On-Policy Reinforcement Learning from Failure Under Sparse Reward Environments

Anonymous authors
Paper under double-blind review

Abstract

Learning control policies in sparse reward environments is challenging due to the difficulty of designing effective exploration mechanism, which often requires dense rewards. To enable effective exploration in sparse reward environments, the state-of-art methods rely on expert demonstrations that guide the exploration towards optimal or sub-optimal policies. However, one key challenge is to obtain expert demonstration, which can be cumbersome and even impossible. To address the challenge, this paper presents a new reinforcement learning (RL) method that learns control policies from only failed experiences under sparse reward environments, via designing a new exploration mechanism specifically for failed experiences. Our approach is based on training a discriminator that calculates the dissimilarity between the state-action pair generated from the current policy and the one sampled from failed experiences. This dissimilarity is then used to augment the sparse reward signals generated from the environment in the form of a shaped reward function, which is then used to guide the policy search. We finally present a few experimental studies that show the effectiveness of the proposed approach in learning effective policies when comparing with some state-of-the-art methods.

1 Introduction

Machine learning has been revolutionized at a unprecedented pace, among which reinforcement learning (RL) has achieved great success as a decision making tool in dense reward environments where reward signals are readily available (Mnih et al., 2013; Bellemare et al., 2013). Those environments, such as Atari and Mujoco, provide clear scores as the rewards for the RL agents to optimize their strategies to accomplish the tasks. However, RL agents often struggle with their strategies optimization under sparse reward environments (Powell, 2012; Siddique et al., 2023; White et al., 2024). In practical, it is very difficult to design proper reward functions for environments that are characterized with high dimensional action and state spaces, such as robotic environments (Plappert et al., 2018). Hence, RL agents often suffer from inadequate exploration, leading to delay in learning or local optima under sparse reward environments (Wu et al., 2023).

Hindsight experience replay (HER) is one approach developed to improve the sample efficiency under sparse reward environments for off-policy reinforcement learning algorithms (Andrychowicz et al., 2017). By assigning some achievable intermediate goals, HER can provide a more effective exploration mechanism to improve learning process. Despite its benefits, HER remains time-consuming due to computational load when learning from multiple intermediate goals to achieve the desired goal. Alternatively, further utilization of demonstration is shown to enable the improvement of learning process. For example, Nair et al. (Nair et al., 2018) proposed a method combining behavior cloning (BC) and HER to enhance the learning process. This method trains the RL agent's policy in two parallel policy updating processes, one with deep deterministic policy gradient (DDPG) (Lillicrap et al., 2015) associated with HER, the other with behavior cloning under expert demonstrations. During training, BC can improve the exploration by the expert guidance from additional expert demonstrations. However, HER can only be applied in off-policy RL methods where RL

agents learn the policy from the collected data by following a different behavior policy than the one being learned. It also shows effective mostly for deterministic policy updating, since HER assigns those intermediate goals based on the achieved states within the buffer (Andrychowicz et al., 2017). Yet on-policy RL methods where RL agents learn the policy by following the current policy still encounter inadequate exploration problem under sparse reward environments. Additional methods include, e.g., policy shaping that leverages domain knowledge to provide guidance to RL agents (Griffith et al., 2013), and reward shaping that accelerates learning by providing additional exploration guidance via augmenting the available reward signals or modifying the environments' dynamic (Cederborg et al., 2015). Bingyi et al. (Kang et al., 2018) trained a classifier with expert demonstrations to calculate the likelihood between real-time RL agent's action and the action sampled from the expert demonstrations under a same state. This likelihood serves as an augmenting signal, complementing the reward function, to guide the exploration direction by assessing the similarity between the RL agent's current state action pair and expert demonstrations distribution.

However, in real world, expert demonstrations are costly to collect due to its need for specific expertise and resources (Shiarlis et al., 2016). In contrast, failed experiences can be less challenging to collect in various scenarios compared to expert ones. We define a failed experience that differs from an expert demonstration by a margin. There are limited works exploring the use of failed experiences in RL. Recent research has shown promising results by incorporating inverse reinforcement learning (IRL), Bayesian inverse reinforcement learning (BIRL), and halfspace technique with some failed data (Xie et al., 2019). However, these methods still require additional expert demonstrations to guide the exploration, which differs from our approach that only utilizes failed experiences.

