2024 IEEE International Conference on Robotics and Automation (ICRA)
May 13-17, 2024. Yokohama, Japan

# Infer and Adapt: Bipedal Locomotion Reward Learning from Demonstrations via Inverse Reinforcement Learning

Feiyang Wu, Zhaoyuan Gu, Hanran Wu, Anqi Wu†, and Ye Zhao†

2024 IEEE International Conference on Robotics and Automation (ICRA) | 979-8-3503-8457-4/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICRA57147.2024.10611685

***Abstract*— Enabling bipedal walking robots to learn how to maneuver over highly uneven, dynamically changing terrains is challenging due to the complexity of robot dynamics and interacted environments. Recent advancements in learning from demonstrations have shown promising results for robot learning in complex environments. While imitation learning of expert policies has been well-explored, the study of learning expert reward functions is largely under-explored in legged locomotion. This paper brings state-of-the-art Inverse Reinforcement Learning (IRL) techniques to solving bipedal locomotion problems over complex terrains. We propose algorithms for learning expert reward functions, and we subsequently analyze the learned functions. Through nonlinear function approximation, we uncover meaningful insights into the expert's locomotion strategies. Furthermore, we empirically demonstrate that training a bipedal locomotion policy with the inferred reward functions enhances its walking performance on unseen terrains, highlighting the adaptability offered by reward learning.**

## I. Introduction

Humans exhibit a remarkable ability to achieve and generalize locomotion strategies from expert demonstrations. This inference ability enables the knowledge transfer from simple tasks to novel tasks and the efficient acquisition of new locomotion skills [1]–[4]. Despite this amazing ability inherent in the human brain, our understanding remains limited regarding the internal representation of a locomotion skill and more importantly, the mechanism for applying acquired skills to novel tasks. Inspired by human's ability to learn from expert demonstrations, this study takes an initial step to mimic this learning ability in the context of bipedal robot locomotion. Moreover, we seek the explainability of the learned skills and demonstrate their generalizability by subjecting the robot to maneuver over various rough terrains.

Imitation learning has been extensively explored as a methodology for learning from demonstration [5]–[8]. Although unable to infer the true intention behind the demonstrations, imitation learning often adopts Reinforcement Learning (RL) formulations to sidestep the problem of lacking an accurate reward function. This RL-based approach requires only designing a reward for tracking the demonstrated actions. The development of efficient RL algorithms facilitated a wide range of successful applications of imitation learning for agile bipedal locomotion, such as running [9], jumping [10], climbing stairs [11], playing soccer [12], carrying loads [13], and walking over diverse terrains [14]. However, a majority of these works still adopt handcrafted reward functions that heavily rely on domain knowledge and experience. Such reward functions are often tailored for specific environments and have a combination of specific features from the robot's state. Consequently, agents learned under such rewards often lack generalizability and struggle to adapt to new environments. Inverse Reinforcement Learning (IRL) [15], [16], on the other hand, subsumes the aforementioned imitation learning problem. IRL not only recovers the expert's policy but also the underlying reward function, which captures the essence of the expert's intention and enables adaptations of the robot's motion to unseen tasks. Therefore, IRL has gained considerable interest within the robotics community [17]–[20], with some studies employing IRL to gain a deep understanding of the reward function.

However, prior IRL works often presuppose a predetermined feature space and reward structure [19], [21]. This constrains the expressiveness of reward modeling and leads to limited performance in estimating the true reward functions. Furthermore, the existing robotics IRL works do not analyze the learned reward functions for further usage in practice such as adapting the learned reward for RL during challenging unseen tasks. It remains unclear how one can leverage and transfer the information learned from the reward functions in new environments. Moreover, computational complexity has been a hurdle for IRL methods to be widely adopted in the robotics learning community. Recent advances focus on accelerating algorithm efficiency of IRL [22]–[25].

In this paper, we develop a novel framework of reward learning, interpretation, and adaptation (Fig. 1) to address the aforementioned issues of the existing robotics IRL works. During the learning phase, we employ the Inverse Policy Mirror Descent (IPMD) method [25] to infer the reward from demonstrations. IPMD has been shown to be computationally efficient. It solves the IRL problem with a novel average-reward criterion under a Maximum Entropy framework [26], [27]. The Maximum Entropy framework can discern the most accurate reward estimation by guiding the policy search with the maximum entropy principle. The average-reward criterion also helps to accurately identify reward by dropping the discounted factor that is often used under the classic discounted-reward setting. Since demonstrations often lack an explicit discount factor, using a mismatching discounted factor from the ground truth will lead to drastically er-

F. Wu and A. Wu are with the School of Computational Science and Engineering, Georgia Institute of Technology, Atlanta, GA 30308, USA. {feiyangwu, anqiwu}@gatech.edu

Z. Gu, and Y. Zhao are with the Woodruff School of Mechanical Engineering, Georgia Institute of Technology, Atlanta, GA 30308, USA. {zgu78, yezhao}@gatech.edu

