2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Experience-Learning Inspired Two-Step Reward Method for Efficient Legged Locomotion Learning Towards Natural and Robust Gaits

Yinghui Li, Jinze Wu, Xin Liu, Weizhong Guo*, Yufei Xue

Abstract— **Legged robots excel in navigating complex terrains, yet learning natural and robust motions in such environments remains challenging. Inspired by animals' experience-based stepwise learning process, we propose a two-stage framework for legged robots to progressively learn naturally robust movements using a two-step reward method. Initially robots learn the fundamental gaits on flat terrains with gait-rewards and generating valuable motion data. Subsequently, leveraging learned motion experience, they adopt adversarial imitation learning to tackle challenging terrains with refined movements. Our method addresses the challenge of acquiring effective imitation data and facilitates the learning process under various gait parameters with ease. The effectiveness of this approach has been validated on both quadruped and hexapod robots, demonstrating naturally robust gaits in real-world applications.**

I. Introduction

The intersection of biology and robotics has been a fertile ground [1] and the roboticists aspire to learn from creatures to design robots and enable them to learn to move. Animal locomotion learning typically progresses from simple tasks, like standing or swing legs to more complex movements in varied terrains. However, in this progressive learning model, how previously motion experiences influence the learning of new complex movements, and the logic of this biological subconscious learning, remains unknown.

In current research, the reinforcement learning method is pivotal in the locomotion learning of legged robots. Some researchers [2] [3] [4] [5] focus solely on the environmental adaptability of robot movement without imposing specific gait constraints, often resulting in unnatural motion behaviors. To ensure natural robot movements, some researchers [6] [7] [8] [9] establish basic reference trajectories and employ learning methods to generate compensatory motion signals for constrained gait movement in complex environments. However, different terrains often require distinct reference trajectories [10], and these trajectories significantly restrict effective exploration by the robot. Beyond using reference trajectories, a substantial body of research [11] [12] also involves setting extensive gait reward functions to induce robots to learn constrained motions. Nevertheless, the existing manual setting of reward functions struggles to generate natural and robust movements in complex environments.

Imitation learning methods make good substitutes for these complex reward functions [13] [14], effectively allowing robots to learn a wide variety of behaviors that would be challenging to manually encode. By leveraging animal reference motion trajectories, legged robots [14] have demonstrated the capability to effectively generate complex dynamic motion gaits. However, these behavioral cloning methods typically requires large datasets to be effective due to cumulative errors caused by covariate shift. To address these limitations, adversarial imitation learning [15] incorporates Generative Adversarial Networks (GANs) into the imitation learning framework. This approach employs a discriminator to evaluate the similarity between the policy outputs and reference motions, enhancing the robustness of behavior cloning. Peng and others have proposed AMP [16] and ASE [17] as exemplary adversarial imitation learning methods, which combine adversarial imitation learning with auxiliary task objectives, enabling agents to imitate behaviors from large unstructured motion datasets while performing high-level tasks. However, for a variety of robots, especially those with unique designs, obtaining effective reference data and training them to perform efficiently in complex terrains remains a significant challenge.

In this paper, we introduce a novel bioinspired two-stage learning framework with two-step reward setting that leverages prior motion experiences from simple locomotion tasks, utilizing adversarial imitation learning method to effectively induce naturally robust motion behaviors in complex terrains. This method has been successfully applied to both quadruped and hexapod robots, allowing them to achieve natural and robust gaits in challenging environments. The main contributions are listed as follows:

1) We introduce a generalized two-stage learning framework with two-step reward that utilizes the prior motion experience of robots to facilitate efficient mastery of natural and robust locomotion in complex environments.
2) Specific rewards setting and training for different robots are developed, demonstrating efficient application and validation of the proposed methods.
3) Employing a Teacher-Student strategy, these learned methods are successfully implemented on real robots, showcasing their capability to execute natural and robust locomotion in various challenging environments.

II. Method

The aim of this study is to develop a locomotion controller for legged robots, designed to perform natural and robust movement even in complex environments. The overall methodology is illustrated in Fig. 1, with the algorithm applied to both quadruped and hexapod robots. Here we

All authors are with School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai, China.
*Corresponding author
This work was supported by the National Key Research and Development Plan(2021YFF0307901).