Our study explores the potential value of failed experiences in guiding RL agent's exploration under sparse reward environments. Failure, often ignored in reinforcement learning, can serve as negative examples that are able to guide the learning and exploration towards the right direction in real world (Harteis et al., 2008). Hence, failed experiences can also be used to guide the RL agent's exploration direction by serving as negative examples that should be avoided. The objective of the paper is to develop a novel method that can leverage failed experiences to guide the RL agent's exploration. Specifically, we propose a novel reward function that improves RL agent's exploration during learning process by efficiently utilizing failed experiences.

Our contributions are summarized as follows. First, we show that failed experiences also have their values to improve learning in on-policy RL scenarios under sparse reward function where exploration is challenging. Second, we show that our approach is different from RL from expert demonstrations methods, which normally require high quality demonstrations that are not easily accessible. Failed experiences possess more flexibility and enable us to utilize a board range of data. Third, we provide experimental studies to validate that our method can enhance RL performance by utilizing only failed experiences under sparse reward environments when comparing it with learning from expert demonstrations.

2 Preliminaries and Background

We consider the general Markov Decision Process (MDP), which is defined by a tuple $(\mathcal{S}, \mathcal{A}, P, R, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ the action space, P the state transition probability distribution, R the reward function, and $\gamma \in [0, 1)$ is the discount factor that limits the influence of infinite future rewards. At each state $s \in \mathcal{S}$, the RL agent takes an action $a \in \mathcal{A}$, receives a reward r from $R(s, a)$, and moves to the next state s' determined by $P(s'|s, a)$. The goal is to learn a policy π that maps states to actions to maximize the expected discounted accumulative rewards. This can be formulated by the action-value function

$$Q(s, a) = \mathbb{E}_{a_t \sim \pi} \left[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t) \right], \tag{1}$$

where t represents the t^{th} timestep in the training process. The performance of a policy π is normally evaluated by the discounted accumulative rewards

$$J(\theta_\pi) = \mathbb{E}_{s\sim\mu}\left[Q(s, \pi(s|\theta_\pi))\right], \tag{2}$$

where μ represents the initial state distribution and θ_π is the policy network parameter.

As can be observed above, the update of policy relies much on the accumulative rewards. However, standard RL methods may perform poorly due to the lack of exploration in sparse reward environments. One promising technique that employs human demonstrations to guide the RL agents' exploration under sparse reward environments has shown success with different approaches, such as Generative Adversarial Imitation Learning (GAIL) (Ho & Ermon, 2016), where agent is expected to mimic the behaviors from expert demonstrations. In GAIL, the agent learns the policy via imitation learning while training a discriminator to distinguish between its trajectories and expert demonstrations. The objective for the agent is to generate trajectories that are indistinguishable to the discriminator. However, expert demonstrations are costly to obtain, demanding a substantial amount of data collection (Xie et al., 2019). In contrast, failure, or failed experiences, are normally more accessible. In this paper, we define a failed experience as a trajectory $\tau_f = (s_0^f, a_0^f, ..., s_{t-1}^f, a_{t-1}^f, s_t^f)$ that does not lead to the desired outcome. Different from existing studies on the use of expert demonstrations, we focus on studying how to use failed experiences to guide the RL agent's search direction during exploration, eventually allowing the agent to accomplish the given task using only failed experiences. Specifically, we seek to design an augmented reward function with an integrated discriminator that quantifies the likelihood between the agent's trajectories and failed experiences. Subsequently, RL agents can optimize their policies by maximizing the accumulative rewards generated from the augmented reward function.