H. Wu is with the College of Computing, Georgia Institute of Technology, Atlanta, GA 30308, USA. hanran.wu@gatech.edu

† Co-senior authorship

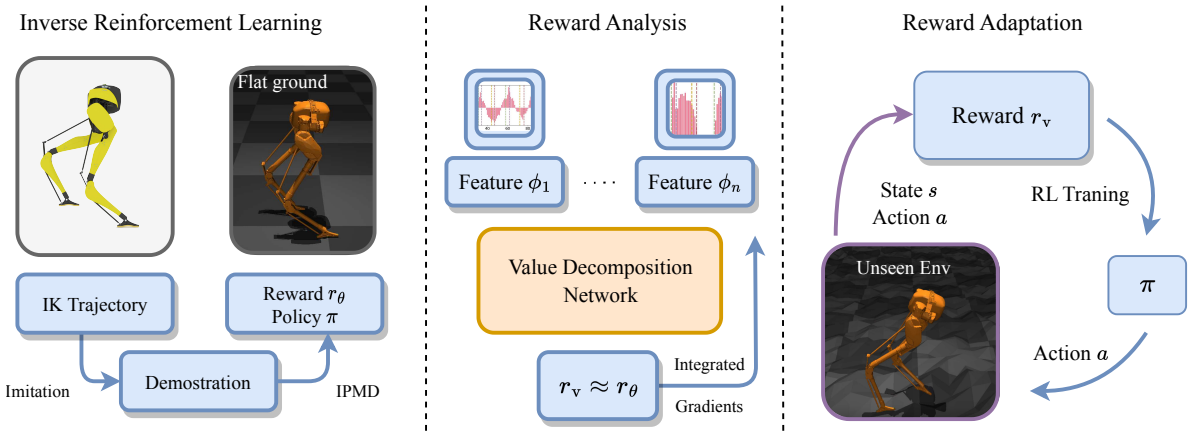



Fig. 1: In this work, we investigate the reward function learned by Inverse Reinforcement Learning algorithms. We propose a two-stage training algorithm for Cassie to learn reward functions and optimal policies from demonstrations. We then analyze the reward function learned from those demonstrations. The learned reward is further used to train RL agents in difficult environments.

roneous reward function estimations under the discounted setting [25]. Moreover, the average-reward criterion has been thoroughly investigated in the literature and has also been adopted in robotics learning tasks [28], [29], [29]–[33]. It has become a common practice for RL benchmarks to use an average-reward metric for evaluation, which further motivates the adoption of the average-reward criterion for solving locomotion tasks.

To gain an in-depth understanding of the learned reward, we employ a Value Decomposition Network (VDN) [34] and utilize Integrated Gradients (IG) [35] to obtain meaningful knowledge of locomotion features leading to high rewards. We will then incorporate such important features into reward design for locomotion in challenging unseen environments, which we refer to as reward adaptation. Note that it is not a new topic to adapt motor and locomotion skills learned from human demonstrations to robots [36]–[38] or from simulated environments to real-life environments [7], [21], [39]. However, these works require a sophisticated design and learning of policies or controllers to achieve robust adaptation. Instead, we investigate the possibility of adapting reward functions. Related methods in adapting reward [40]–[42] require crafting intricate, domain-specific reward functions and learning those reward functions under diverse environments to promote the robustness of the policy. In this work, we use IRL to learn a free-form reward function parametrized by a neural network with inputs directly from the robot's state and action space. We show that the learned reward functions contain transferable information about robot locomotion behaviors and verify such properties by training agents using the learned rewards in diverse challenging environments that are not previously seen. We observe a significant performance boost in walking speed and robustness by incorporating such information. To the best of our knowledge, we are the first to analyze and adapt free-form rewards in a principled way.

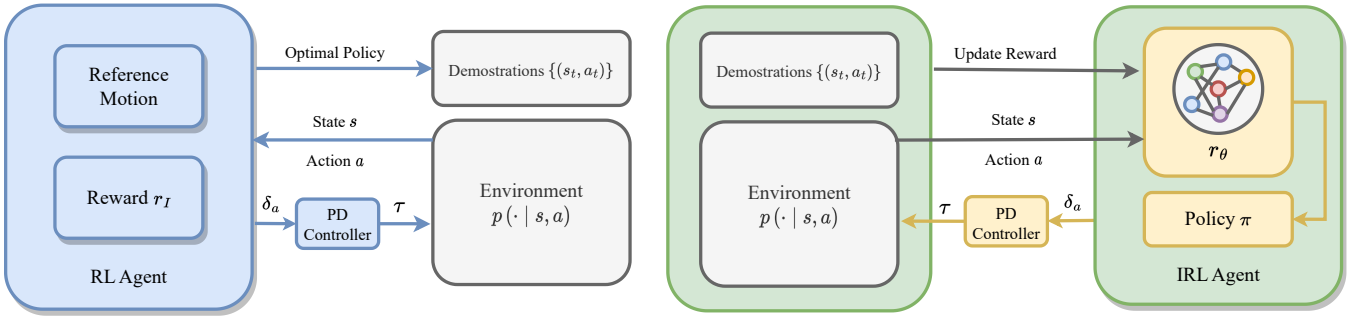
The salient contributions of our work are listed as follows:

- **Inverse Reinforcement Learning for Bipedal Locomotion**: We propose a two-stage IRL paradigm to address bipedal locomotion tasks via IPMD. In stage one, we obtain expert policies from a fully-body inverse kinematics function of Cassie. In the next stage, IPMD learns reward functions from the near-optimal demonstrations generated by the policies learned in the first stage. Our work is the first study that applies IRL to bipedal locomotion under the average-reward criterion.
- **Importance Analysis of Expert Reward Function**: We employ a Value Decomposition Network (VDN) to approximate the inferred locomotion reward function and Integrated Gradients (IG) to analyze the VDN for reward interpretation. By ensuring the monotonicity of the feature space, VDN enables the interpretation of the reward function with IG while preserving model expressiveness. We successfully perform a rigorous analysis of the importance of individual features, exposing components of the locomotion behavior that are crucial to its reward functions, thereby guiding the design of new rewards for new environments.
- **Reward Adaptation in Challenging Locomotion Environments**: We further verify that the learned reward from a flat terrain and the important features extracted from our reward analysis can be seamlessly adapted to novel, unseen terrains. Our empirical results substantiate that the inferred reward function encapsulates knowledge highly relevant to robotic motions that are generalizable across different terrain scenarios.