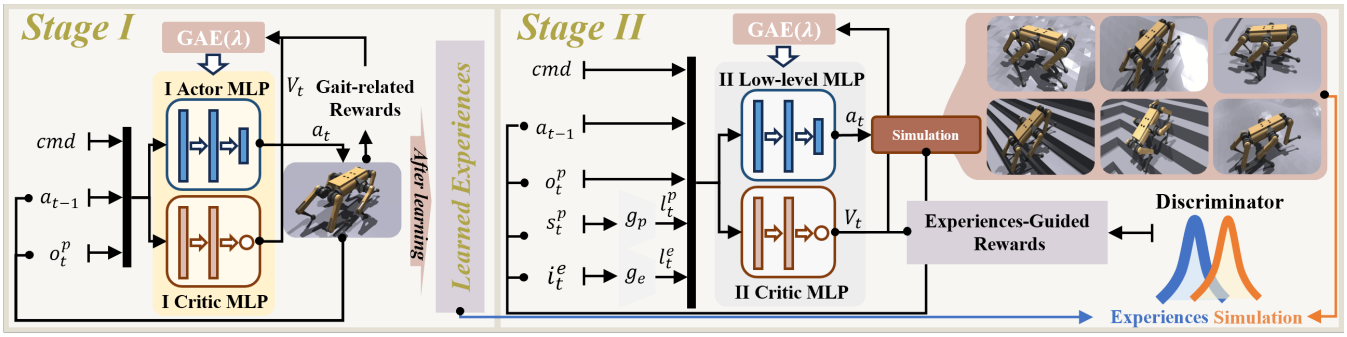

Fig. 1. Our approach involves a two-phase training process: first, inducing a natural and robust tripod gait on flat terrain using gait-related rewards for velocity tracking, and second, applying experiences from past motions to guide learning in complex environments. Initially, the robot learns to maintain tripod gait, foot trajectory, and body state through tailored reward functions. Post-training, it generates motion experiences tailored to specific tasks. In complex terrains, these experiences guide the training, with a discriminator network identifying task similarities and generating mimic rewards.

primarily focuses on the hexapod robot's natural and robust diagonal gait learning in such environments and the choice of hexapod robot is motivated by its high redundancy, which ensures stability in complex environments, even amidst partial motor failures, but introduces more exploration space that poses challenges in defining reward functions.

A. Reinforcement Learning Problem Formulation

The control issue adopts a discrete-time dynamic model. At each discrete interval, denoted by time step t, the system's state is completely characterized by $\boldsymbol{x}_t$. An action $\boldsymbol{a}_t$ is executed according to the policy, leading to a progression to the subsequent state $\boldsymbol{x}_{t+1}$, which occurs with a probability defined by $P\left(\boldsymbol{x}_{t+1} \mid \boldsymbol{x}_t, \boldsymbol{a}_t\right)$, and yields a reward r_t. The objective is to identify the optimal parameters π_θ that optimize the cumulative expected return, taking into account the decay of future rewards as expressed by the discount factor γ^t. This is represented as the maximization of:

$$J\left(\theta\right) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t r_t\right] \tag{1}$$

Observation Space: The observation spaces differ between stages due to task complexities. For flat terrain gait learning and velocity tracking, observations $\boldsymbol{x}_t^{\mathrm{I}}$ include proprioceptive data $\boldsymbol{o}_t^p$ (body angular velocity, gravity vector, joint positions, velocities), velocity commands, and prior action commands. In complex environments, observations $\boldsymbol{x}_t^{\mathrm{II}}$ expand to include terrain height scan $\boldsymbol{i}_t^e$ and privileged information $\boldsymbol{s}_t^p$ like body linear velocity and dynamic parameters (friction, contact forces, perturbations, collision states). For deployment training, the policy state, $\boldsymbol{x}_t^{\text{deploy}}$, is limited to proprioceptive observations $\boldsymbol{o}_t^p$ only.

Action Space: The policy action $\boldsymbol{a}_t$ is an 18-dimensional vector interpreted as a target joint position offset, which is added to the time-invariant nominal joint position to specify the target motor position for each joint. These position targets would be used to compute desired torques by low-level joint PD controllers $\tau = \boldsymbol{K}_p\left(\boldsymbol{q}_d - q\right) + \boldsymbol{K}_d\left(\dot{q}_d - \dot{q}\right)$, in which we determine the target joint velocity to 0.

Reward Design: Across different stages, there are several consistent reward function settings: a task-focused reward r_t^g and a regularization reward r_t^l. Additionally, the first stage focused on gait learning in flat plane, incorporates a specific gait reward function r_t^{gait}. In the second stage, which centers on experience-guided natural and robust movement learning, a distinct reward function r_t^e is implemented. These reward functions and their scales are listed in Table II-C.

B. Gait-Rewards Induced Learning for Simple Tasks

The primary task of the first stage is to enable a hexapod robot to perform natural tripod gait to tracking velocity commands. The hexapod's design, featuring 18 joints across six legs, introduces significant redundancy that can disrupt training, often leading the robot to neglect two legs. Even when a tripod gait is achieved, the leg trajectories might not be symmetrical. To address this, we designed gait-rewards functions inspired from [18] to effectively induce the robot to produce natural and robust velocity tracking movements.

