energy efficiency of the resulting locomotion using the *Cost of Transport* (CoT), a standard dimensionless measure that normalizes energy expenditure by weight and distance. Unlike the total mechanical work $\int_0^T \boldsymbol{\tau}^\top \dot{\mathbf{q}}\, dt$, which grows with episode duration, CoT captures the energy required to move a unit weight over a unit distance, enabling fair comparison across controllers and payloads. It is defined as

$$\mathrm{CoT} = \frac{\sum_{i=0}^{T} \langle |\boldsymbol{\tau}_i|,\ |\dot{\mathbf{q}}_i| \rangle \, \Delta t}{mgd}, \tag{9}$$

following the formulation in [21]. Here $\boldsymbol{\tau}_i$ and $\dot{\mathbf{q}}_i$ are the joint torque and velocity vectors at time i, Δt is the control timestep (0.02 s), m the total mass (robot plus payload), g the gravitational constant, and d the distance traveled until termination time T.

Fig. 8 shows the mean CoT over 1024 robots, each run for 500 control steps (10 s) on terrains of mixed difficulty (slopes of varying degrees and stairs of multiple heights). The downward trend in CoT with increasing payload suggests that mechanical work grows sublinearly with added mass, whereas mgd scales linearly, yielding a lower overall ratio. Controllers with higher CoT also exhibit larger body-height tracking errors (Figs. 4, 5), indicating that poor height regulation leads to inefficient, often crouched postures that demand higher joint torques and reduce swing clearance. The reduced swing height makes it harder to clear tall steps or uneven terrain, shortening the distance traveled and further increasing the CoT ratio. Our framework achieves the lowest CoT at high payloads while remaining competitive at lower payloads, consistent with its superior success rate and body-height tracking across the same terrain distribution. Thus, the adaptive framework delivers both robustness and energy efficiency across the range of terrain–payload combinations.

B. Hardware Deployment

Real-world experiments were conducted on a Unitree Go1 robot equipped with a 500 g stainless steel tray mounted on its base. To simulate dynamic payload variations, multiple 1 kg iron balls were placed in the tray, allowing them to shift freely and induce changes in the CoM during motion. Additionally, controlled static payload variations were tested by adding and removing 3 kg and 5 kg weight disks at different times. Joint angles commanded by the policy were tracked using a PD controller with gains $k_p = 20.0$ and $k_d = 0.5$. Fig. 9 compares the baseline and adaptive controllers as payload increases up to 10 kg. The baseline policy exhibits instability and foot scuffing at higher loads, whereas the adaptive controller maintains balance and coordination throughout. The bottom plot shows the norm

IEEE Robotics and Automation Letters (RA-L) paper, presented at ICRA 2026, Vienna, Austria. Cite as RA-L paper.

of the adaptive action over time, with distinct spikes aligning with each payload addition indicating the controller's response to changing dynamics.

Finally, we validated the controller's robustness on a range of outdoor environments, including 16 cm stairs, grass, soil, and uneven leafy terrain (Fig. 1). Despite real-world uncertainties, the trained controller successfully maintained stable locomotion under varying terrain and payloads, consistent with the simulation results.

V. Conclusion

We presented an RL-based adaptive control framework for quadrupedal locomotion across varying terrains and payload configurations. The proposed approach introduces an adaptive policy that provides corrective actions to complement the nominal policy, improving overall performance. Both policies are trained to optimize their respective rewards, with some common objectives, such as stability, shared between them. This shared reward structure fosters implicit cooperation, where the adaptive policy complements rather than conflicts with the nominal policy, injecting only the minimal corrective actions necessary to adapt to unexpected disturbances without overriding the baseline behavior. Instead of explicitly modeling dynamics or learning latent variables, the adaptive policy directly modifies the actions, making it easier to interpret what the policy is learning and how it adapts to changes. This modular design allows the system to retain the nominal policy's learned behaviors under normal conditions while leveraging the adaptive policy's rapid response capability when deviations are detected.

The proposed Adaptive RL framework was successfully deployed on a Unitree Go1 robot and validated on flat ground, slopes, and stairs under a range of static and dynamic payloads. Although our controller outperforms the baseline and ablation variants, success rates decline on steeper terrains with heavier loads. Enhancing performance in such challenging settings and extending the framework beyond load adaptation remain promising directions for future work. Furthermore, exploring the possibility of training only the adaptive policy while keeping the nominal policy frozen could simplify the training process and enable a broader applicability. This formulation may generalize to other scenarios where the nominal policy must be preserved while integrating new sensory modalities, such as vision or terrain perception, thereby enabling incremental adaptation without altering the underlying nominal controller.