## II. Background

In this section, we introduce preliminaries for Average-reward Markov Decision Processes (AMDPs). An AMDP is formalized by a tuple $(\mathcal{S}, \mathcal{A}, \mathsf{P}, r)$, where $\mathcal{S}$ signifies the state space, $\mathcal{A}$ represents the action space, $\mathsf{P}$ denotes the transition probability, and $r$ is the reward function. At each time instance $t$, the agent selects an action $a \in \mathcal{A}$ from the current state $s \in \mathcal{S}$. The system then transitions to a subsequent state $s' \in \mathcal{S}$ based on the probability $\mathsf{P}(s'|s,a)$, while the agent accrues an instantaneous reward $r(s,a)$.

The primary objective of the agent is to establish a policy




Stage 1: Generate demonstrations via Imitation Learning

Stage 2: Learn reward functions and policies via IRL

Fig. 2: Our two-stage training pipeline. The blue box denotes the imitation learning part (first stage). The agent is then used to generate expert demonstrations, which are used by the second stage to update the reward and policy using Inverse Policy Mirror Descent.

$\pi : \mathcal{S} \to \mathcal{A}$ that optimizes the long-term average reward, mathematically given by

$$\rho^\pi(s) := \lim_{T\to\infty} \frac{1}{T}\mathbb{E}\left[\sum_{t=0}^{T-1} r(s_t, a_t)|s_0 = s\right]. \quad (1)$$

Given an expert demonstration set $\{(s_i, a_i)\}_{i\geq 1}$, IRL aims to extract a reward function that most accurately captures the behavior of the expert. Particularly, in this work, we adopt the Maximum Entropy Inverse Reinforcement Learning (MaxEnt-IRL) framework [26].

We denote $r_\theta$ as the estimation of the reward function, where $\theta$ is the parameter of the model of choice to represent the reward function $r(s_t, a_t)$ in Eq. (1). For example, $\theta$ can be the weights and biases in a neural network that parameterize the reward.

In this work, we adopt the environment designed in [43] with the robot's joint-space state as the state space: for any state $s = (x, \hat{x}) \in \mathcal{S}$, let $x = (q, \dot{q}) \in \mathbb{R}^{2N}$ represent the robot joint position and velocity, $N = 14$ be the number of joints of Cassie and $\hat{x} \in \mathbb{R}^{2N}$ represent the reference motions. Given a reference action $\hat{a}$ at a reference state $\hat{x}$, the policy outputs an augmentation term $\delta a$ that corrects the reference action, where $\hat{a}, \delta a \in \mathbb{R}^M, M = 10$. The result is a Proportional Derivative (PD) target, $a = \delta a + \hat{a}$, for a low-level PD controller, which generates a torque $\tau \in \mathbb{R}^M$ to track joint angles.

## III. Methods

In this section, we first introduce the pipeline that applies Inverse Policy Mirror Descent (IPMD) for bipedal locomotion to learn reward functions. We then outline our approach to analyze the learned reward function and methodology of conducting reward adaptation experiments.

### A. Two-Stage Learning Pipeline

Recent RL techniques for bipedal locomotion rely on carefully constructing the state and action space and designing sophisticated reward functions [43]–[45]. IRL models endow capabilities to learn from demonstrations. However, a practical challenge often arises: *what type of trajectory data should IRL leverage for effective learning?* Directly recording trajectories from robots such as motion capture approaches can be laborious and time-consuming, while data derived from model-based methods such as inverse kinematics or trajectory optimization often suffer from inaccurate models and unrealistic assumptions. To get high-quality demonstrations for effective IRL, we will use imitation learning with the Markov Decision Process (MDP) environment similar to [43], which can produce computationally convenient and dynamically accurate expert demonstrations, even if we only have trajectory data generated by model-based methods.

Accordingly, we propose a two-stage IRL learning pipeline that utilizes both imitation learning and IPMD. Our approach is graphically summarized in Fig. 2. In the first stage, we apply imitation learning on data generated via inverse kinematics to create near-optimal demonstrations (albeit dynamically infeasible), as subsequent IRL training and reward analysis require dynamically accurate demonstrations. The imitation learning style reward function $r_I$ used in this environment is defined as a weighted sum of tracking rewards at the joint level:

$$r_I = c_1 e^{-E_{\text{joint}}} + c_2 e^{-\|p_{\text{CoM}} - p^r_{\text{CoM}}\|} + c_3 e^{-\|p_o - p^r_o\|} \quad (2)$$

where $c_1, c_2, c_3$ are constant coefficients, $E_{\text{joint}}$ is a weighted Euclidean norm of the difference between the current joint position $q$ and the reference joint position $q^r$: $E^2_{\text{joint}} := w^T(q - q^r)^2$, $w, q, q^r \in \mathbb{R}^N$. $p_{\text{CoM}}$ denotes the Center of Mass (CoM) position, and $p_o$ denotes pelvis orientation. The superscript $r$ denotes the reference motion.

Using expert demonstrations generated from the first stage, the second stage employs our IPMD method to learn both the optimal policy and the associated reward function in the form of a deep neural network. Concretely, in each iteration of the IPMD algorithm, we sample state-action pairs by interacting with the environment and also sample state-action pairs from demonstrations. We then employ Temporal-Difference (TD) to evaluate our current policy given the first set of sampled pairs from the environment and apply a Mirror Descent step to improve the current policy. At the end of the iteration, we update the reward estimation through gradient descent given the two sets of sampled pairs. Due to the space limit, more details can be referred to in [25].