Gait-Rewards: In this stage, apart from the task-related reward r_t^g, and the regularization reward r_t^l, we established four types of gait-rewards to regulate the gait. The phase tracking function utilizes the difference between foot forces and velocities and the ideal swing-support state to induce a tripod gait. The Raibert Heuristic function calculates the desired foot position on the ground plane, adjusting the baseline stance width in line with the desired contact schedule and body velocity. The foot-swing height tracking function first computes each foot's desired contact state based on phase and timing variables, then calculates a penalty function based on the target foot height difference to constrain foot motion.

Training: The Stage I policy $\pi_\theta^{\text{stageI}}$ comprises a actor network and a critic network, shown in table IV and is directly trained using the PPO algorithm [19]. During the training process, we utilize Generalized Advantage Estimation (GAE) to reduce variance and increase the stability of advantage estimates by blending rewards from multiple time steps.

Experiences Generation: After training, the controller enables the robot to execute simple tasks such as moving forwards and backwards, sidestepping, and turning with a regular diagonal gait. The actions performed during these tasks are captured and compiled into multiple 20-second trajectory experience datasets $\mathcal{D}$, which is then used for imitation learning in complex environments. Each state in dataset $\boldsymbol{s}_t^{AMP}$ in $\mathbb{R}^{42}$ includes joint positions, velocity,

base linear and angular velocities. State transitions from the dataset $\mathcal{D}$ are used as real samples to train the discriminator.

C. Experiences-Reward Induced Learning for Tough Tasks

In the second stage, where complex terrains may cause sudden gait changes, we address this challenge of effectively constraining movements through the Adversarial Motion Priors method, aiming to emulate biological progressive learning by drawing from previously accumulated motion experiences to generate more natural and robust movements. This method employs a GAN network to assess the similarity between current movements and reference experience trajectories, thereby generating a experiences-reward signal that ensures the robot's natural and robust gait.

Experiences-Reward: In this stage, the reward function is composed of three elements: a task-related reward r_t^g, a experience-guided reward r_t^e, and a regularization reward r_t^l, combined as $r_t = r_t^g + r_t^e + r_t^l$. The experience reward assesses how closely the agent's actions mirror those of the demonstrator, with higher rewards for greater similarity. Given the superior stability of the tripod gait for hexapods on uneven terrains, we employ a experience-guided reward based on adversarial motion priors to encourage our robot to adopt a tripod gait, mirroring behaviors from a reference experience dataset $\mathcal{D}$. Adopting the approach from [16], we introduce a discriminator D_φ, represented by a neural network with parameters φ, to discern whether a state transition $T_s = (\boldsymbol{s}_t, \boldsymbol{s}_{t+1})$ is an authentic sample from $\mathcal{D}$ or a fabricated sample by the policy π. The discriminator's training objective is defined as:

$$\begin{aligned} &\arg\min_{\varphi} \mathcal{L}_1 + \mathcal{L}_2 \\ &\mathcal{L}_1 = \mathbb{E}_{T_s\sim\mathcal{D}}\left[\left(D_\varphi(T_s)-1\right)^2\right] + \mathbb{E}_{T_s\sim\pi}\left[\left(D_\varphi(T_s)+1\right)^2\right] \\ &\mathcal{L}_2 = \frac{\alpha^{gp}}{2}{}_{T_s\sim\mathcal{D}}\left[\left\|\nabla_\varphi D_\varphi(T_s)\right\|_2\right], \end{aligned} \tag{2}$$

where the first loss function $\mathcal{L}_1$ uses a least square GAN formulation, focusing on reducing the Pearson divergence between the distribution of the agent's state transitions and that of the reference data. This aims to train the discriminator to effectively identify whether a state transition originates from the policy π or the reference experience dataset $\mathcal{D}$. Additionally, we incorporate a gradient penalty in the second loss term $\mathcal{L}_2$ in Eq. (2) to prevent the discriminator from assigning non-zero gradients to the real data samples' manifold. This penalty is vital for ensuring stable training and effective performance, as demonstrated in [16]. The coefficient α^{gp} is determined manually(we set α^{gp}=10). The tripod mimic reward is established based on:

$$r_t^s\left[T_s\sim\pi\right] = \max\left[0, 1-0.25\left(D_\varphi(T_s)-1\right)^2\right], \tag{3}$$

where the experience-reward is scaled to the range $[0, 1]$.

Curriculum Design: Training legged robots for blind locomotion on varied terrains involves significant challenges due to uncertain environmental interactions. Drawing on previous findings that diverse terrain training enhances complex locomotion skills, we introduce six types of procedurally generated terrains. Details of terrain types and their difficulty ranges are provided in Table III. Each terrain type is categorized into ten difficulty levels, with the rough slopes featuring added noise and the stairs having a consistent width. Given the initial instability of RL training, we employ a terrain curriculum [20], gradually introducing more complex terrains as the robot adapts to current levels, measured by its ability to maintain high linear velocity tracking rewards.