References

[1] G. Tournois, M. Focchi, A. Del Prete, R. Orsolino, D. G. Caldwell, and C. Semini, "Online payload identification for quadruped robots," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2017, pp. 4889–4896.

[2] Y. Sun et al., "Online learning of unknown dynamics for model-based controllers in legged locomotion," *IEEE Robot. Automat. Lett.*, vol. 6, no. 4, pp. 8442–8449, Oct. 2021.

[3] M. Sombolestan, Y. Chen, and Q. Nguyen, "Adaptive force-based control for legged robots," in *IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2021, pp. 7440–7447.

[4] S. Xu, L. Zhu, H.-T. Zhang, and C. P. Ho, "Robust convex model predictive control for quadruped locomotion under uncertainties," *IEEE Trans. Robot.*, vol. 39, no. 6, pp. 4837–4854, Dec. 2023.

[5] M. V. Minniti, R. Grandia, F. Farshidian, and M. Hutter, "Adaptive CLF-MPC with application to quadrupedal robots," *IEEE Robot. Automat. Lett.*, vol. 7, no. 1, pp. 565–572, Jan. 2022.

[6] Y. Chen and Q. Nguyen, "Learning agile locomotion and adaptive behaviors via RL-augmented MPC," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2024, pp. 11436–11442.

[7] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "RMA: Rapid motor adaptation for legged robots," *Robot.: Sci. Syst. XVII*, 2021.

[8] G. B. Margolis and P. Agrawal, "Walk these ways: Tuning robot control for generalization with multiplicity of behavior," in *Proc. Conf. Robot Learn.*, 2023, pp. 22–31.

[9] I. M. A. Nahrendra, B. Yu, and H. Myung, "Dreamwaq: Learning robust quadrupedal locomotion with implicit terrain imagination via deep reinforcement learning," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2023, pp. 5078–5084.

[10] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Proc. Conf. Robot Learn.*, PMLR, 2022, pp. 91–100.

[11] J. Long, Z. Wang, Q. Li, L. Cao, J. Gao, and J. Pang, "Hybrid internal model: Learning agile legged locomotion with simulated robot response," in *Proc. 12th Int. Conf. Learn. Representations*, 2024.

[12] A. Shirwatkar, N. Saxena, K. Chandra, and S. Kolathaya, "Pip-loco: A proprioceptive infinite horizon planning framework for quadrupedal robot locomotion," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2025, pp. 11198–11204.

[13] G. Tiboni et al., "Domain randomization via entropy maximization," in *Proc. Int. Conf. Learn. Representations*, 2024.

[14] Y.-H. Lien, P.-C. Hsieh, and Y.-S. Wang, "Revisiting domain randomization via relaxed state-adversarial policy optimization," in *Proc. 40th Int. Conf. Mach. Learn.* in ser. Proceedings of Machine Learning Research, A. Krause, E. Brunskill, K. Cho, B. Engelhardt, S. Sabato, and J. Scarlett, Eds., Jul. 23–29, 2023, vol. 202, pp. 20939–20949.

[15] M. Sung, S. H. Karumanchi, A. Gahlawat, and N. Hovakimyan, "Robust model based reinforcement learning using $\mathcal{L}_1$ adaptive control," in *Proc. 12th Int. Conf. Learn. Representations*, 2024.

[16] Z. Li, C. Hu, Y. Wang, Y. Yang, and S. E. Li, "Safe reinforcement learning with dual robustness," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 12, pp. 10876–10890, Dec. 2024.

[17] S. Gangapurwala, M. Geisert, R. Orsolino, M. Fallon, and I. Havoutis, "RLOC: Terrain-aware legged locomotion using reinforcement learning and optimal control," *IEEE Trans. Robot.*, vol. 38, no. 5, pp. 2908–2927, Oct. 2022.

[18] H. Liu, Y. Cheng, R. Li, X. Hu, L. Ye, and H. Liu, "MBC: Multi-brain collaborative control for quadruped robots," in *Proc. 8th Annu. Conf. Robot Learn.*, 2024, pp. 3688–3704.

[19] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," 2017, *arXiv:1707.06347*.

[20] J. Hwangbo et al., "Learning agile and dynamic motor skills for legged robots," *Sci. Robot.*, vol. 4, Jan. 2019, Art. no. eaau5872.

[21] Z. Zhang, G. Bellegarda, M. Shafiee, and A. Ijspeert, "Online optimization of central pattern generators for quadruped locomotion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2024, pp. 13547–13554.