### B. Analysis of the Learned Reward Function

We extend our study to a detailed analysis of the learned reward function. The reward function $r_\theta$ is a deep neural

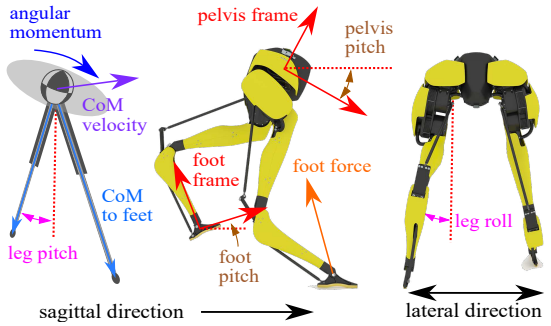


Fig. 3: Illustration of important features for Cassie locomotion.

TABLE I: Considered Features for Approximating Learned Rewards

| state | action | Euclidean norm of action |
|---|---|---|
| leg roll | leg pitch | pelvis pitch |
| hip yaw | foot pitch | foot force |
| CoM velocity | CoM angular momentum | CoM to center of foot |

network that inherently lacks interpretability due to its black-box nature. To tackle this issue, we employ a more interpretable model, Value Decomposition Network (VDN) [34], which approximates the reward function and explains the significance of locomotion features in determining the reward value. VDN maintains a monotonic relationship between its input and output by constraining the weights and biases of the network to be positive, ensuring continuous positive gradients [46]. This property of VDN allows us to establish a monotonic mapping from the state space to the reward output without compromising the learned reward's accuracy due to its usage of neural networks [46].

Additionally, we aim to explore the features that are highly relevant to bipedal locomotion but may not be directly present in the state space, such as the leg length or ground reaction force, to study how these indirectly observed features affect the reward function. To facilitate this, we extend the input space of our approximation model to include these features. The full list of selected features is in Table I and a majority of them are annotated in Fig. 3. Through this approximation, we establish a relationship between the selected features and the reward function, while keeping the IRL training process separate and intact, allowing it to preserve the expressive power of deep neural nets.

Equipped with an interpretable approximation from VDN, we proceed to further dissect the learned reward function using a set of neural network interpretation techniques. In particular, we find Integrated Gradients (IG), a widely recognized tool in the Deep Learning community, to be highly suitable for our objectives [35]. IG allows us to analyze the effect of individual features on the overall landscape of the reward function by perturbing the input and observing the resulting gradient changes, which in our case are manifested as variations in the neural network weights. We also find that directly applying IG to the original reward function itself does not yield any meaningful outcome, due to the highly nonlinear relationship between the input (states and actions) and the output (rewards). This validates the necessity of using VDN to approximate the original reward function for better reward interpretation with IG.

### C. Adaptability of the Learned Rewards on Difficult Terrains

In this context, we explore whether our learned reward function harbors generalized knowledge that enables adaptability across varying terrains. Specifically, we test its efficacy in a purely RL-driven training paradigm, without the need for additional expert demonstrations. Intriguingly, the RL guided by the learned reward not only allows training from scratch but also produces a better performance compared to policies learned from the hand-crafted reward. Even though the reward function was originally trained on flat terrain, our learned reward successfully guides the agent's learning in more complex environments.

This observation aligns well with the intuition that a well-designed reward function encapsulates generalizable environmental knowledge. To validate this point, we present results showcasing Cassie's capability to navigate difficult terrains.

More interestingly, with the understanding of reward functions, we show that factored components inside the reward function, i.e., those found during our reward function analysis, can improve the quality of locomotion behaviors. This constitutes a significant contribution to the field, as traditional algorithms often require the crafting of intricate, domain-specific reward functions.

## IV. Experiments

### A. Two-Stage Learning Setup

Our experiments of Cassie locomotion were conducted using the MuJoCo physics simulator [47]. The training pipeline consists of two main stages as illustrated in Fig. 2.

*1) First Stage – Training the Imitation Agent:* We train the Imitation agent using Soft Actor-Critic (SAC) [48]. The discount factor $\gamma$ for this stage is set to $0.99$. Both the policy and value functions are parameterized by $256 \times 256$ Multi-Layer Perceptrons (MLPs). For implementation, we adopt the state-of-the-art codebase from stable-baselines3 [49].

*2) Second Stage – Learning reward functions and policies via IRL:* We use the Inverse Policy Mirror Descent (IPMD) method described in [25]. The reward function, policy, and value functions are all represented by $256 \times 256$ MLPs.

*3) Training Parameters:* Both agents are trained using $5 \times 10^6$ samples. We employ an experience replay buffer with a capacity of $1 \times 10^6$ and utilize a batch size of $512$. The Adam optimizer [50] is employed with a learning rate set at $3 \times 10^{-4}$. These parameter settings are consistent with established norms for training Deep RL algorithms.