Domain Randomization To enhance our policy's robustness and ease its adaptation from simulations to real-world conditions, we vary several dynamics parameters in each episode which are outlined in Table II

Network architecture: The stage II policy $\pi_\theta^{\text{stageII}}$ contains three parts: a terrain encoder g_e, a privileged encoder g_p, and a low-level network. The terrain encoder compresses terrain information $\boldsymbol{i}_t^e \in \mathbb{R}^{187}$ into a 16-dimensional latent space, while the privileged encoder reduces the privileged state $\boldsymbol{s}_t^p \in \mathbb{R}^{42}$ to an 8-dimensional latent representation. These encodings, combined with proprioceptive observations $\boldsymbol{o}_t^p \in \mathbb{R}^{60}$, are processed by the low-level network with a $tanh$ output layer to produce actions. The discriminator D_φ is a simpler network with two hidden layers and a linear output. More details on each layer are shown in Table IV.

Training: We train the stage II policy using PPO [19] with access to privileged state and terrain information. Training of the policy and the discriminator occurs in synchronized. The policy generates state transitions $\boldsymbol{T}_s^{AMP} = \left(\boldsymbol{s}_t^{AMP}, \boldsymbol{s}_{t+1}^{AMP}\right)$ for the discriminator D_φ to evaluate $D_\varphi(T_s)$, contributing to the calculation of the mimic reward r_t^e. This stage's policy parameters θ are optimized for maximum return, while the discriminator's parameters φ are tuned to distinguish between real and generated transitions.

D. Deploy Training Based on Teacher-Student Methods

Due to the lack of exteroceptive sensory input in physical world, the terrains remain only partially observed, rendering the blind locomotion scenario a Partially Observable Markov Decision Process (POMDP).To realize the deployment of trained agent in the real world, we utilize a method known as privileged learning, as explored by [21]. The 'teacher' policy, referring to the stage II policy, is distilled through supervised learning into a 'student' policy. This 'student' policy is trained to infer dynamic characteristics from a sequence of past observations, effectively embodying the knowledge of II policy.

Network architecture: The student policy is built with a memory encoder (one LSTM-based RNN) and an MLP, identical in structure to the teacher's low-level net. Here, proprioceptive observations o_t^p and previous states (h_{t-1}, c_{t-1}) are encoded by the RNN into intermediate states m_t, and then processed by a neural network g_m to produce the student's latent representation l_t^s. To accelerate training, we initialize the student's low-level net with the teacher's pretrained weights. More layer details are in Table IV.

Training: The student policy is trained to replicate the teacher's actions without privileged information $\boldsymbol{s}_t^p$ and $\boldsymbol{i}_t^e$.

TABLE I

REWARD TERMS FOR VELOCITY COMMANDS TRACKING TASK, REGULARIZATION (STABILITY, SMOOTHNESS, SAFETY), AND SPECIFIC STAGE.