From a simulation experiment, the optimal expert agent obtained an episodic reward of $447.2$ while generating the corresponding expert demonstration data for the second stage; the IRL agent trained with IPMD reached a better performance—an episodic reward of $482.87$. All metric is measured by the reward $r_v$. The fact that IPMD agent outperforms the expert agent is also observed in [25]. We suspect this is due to the expert agent is trained with a discount factor but IPMD is based on average-reward. The

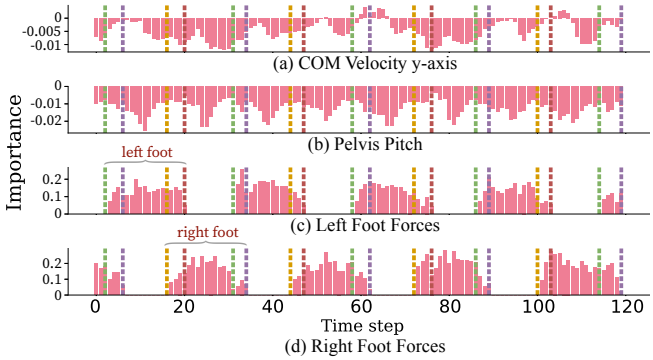
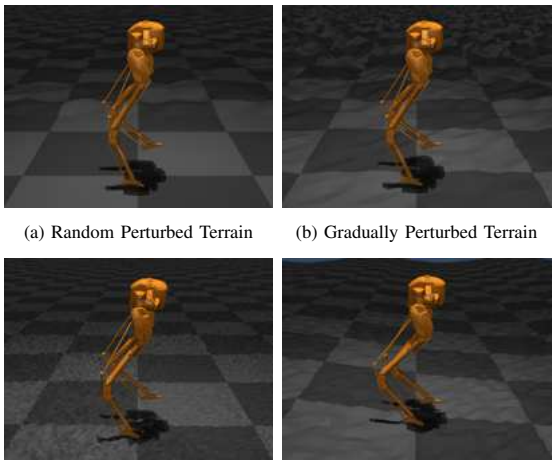



Fig. 4: The top four most important features: CoM lateral velocity, pelvis pitch angle, left and right foot forces. $y$-axis is the importance value reported by IG. The vertical dashed lines represent time steps when the foot touches and leaves the ground. Green indicates when the left foot strikes and red is for the left foot taking off from the ground. The same for orange (strike) and purple (take off) for the right foot.



(a) Random Perturbed Terrain (b) Gradually Perturbed Terrain

(c) Gravel Terrain (d) Sine Wave Terrain

Fig. 5: Random terrains generated for testing the learned reward function.

qualitative performance of the IRL agent has no distinguishable difference compared to the imitation agent, this is surprising since we learn both the reward functions and policies from scratch, while in the imitation learning case, a complicated reward function has already been established.

### B. Reward Analysis

For the Value Decomposition Network (VDN), we adhere to the same network structure as described in [46]. We gather training samples by recording the states of the Cassie robot, along with additional data necessary for computing the features of interest. We list all features we find worth investigating in Table I. As we aim to approximate the learned reward function, we use the rewards generated by $r_\theta$ as regression targets for the VDN. The optimization objective is the Mean Squared Error (MSE), thereby transforming the training of VDN into the following optimization problem: $\min_\psi$ MSE(VDN($\psi$), $r_\theta$), where $r_\theta$ is the learned reward function and $\psi$ represents the parameters of the VDN, i.e., the weights and biases in neural networks. We record and compute specified feature data as input, and collect rewards computed from those data using the learned reward functions as regression targets. We employ the Adam optimizer with a learning rate of $3 \times 10^{-4}$ to train the VDN. To interpret the contribution of each feature to the reward function, we employ Integrated Gradients (IG) [35], which is further implemented by Captum [51]. Fig. 4 demonstrates that the reward function approximated by the VDN aligns well with our intuitive understanding of what features are important for bipedal locomotion. We plot the importance change of four features to the reward during one typical Cassie walking motion executed by the IRL agent.

We find that some features of interest exhibit periodic patterns, due to the nature of the periodic walking motion. This aligns with our understanding of bipedal locomotion. Some particular features exhibit a strong influence on the reward even if they have no particular pattern. We note that pelvis pitch, plotted in Fig. 4, has significant values compared to its small-scale raw input data. We conjecture that the pelvis pitch plays an important role in maintaining the stability of the robot during walking. Other features also have strong correlations with their physical meaning. For example, the left foot has ground reaction force only when it is in contact with the ground. This is rather intuitive for robot locomotion.

### C. Adaptive Reward Function

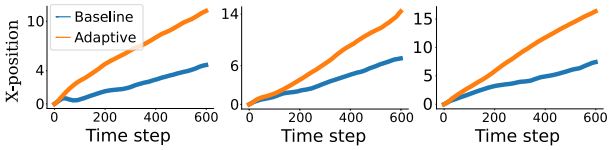
We generate a variety of uneven terrains in MuJoCo environments as shown in Fig. 5. In particular, we create (a) random perturbed terrain, (b) gradually perturbed terrain, (c) gravel terrain, and (d) sine wave terrain, each with maximum height capped at 0.2, 0.3, 0.1, 0.4 meters respectively. These categories serve to evaluate the adaptability and generalization capacity of our learned reward function.

We train the agent from scratch using SAC with a discount factor of $\gamma = 0.99$, following the same setup as in our imitation learning model. For comparative analysis, we also train a baseline RL agent with a handcrafted reward function defined as $r_h = r_f + r_s - r_c$, where $r_f$ encourages forward movement and corresponds to the sagittal velocity; $r_s$ is a locomotion survival reward, awarded when Cassie torso remains upright; and $r_c$, the control cost, is defined as $r_c = \|a\|_2$.

The baseline agent manages to navigate these terrains, albeit in a less graceful manner with jerky motions (see the submitted video). In contrast, our approach uses a modified reward function: $r = r_h + r_\theta$, where $r_\theta$ is the reward function learned from IRL. We refer to $r$ as the Adaptive reward. We record the average sagittal velocity of CoM when comparing the baseline reward model and the adaptive reward model side by side. The results can be found in Table II. We also plot the sagittal travel distance in each environment, which is shown in Fig. 6. We find that incorporating $r_\theta$ significantly accelerates learning and produces more natural and robust locomotion behaviors, substantiating the transferability of the learned reward function across domains.

### D. Analysis-based Adaptive Reward Design

With the adaptive reward, the robot is able to walk on unseen rough terrains. However, instances of undesirable




(a) Perturbed (b) Gradual (c) Gravel

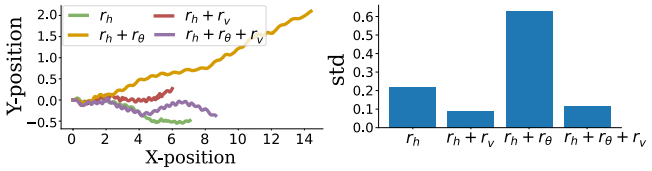
Fig. 6: Sagittal travel distance comparison between baseline model using $r_h$, and adaptive reward model using $r$. Note that even though the Baseline model can walk up to maximum time steps, it can not walk as far as the one using the adaptive reward.

TABLE II: Average Center of Mass velocity (m/s) in sagittal direction

| Terrain | Baseline | Adaptive |
|---|---|---|
| Perturbed | 0.2617 | 0.6249 |
| Gradual | 0.3970 | 0.8015 |
| Gravel | 0.4132 | 0.9106 |

walking gait still occasionally occur. Specifically, using the adaptive reward alone, Cassie's CoM exhibits a higher sagittal CoM velocity. In reality, such behavior is undesirable as this inclination creates instability during locomotion in rough terrains. Consequently, the robot needs to maneuver agilely to maintain balance during walking. This leads to the robot deviating from the original lateral position, which is reflected by large variations of CoM velocity along the lateral direction. With the understanding of the learned reward, a natural question arises: *can we further exploit the learned reward functions to shape the locomotion behavior?* We answer this question affirmatively. The top important features uncovered in the Reward Analysis improved the stability of walking behaviors when incorporated with the learned reward. As such, we incorporate important features discovered from the reward analysis to boost the stability of the robot, or "regularize" the robot's motion. To do this, we add three additional terms with high importance scores to the adaptive reward: pelvis orientation, pelvis pitch angle, and CoM velocity, which are implemented to follow their reference motions on the flat ground. We denote such rewards as $r_v = e^{-\|q_o - q_o^r\|_2} + e^{-\|q_{\text{pitch}} - q_{\text{pitch}}^r\|_2} + e^{-\|v_{\text{CoM}} - v_{\text{CoM}}^r\|_2}$, where $q_o$ denotes the pelvis orientation in a quaternion form, $q_{\text{pitch}}$ is the pelvis pitch angle, and $v_{\text{CoM}}$ is the CoM velocity.

To verify the efficacy of the $r_v$, we train RL agents with SAC on four combinations of reward functions: the baseline model $r_h$, the regularized model $r_h + r_v$, the adaptive model $r_h + r_\theta$, and the regularized adaptive model $r_h + r_\theta + r_v$. We plot the CoM trajectory, and standard deviation of the velocity drift along the lateral direction in Fig. 7. Although the adaptive model allows the robot to walk further, it has a higher deviation from its original lateral position and a higher deviation of lateral velocity. We conjecture that this is partially due to the fact that the orientation is less emphasized by the adaptive reward. We also observe that purely using the adaptive reward results in a "hopping" behavior where each walking step has a brief flight phase. In reality, such loss of ground contact can lead to a highly unstable walking motion and pose a risk of failure. Surprisingly, the integration of additional regularizing terms in the reward function $r_v$ mitigates such undesirable hopping behaviors. We plot the




(a) CoM trajectory using different reward on Gradual (b) Standard deviation of lateral velocity

Fig. 7: Results for regularizing robotics behavior.

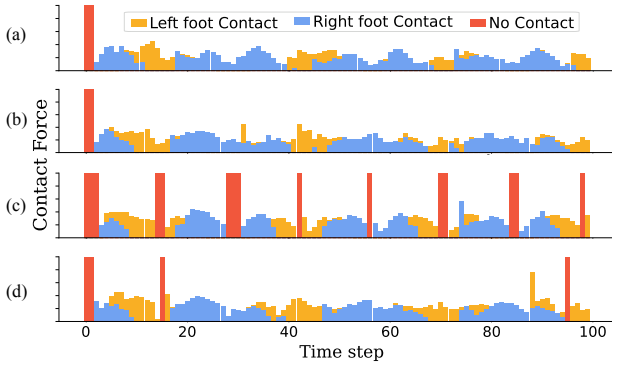



Fig. 8: Ground reaction force with four reward setups: (a) $r_h$, (b) $r_h + r_v$, (c) $r_h + r_\theta$, (d) $r_h + r_\theta + r_v$. The orange bar denotes the left foot force, while the blue the right. The red bar denotes time steps when no ground reaction force exists for either foot. Data records a random middle portion of a trajectory.

ground reaction force of all four models in Fig. 8. Time steps when undesirable behaviors (both feet are in the air) occur are annotated with red color bars.

Fig. 8(b) and (d) show a more stable and natural walking motion, compared with Fig. 8(c) (also shown in the video), indicating the efficacy of the $r_v$ reward in regulating the robot's behavior. This result further demonstrates that the augmentation of the reward function with relevant extracted features leads to improved locomotion performance.