Stage	Term		annotation	equation	scale
For Both	Task r^g		Linear velocity tracking	$\exp\left(-\|\mathbf{v}_{t,xy}^{\text{des}} - \mathbf{v}_{t,xy}\|_2/0.15\right)$	$1dt$
			Angular velocity tracking	$\exp\left(-\|\omega_{t,z}^{\text{des}} - \omega_{t,z}\|_2/0.15\right)$	$0.8dt$
	Regularization r^l	Stability	Linear velocity penalty	$-v_{t,z}^2$	$2dt$
			Angular velocity penalty	$-\|\boldsymbol{\omega}_{t,xy}\|_2$	$0.05dt$
			Body height penalty	$-\|\boldsymbol{h}_z - \boldsymbol{h}_z^{\text{des}}\|_2$	$0.2dt$
		Smoothness	Joint torque	$-\|\boldsymbol{\tau}\|_2$	$1e^{-5}dt$
			Joint acceleration	$-\|\ddot{\mathbf{q}}\|_2$	$2.5e^{-7}dt$
			Action rate	$-\|\mathbf{a}_{t-1} - \mathbf{a}_t\|_2$	$0.01dt$
		Safety	Collisions	$-n_{collision}$	$0.1dt$
			Joint torque limits	$-\|\max\left(\|\boldsymbol{\tau}_t\| - \boldsymbol{\tau}^{limit}, 0\right)\|_2$	$0.01dt$
			Joint velocity limits	$-\|\max\left(\|\dot{\boldsymbol{q}}_t\| - \dot{\boldsymbol{q}}^{limit}, 0\right)\|_2$	$0.1dt$
			Contact force penalty	$-\|\max\left(\|\mathbf{f}_t\| - \mathbf{f}^{limit}, 0\right)\|_2$	$0.02dt$
Stage I	Gait Rewards r^{gait}		Swing phase tracking(force)	$\sum_{\text{foot}}\left[1 - C_{\text{foot}}^{\text{cmd}}\left(\boldsymbol{\theta}^{\text{cmd}}, t\right)\right]\exp\left\{-\left\|\mathbf{f}^{\text{toot}}\right\|^2/\sigma_{cf}\right\}$	$4dt$
			Stance phase tracking(velocity)	$\sum_{\text{foot}}\left[C_{\text{foot}}^{\text{cmd}}\left(\theta^{\text{cmd}}, t\right)\right]\exp\left\{-\left\|\mathbf{v}_{xy}^{\text{foot}}\right\|^2/\sigma_{cv}\right\}$	$4dt$
			Raibert footswing tracking	$\left(\mathbf{p}_{x,y,\text{ foot}}^f - \mathbf{p}_{x,y,\text{ foot}}^f\left(\boldsymbol{s}_y^{\text{des}}\right)\right)^2$	$10dt$
			footswing height tracking	$\sum_{\text{foot}}\left(\boldsymbol{h}_{z\text{ foot}}^f - \boldsymbol{h}_z^{f,\text{des}}\right)^2 C_{\text{foot}}^{\text{des}}\left(\boldsymbol{\theta}^{\text{des}}, t\right)$	$2dt$
Stage II	Experiences Reward r^e		Score of discriminator	$\max\left[0, 1 - 0.25\left(d_t^{\text{score}} - 1\right)^2\right]$	$1dt$

TABLE II

DYNAMIC PARAMETERS AND THE RANGE OF THEIR RANDOMIZATION VALUES USED DURING TRAINING.

Parameters	Range[Min, Max]	Unit
Link Mass	[0.8, 1.2]×nominal value	Kg
Payload Mass	[0, 5]	Kg
Payload Position	[-0.1, 0.1]relative to base position	m
Ground Friction	[0.05, 2.75]	-
Motor Strength	[0.8, 1.2]	-
Joint K_p	[0.8, 1.2]×80	-
Joint K_d	[0.8, 1.2]×1	-
Joint Position	[0.5, 1.5]×nominal value	rad

TABLE III

TERRAIN TYPES AND THE RANGE OF THEIR LEVEL-PROPERTIES USED DURING TRAINING.

Types	Level-Properties	Range[Min, Max]	Unit
Slopes (rough/normal)	Slope inclination	[0, 25]	deg
Stairs (up/down)	Step Height	[0.05, 0.2]	m
Waves	Wave Amplitude	[0.2, 0.5]	m
Discrete Steps	h^{step}	[0.05, 0.15]	m

Training involves imitation and reconstruction losses, the former for action mimicry and the latter for replicating the teacher's latent representations.

III. EXPERIMENTAL SETUP

Simulation: In our training, we simultaneously engaged 4096 agents across 25,000 episodes:5000 episodes for the Stage I policy, 10,000 episodes for the staget II policy and 10,000 for the student policy, using the IsaacGym [20]. Each RL episode was capped at 1000 steps, equating to 20 seconds, with early termination possible upon meeting specific criteria. The policies operated at a control frequency of 50 Hz, with each simulation step representing 0.02 seconds. All training costs about 20 hours on a NVIDIA RTX 4090 GPU.

Hardware: Our hexapod robot, standing at a height of 30 cm and weighing 25.5 kg, features six legs: Right Front (RF), Right Middle (RM), Right Hind (RH), Left Hind (LH), Left Middle (LM), and Left Front (LF). To boost stability and minimize leg collisions, the middle legs on each side are extended outward by 13.7 cm. The robot is designed with 24 degrees of freedom, of which 18 are actuated, equipped with three motors per leg. Sensor-wise, it includes joint position encoders and an Inertial Measurement Unit (IMU). Our controller runs at 50 Hz on an onboard computer.

TABLE IV

NETWORK ARCHITECTURE FOR TWO STAGES' POLICY AND STUDENT POLICY. ALL NETWORKS USE ELU ACTIVATIONS FOR HIDDEN LAYERS.