### *E. Zero-Shot Validation*

We observe that agents trained on diverse terrains display enhanced stability when deployed in unseen environments. For example, Cassie is able to navigate sinusoidal terrains with random height variations (Fig. 5d), without additional training. This corroborates the idea that the learned reward embodies a form of generalized knowledge beneficial for robotic locomotion across a range of terrain scenarios.

## V. Conclusion

In this work, we employ an IRL method to solve bipedal locomotion problems. Our analyses reveal that the learned reward function encapsulates meaningful insights and also serves as a valuable guide to understanding the underlying principles of robotic motion. The ability to learn and adapt using the inferred reward function paves the way for new avenues of research in robotics, particularly in the domain of reward inference and environmental adaptability. Our work supports the notion that leveraging learned reward functions could substantially accelerate the design, training, and deployment of robotic systems across a myriad of real-world scenarios. Our future direction will focus on hardware implementation on the Cassie robot.

## REFERENCES


[1] H. Van Hedel, M. Biedermann, T. Erni, and V. Dietz, "Obstacle avoidance during human walking: transfer of motor skill from one leg to the other," *The Journal of physiology*, vol. 543, no. 2, pp. 709–717, 2002.

[2] R. F. Reynolds and A. M. Bronstein, "The moving platform after-effect: limited generalization of a locomotor adaptation," *Journal of neurophysiology*, vol. 91, no. 1, pp. 92–100, 2004.

[3] H. Farmer, A. Ciaunica, and A. F. d. C. Hamilton, "The functions of imitative behaviour in humans," *Mind & language*, vol. 33, no. 4, pp. 378–396, 2018.

[4] D. M. Mariscal, E. V. Vasudevan, L. A. Malone, G. Torres-Oviedo, and A. J. Bastian, "Context-specificity of locomotor learning is developed during childhood," *eneuro*, vol. 9, no. 2, 2022.

[5] S. Schaal, "Is imitation learning the route to humanoid robots?" *Trends in cognitive sciences*, vol. 3, no. 6, pp. 233–242, 1999.

[6] H. Ravichandar, A. S. Polydoros, S. Chernova, and A. Billard, "Recent advances in robot learning from demonstration," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 3, no. 1, pp. 297–330, May 2020. [Online]. Available: https://doi.org/10.1146/annurev-control-100819-063206

[7] 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*, vol. 40, no. 4, pp. 1–20, 2021.

[8] P. Florence, C. Lynch, A. Zeng, O. A. Ramirez, A. Wahid, L. Downs, A. Wong, J. Lee, I. Mordatch, and J. Tompson, "Implicit behavioral cloning," in *Conference on Robot Learning*. PMLR, 2022, pp. 158–168.

[9] J. Siekmann, Y. Godse, A. Fern, and J. Hurst, "Sim-to-real learning of all common bipedal gaits via periodic reward composition," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 7309–7315.

[10] Z. Li, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath, "Robust and versatile bipedal jumping control through reinforcement learning," 2023.

[11] J. Siekmann, K. Green, J. Warila, A. Fern, and J. Hurst, "Blind bipedal stair traversal via sim-to-real reinforcement learning," in *Robotics: Science and Systems*, 2021.

[12] T. Haarnoja, B. Moran, G. Lever, S. H. Huang, D. Tirumala, M. Wulfmeier, J. Humplik, S. Tunyasuvunakool, N. Y. Siegel, R. Hafner *et al.*, "Learning agile soccer skills for a bipedal robot with deep reinforcement learning," *arXiv preprint arXiv:2304.13653*, 2023.

[13] J. Dao, K. Green, H. Duan, A. Fern, and J. Hurst, "Sim-to-real learning for bipedal locomotion under unsensed dynamic loads," in *International Conference on Robotics and Automation*, 2022, pp. 10 449–10 455.

[14] L. Krishna, G. A. Castillo, U. A. Mishra, A. Hereid, and S. Kolathaya, "Linear policies are sufficient to realize robust bipedal walking on challenging terrains," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2047–2054, 2022.

[15] A. Y. Ng, S. Russell *et al.*, "Algorithms for inverse reinforcement learning." in *Icml*, vol. 1, 2000, p. 2.

[16] S. Arora and P. Doshi, "A survey of inverse reinforcement learning: Challenges, methods and progress," *Artificial Intelligence*, vol. 297, p. 103500, 2021.

[17] M. Wulfmeier, D. Rao, D. Z. Wang, P. Ondruska, and I. Posner, "Large-scale cost function learning for path planning using deep inverse reinforcement learning," *The International Journal of Robotics Research*, vol. 36, no. 10, pp. 1073–1087, 2017.

[18] W. Liu, J. Zhong, R. Wu, B. L. Fylstra, J. Si, and H. H. Huang, "Inferring human-robot performance objectives during locomotion using inverse reinforcement learning and inverse optimal control," *IEEE Robotics and Automation Letters*, vol. 7, no. 2, pp. 2549–2556, 2022.

[19] L. Gan, J. W. Grizzle, R. M. Eustice, and M. Ghaffari, "Energy-based legged robots terrain traversability modeling via deep inverse reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 8807–8814, 2022.

[20] L. Chen, S. Jayanthi, R. R. Paleja, D. Martin, V. Zakharov, and M. Gombolay, "Fast lifelong adaptive inverse reinforcement learning from demonstrations," in *Conference on Robot Learning*. PMLR, 2023, pp. 2083–2094.

[21] M. Wulfmeier, A. Bewley, and I. Posner, "Addressing appearance change in outdoor robotics with adversarial domain adaptation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2017, pp. 1551–1558.

[22] D. Garg, S. Chakraborty, C. Cundy, J. Song, and S. Ermon, "Iq-learn: Inverse soft-q learning for imitation," *Advances in Neural Information Processing Systems*, vol. 34, pp. 4028–4039, 2021.

[23] T. Ni, H. Sikchi, Y. Wang, T. Gupta, L. Lee, and B. Eysenbach, "f-irl: Inverse reinforcement learning via state marginal matching," in *Conference on Robot Learning*. PMLR, 2021, pp. 529–551.

[24] S. Zeng, C. Li, A. Garcia, and M. Hong, "Maximum-likelihood inverse reinforcement learning with finite-time guarantees," *arXiv preprint arXiv:2210.01808*, 2022.

[25] F. Wu, J. Ke, and A. Wu, "Inverse reinforcement learning with the average reward criterion," *arXiv preprint arXiv:2305.14608*, 2023.

[26] B. D. Ziebart, A. L. Maas, J. A. Bagnell, A. K. Dey *et al.*, "Maximum entropy inverse reinforcement learning." in *AAAI Conference on Artificial Intelligence*, 2008.

[27] B. D. Ziebart, J. A. Bagnell, and A. K. Dey, "Modeling interaction via the principle of maximum causal entropy," in *ICML*, 2010.

[28] M. L. Puterman, *Markov decision processes: Discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[29] S. Zhang, Y. Wan, R. S. Sutton, and S. Whiteson, "Average-reward off-policy policy evaluation with function approximation," in *International Conference on Machine Learning*. PMLR, 2021, pp. 12 578–12 588.

[30] Y. Jin and A. Sidford, "Towards tight bounds on the sample complexity of average-reward MDPs," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5055–5064.

[31] T. Li, F. Wu, and G. Lan, "Stochastic first-order methods for average-reward markov decision processes," *arXiv preprint arXiv:2205.05800*, 2022.

[32] J. Peters, S. Vijayakumar, and S. Schaal, "Reinforcement learning for humanoid robotics," in *Proceedings of the third IEEE-RAS international conference on humanoid robots*, 2003, pp. 1–20.

[33] W. Ouyang, H. Chi, J. Pang, W. Liang, and Q. Ren, "Adaptive locomotion control of a hexapod robot via bio-inspired learning," *Frontiers in Neurorobotics*, vol. 15, p. 627157, 2021.

[34] P. Sunehag, G. Lever, A. Gruslys, W. M. Czarnecki, V. Zambaldi, M. Jaderberg, M. Lanctot, N. Sonnerat, J. Z. Leibo, K. Tuyls *et al.*, "Value-decomposition networks for cooperative multi-agent learning," *arXiv preprint arXiv:1706.05296*, 2017.

[35] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International conference on machine learning*. PMLR, 2017, pp. 3319–3328.

[36] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *2009 IEEE International Conference on Robotics and Automation*. IEEE, 2009, pp. 763–768.

[37] 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*, vol. 41, no. 4, pp. 1–17, 2022.

[38] Y. Yang, X. Meng, W. Yu, T. Zhang, J. Tan, and B. Boots, "Learning semantics-aware locomotion skills from human demonstration," in *Conference on Robot Learning*. PMLR, 2023, pp. 2205–2214.

[39] W. Zhao, J. P. Queralta, and T. Westerlund, "Sim-to-real transfer in deep reinforcement learning for robotics: a survey," in *IEEE symposium series on computational intelligence*. IEEE, 2020, pp. 737–744.

[40] J. Nassour, V. Hugel, F. B. Ouezdou, and G. Cheng, "Qualitative adaptive reward learning with success failure maps: Applied to humanoid robot walking," *IEEE transactions on neural networks and learning systems*, vol. 24, no. 1, pp. 81–93, 2012.

[41] C. Huang, G. Wang, Z. Zhou, R. Zhang, and L. Lin, "Reward-adaptive reinforcement learning: Dynamic policy gradient optimization for bipedal locomotion," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[42] Ç. Kaymak, A. Uçar, and C. Güzeliş, "Development of a new robust stable walking algorithm for a humanoid robot using deep reinforcement learning with multi-sensor data fusion," *Electronics*, vol. 12, no. 3, p. 568, 2023.

[43] Z. Xie, G. Berseth, P. Clary, J. Hurst, and M. van de Panne, "Feedback control for cassie with deep reinforcement learning," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2018, pp. 1241–1246.

[44] Z. Li, X. Cheng, X. B. Peng, P. Abbeel, S. Levine, G. Berseth, and K. Sreenath, "Reinforcement learning for robust parameterized locomotion control of bipedal robots," in *IEEE International Conference on Robotics and Automation*. IEEE, 2021, pp. 2811–2817.
[45] Y.-M. Chen and M. Posa, "Optimal reduced-order modeling of bipedal locomotion," in *IEEE International Conference on Robotics and Automation*. IEEE, 2020, pp. 8753–8760.
[46] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson, "Monotonic value function factorisation for deep multi-agent reinforcement learning," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 7234–7284, 2020.
[47] E. Todorov, T. Erez, and Y. Tassa, "Mujoco: A physics engine for model-based control," in *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2012, pp. 5026–5033.
[48] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel *et al.*, "Soft actor-critic algorithms and applications," *arXiv preprint arXiv:1812.05905*, 2018.
[49] A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann, "Stable-baselines3: Reliable reinforcement learning implementations," *Journal of Machine Learning Research*, vol. 22, no. 268, pp. 1–8, 2021.
[50] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *International Conference on Learning Representations*, San Diega, CA, USA, 2015.
[51] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan *et al.*, "Captum: A unified and generic model interpretability library for pytorch," *arXiv preprint arXiv:2009.07896*, 2020.