Module	Inputs	Hidden Layers	Outputs
I Actor (MLP)	o_t^p	[128, 128, 64]	a_t
I Critic (MLP)	o_t^p	[128, 256, 128]	V_t
II Low-Level (MLP)	l_t, o_t^p	[256, 128, 64]	a_t
II Critic (MLP)	x_t	[512, 256, 128]	V_t
Memory (LSTM)	o_t^p, h_{t-1}, c_{t-1}	[256, 256, 256]	m_t
g_p (MLP)	s_t^p	[64, 32]	l_t^p
g_e (MLP)	i_t^e	[256, 128]	l_t^e
g_m (MLP)	m_t	[256, 128]	l_t^{student}
D_φ (MLP)	s_t^{AMP}, s_{t+1}^{AMP}	[1024, 512]	d_t^{score}

IV. RESULTS AND DISCUSSION

A. Ablation Study for Experiences-Reward

To validate the effectiveness of the proposed two-step reward shaping training framework, we conducted ablation experiments using a hexapod robot's velocity tracking task in complex environments. These experiments included: (a) using only basic task rewards r_t^g and regularization rewards r_t^l during training; (b) adding gait reward r_t^{gait} to the basic rewards; (c) incorporating experience-guided reward r_t^e atop basic rewards, with varying its coefficients to assess its impact on training. Throughout these experiments, all basic

reward coefficients remained constant, aligning with the weights shown in Table II-C. We compared the effectiveness of different reward functions by analyzing the trend curves of terrain difficulty changes under various reward settings within a terrain curriculum, shown in Fig 2. The increase in terrain difficulty for the robot's movement only occurs when it achieves significant rewards, indicating better performance. Therefore, comparing the trend curves of total terrain difficulty escalation across different reward functions can effectively reflect their ability to induce efficient motion sampling. Typically, a faster increase in terrain difficulty implies more effective reward function settings. The data shows that with basic reward functions alone, the robot can learn traversable motions to some extent. However, as observed in the following section, these motions are quite unnatural, a consequence of the large search space hindering the collection of effective actions. Introducing Experience-Guided rewards accelerates the growth of terrain difficulty, suggesting more effective learning of reasonable motions. Additionally, as the coefficient of this reward function increases, overall learning speed grows. However, excessively high coefficients can reduce learning ability, indicating that while experience rewards effectively guide effective motion learning, overly stringent imitation of movements learned on flat terrain can hinder adaptation to complex environments. Furthermore, using basic rewards combined with manually set gait rewards does not adapt well to terrain changes, limiting the robot's learning of movements and, consequently, the increase in terrain difficulty.

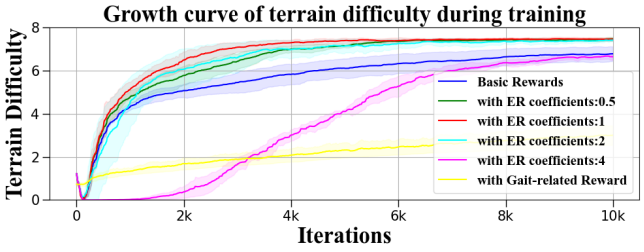

Fig. 2. The variation in terrain difficulty under different rewards indicates the robot's learning speed for effective motions. Each method was tested five times with the same random seeds. Basic rewards combined with well-scaled Experiences reward significantly improves learning in complex terrain.

B. Evaluation of the Natural and Robust Locomotion

After training, the most challenging 20cm staircase was used as a test site to verify the effectiveness of the experience reward function, with the velocity tracking performances and gait behaviors showcased in Fig. 3. The robot received various sine velocity commands (V_x, V_y, W_z) with different frequencies and amplitudes to assess its velocity tracking robustness and the naturalness of its gait. Training with added gait rewards failed to produce effective obstacle-crossing gaits. Hence, we didn't present its movement results. Basic task and regularization rewards generated gaits with velocity tracking capability, but these were unstable and irregular. As seen in Fig. 3 (a) (blue curve), the robot could track velocity commands but with a larger variance. Additionally, its z-direction speed, joint torques, and velocities were more

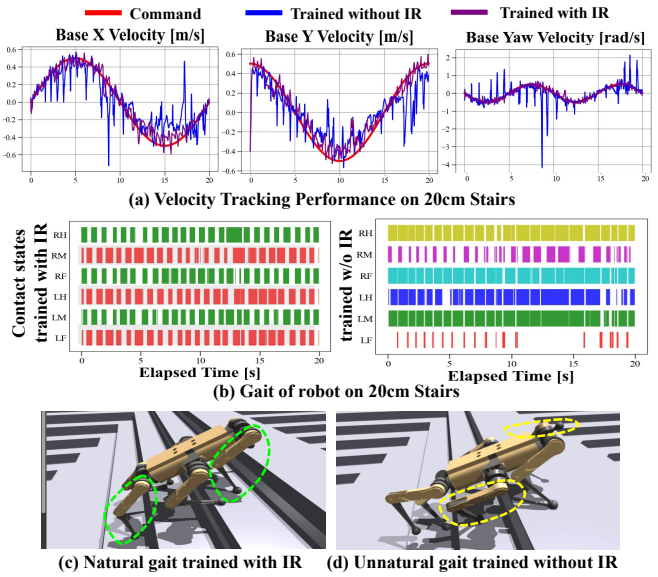

Fig. 3. Comparison of Velocity Tracking Performance and Gait on 20cm Stairs: Evaluating Robot Control with Policies Trained Using Experiences Rewards (ER) Versus Without.

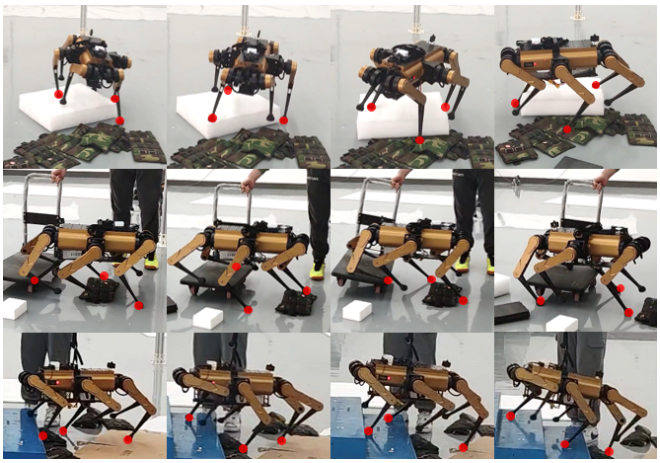

Fig. 4. The real hexapod robot, with its trained controller, achieves a natural tripod gait on soft foam, a sliding cart, and 10cm high stairs, maintaining robust operation even in scenarios of non-foot-end collisions with external objects. The red markings indicate contacts.

oscillatory. Foot contact forces, shown in the right chart of Fig. 3 (b), displayed an irregular gait with legs RH, RF, LH, LM contacting the ground for extended periods while RM and LF (yellow circle) barely touched the ground, leading to unnatural movements captured in Fig. 3 (d). In contrast, the inclusion of experience reward resulted in a natural and robust diagonal gait, as shown in Fig. 3 (a) (purple curve), where the robot tracked velocity commands with minimal error, even on 20cm high stairs. The robot's joint torques and velocities during movement were more stable without additional rewards. As seen in the left chart of Fig. 3 (b), legs LF, LH, RM moved in nearly identical phases, with the remaining diagonal legs similarly synchronized. The robot autonomously adjusted its step frequency to navigate complex terrains without breaking the tripod gait, as illustrated in Fig. 3 (c), where legs RF and RH (green labels) moved in almost identical states, crossing obstacles with a robust gait.

We successfully transferred our trained policy to the

TABLE V

SUCCESS RATES FOR DIFFERENT STEP HEIGHTS

Methods	5cm	10cm	15cm	20cm
MOB	90%	30%	0%	0%
Ours	100%	100%	80%	70%

physical hex robot using a teacher-student strategy, which led to natural, robust gaits in complex terrains, is shown in Fig 4. We compared our method with the MOB learning method [18] across staircases of varying heights, conducting ten trials at each height, and our method consistently achieved a higher success rate in climbing the stairs.

C. Validation of the Proposed Method's Universality

To demonstrate the universality of proposed method, we applied the method to the training of several quadruped robots, aiming to enable them to exhibit various locomotion gaits in complex environments. Utilizing the same approach, we initially induced different regular gaits such as trot, bound, and pace in a flat terrain through gait rewards.Further, guided by learning experiences and incentivized by experiential rewards, the robots were taught to adopt appropriate gaits for varying complex terrains, shown in Fig 5.

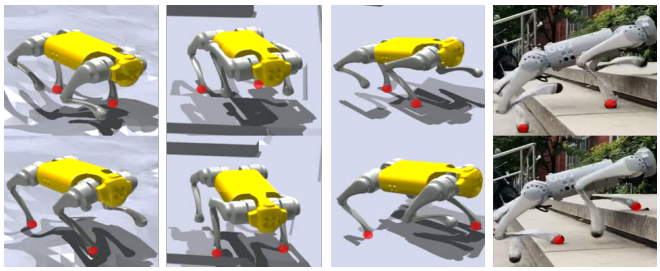

Fig. 5. The quadruped robot GO1, utilizing trained controller, achieves various gaits in complex terrains, displayed from left to right as pace, bound, and trot, along with the results of transferring the trot gait to a real robot.

V. CONCLUSIONS

In this study, we introduce a bioinspired two-stage learning framework with two-step reward settings, enabling diverse legged robots to master naturally robust movements in complex environments. Initially, manual adjustments of reward functions facilitate natural gait generation on flat terrain. Subsequently, biological learning principles are employed, using these initial gaits as baselines for more intricate task learning. This approach significantly reduces the need for extensive manual tuning and enhances learning efficiency. It offers robust universality, allowing similar methodologies to be applied across various robot types and tasks, thereby simplifying the development of natural and robust locomotion controllers. Future efforts will explore techniques for simultaneously satisfying multiple gait requirements and extending this method to a wider array of robotic platforms to further its applicability and versatility.

ACKNOWLEDGMENT

We extend our gratitude to professor Chenkun Qi for providing the Unitree-Go1 and professor Feng Gao for offering the hexapod robot.

REFERENCES

[1] P. Ramdya and A. J. Ijspeert, "The neuromechanics of animal locomotion: From biology to robotics and back," Science Robotics, vol. 8, no. 78, p. eadg0279, 2023.

[2] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "Rma: Rapid motor adaptation for legged robots," in Robotics: Science and Systems, 2021.

[3] I. M. A. Nahrendra, B. Yu, and H. Myung, "Dreamwaq: Learning robust quadrupedal locomotion with implicit terrain imagination via deep reinforcement learning," arXiv, Mar. 2023. [Online]. Available: http://arxiv.org/abs/2301.10602

[4] G. Ji, J. Mun, H. Kim, and J. Hwangbo, "Concurrent training of a control policy and a state estimator for dynamic and robust legged locomotion," IEEE Robotics and Automation Letters, vol. 7, no. 2, pp. 4630–4637, 2022.

[5] G. Margolis, G. Yang, K. Paigwar, T. Chen, and P. Agrawal, "Rapid locomotion via reinforcement learning," in Robotics: Science and Systems, 2022.

[6] J. Hwangbo, J. Lee, A. Dosovitskiy, D. Bellicoso, V. Tsounis, V. Koltun, and M. Hutter, "Learning agile and dynamic motor skills for legged robots," Science Robotics, vol. 4, no. 26, p. eaau5872, 2019.

[7] H. Shi, B. Zhou, H. Zeng, F. Wang, Y. Dong, J. Li, K. Wang, H. Tian, and M. Q.-H. Meng, "Reinforcement learning with evolutionary trajectory generator: A general approach for quadrupedal locomotion," IEEE Robotics and Automation Letters, vol. 7, no. 2, pp. 3085–3092, 2022.

[8] C. Yang, K. Yuan, Q. Zhu, W. Yu, and Z. Li, "Multi-expert learning of adaptive legged locomotion," Science Robotics, vol. 5, no. 49, p. eabb2174, 2020.

[9] M. Thor and P. Manoonpong, "Versatile modular neural locomotion control with fast learning," Nature Machine Intelligence, vol. 4, no. 2, pp. 169–179, 2022.

[10] H. Shi, "Reinforcement learning with evolutionary trajectory generator: A general approach for quadrupedal locomotion," 2021.

[11] H. Duan et al., "Sim-to-real learning of footstep-constrained bipedal dynamic walking," arXiv, May. 2022. [Online]. Available: http://arxiv.org/abs/2203.07589

[12] G. B. Margolis and P. Agrawal, "Walk these ways: Tuning robot control for generalization with multiplicity of behavior," arXiv, Dec. 2022. [Online]. Available: http://arxiv.org/abs/2212.03238

[13] X. B. Peng, P. Abbeel, S. Levine, and M. Van de Panne, "Deepmimic: Example-guided deep reinforcement learning of physics-based character skills," ACM Transactions on Graphics (TOG), vol. 37, no. 4, pp. 1–14, 2018.

[14] X. B. Peng, E. Coumans, T. Zhang, T.-W. E. Lee, J. Tan, and S. Levine, "Learning agile robotic locomotion skills by imitating animals," in Robotics: Science and Systems, 2020.

[15] J. Ho and S. Ermon, "Generative adversarial imitation learning," Advances in Neural Information Processing Systems, vol. 29, 2016.

[16] X. B. Peng, Z. Ma, P. Abbeel, S. Levine, and A. Kanazawa, "Amp: Adversarial motion priors for stylized physics-based character control," ACM Transactions on Graphics (TOG), vol. 40, no. 4, pp. 1–20, 2021.

[17] X. B. Peng, Y. Guo, L. Halper, S. Levine, and S. Fidler, "Ase: Large-scale reusable adversarial skill embeddings for physically simulated characters," ACM Transactions on Graphics (TOG), vol. 41, no. 4, pp. 1–17, 2022.

[18] G. B. Margolis and P. Agrawal, "Walk these ways: Tuning robot control for generalization with multiplicity of behavior," in Conference on Robot Learning, pp. 22–31. PMLR, 2023.

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

[20] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in 5th Annual Conference on Robot Learning, 2021.

[21] D. Chen, B. Zhou, V. Koltun, and P. Krähenbühl, "Learning by cheating," in Conference on Robot Learning, 2020.