3 Proposed Method: A MinMax Formulation

In the context of this paper, failed experiences are defined with a board range of trajectories that are distinguish from expert demonstrations by a margin. The proposed new objective function, leveraging failed experiences as negative examples to guide the RL agent's exploration, is given by

$$\min_\theta \max_\omega L = -\eta(\pi_\theta) - \lambda_1 \mathbb{E}_{\pi_\theta}[\log(D_\omega(s,a))] + \mathbb{E}_{\pi_f}[\log(1 - D_\omega(s,a))] - H(\pi_\theta)), \tag{3}$$

where $\eta(\pi_\theta)$ represents for the vanilla objective of RL that calculates the accumulative rewards with policy π_θ, λ_1 is a tunable parameter that controls the scale of the output of a discriminator D measuring the dissimilarity between RL agent's current policy π_θ and failed experiences distribution π_f, and (s, a) is the state-action pair that sampled from the current policy. As we only provide failed experiences to the discriminator aiming to distinguish the current policy and the failed experiences, the discriminator is updated with binary cross-entropy method (Mao et al., 2023), which is formulated as $\mathbb{E}_{\pi_\theta}[\log(D_\omega(s,a))] + \mathbb{E}_{\pi_f}[\log(1 - D_\omega(s,a))]$, where ω is the parameter of the discriminator. To further avoid potential overfitting problem, we introduce a casual entropy term $-H(\pi_\theta)$ as a regularization. Hence, the proposed objective function aims to maximize the accumulative rewards and the similarity output from the discriminator by minimizing θ and maximizing ω respectively, so that the RL agent can learn to improve performance while deviating from the failed experiences. However, it is not feasible to obtain or train a policy π_f that can mimic the failed experiences as they are too board. Alternatively, we employ the occupancy measurement $\rho_\pi(s) = \sum_{t=0}^{\infty} \gamma^t P(s_t = s|\pi)$ proposed in (Syed & Schapire, 2007) that characterizes the normalized distribution of state-action pairs when implementing the policy π. The probability of taking an action given a state following a policy π can be then defined as $\pi_\rho(a|s) = \frac{\rho(s,a)}{\sum_{a_i} \rho(s,a_i)}$.

Similar to Generative Adversarial Networks (GANs) (Goodfellow et al., 2020), the proposed method is also a minimax problem as the RL agent seeks to maximize the accumulative rewards while the discriminator aims to distinguish the current policy and the failed experiences. However, the accumulative reward term $\eta(\pi_\theta)$ is ineffective or unavailable in sparse reward setting. To address

the issue, we further propose a new shaped reward function that is able to indicate the potential value of the chosen state-action pair. The new shaped reward function is given by

$$r'(s,a) = r(s,a) + \lambda_1 \log(D_{\omega_i}(s,a)) + \frac{1}{1+e^{-A(s,a)}}, \tag{4}$$

where ω_i is the parameter of the discriminator at timestep i, $A(s,a)$ is the advantage function of the current state-action pair (s,a) that sampled from the current policy. The advantage function $A(s,a)$ is computed with the difference between the action-value function $Q(s,a)$ and the state-value function $V(s)$, i.e., $A(s,a) = Q(s,a) - V(s)$, quantifying the expected additional reward that the RL agent can obtain by taking action a in state s. Note that we pass $A(s,a)$ through a *sigmoid* function (Han & Moraga, 1995) to maintain the same dimension as the discriminator's output. By combining (4) and (3), the ultimate minmax problem can be summarized as

$$\min_{\theta} \max_{\omega} L = -\mathbb{E}_{\pi_\theta}(r'(s,a)) - \lambda_1 \mathbb{E}_{\pi_\theta}[\log(D_\omega(s,a))] + \mathbb{E}_{\pi_f}[\log(1 - D_\omega(s,a))] - H(\pi_\theta)). \tag{5}$$

4 Experiments and Results

To evaluate the effectiveness of our proposed method, we conduct experiments under four MuJoCo environments (Towers et al., 2023), namely, HalfCheetah, Hopper, Humanoid and Walker2D. Specifically, HalfCheetah involves controlling a 2-dimensional half cheetah agent to maximize forward velocity while maintaining balance. The agent's goal is to run as fast as possible without falling over. Hopper features a single leg robot that learns to hop forward while maintaining balance. The agent's objective is to keep hopping forward without falling. Humanoid involves controlling a humanoid robot with multiple degrees of freedom to perform walking. The goal of the agent is to walk forward as fast as possible without falling over. Walker2D is similar to the Humanoid environment but features a simpler model of a bipedal walker, where the goal of the agent is to walk forward while maintaining balance.

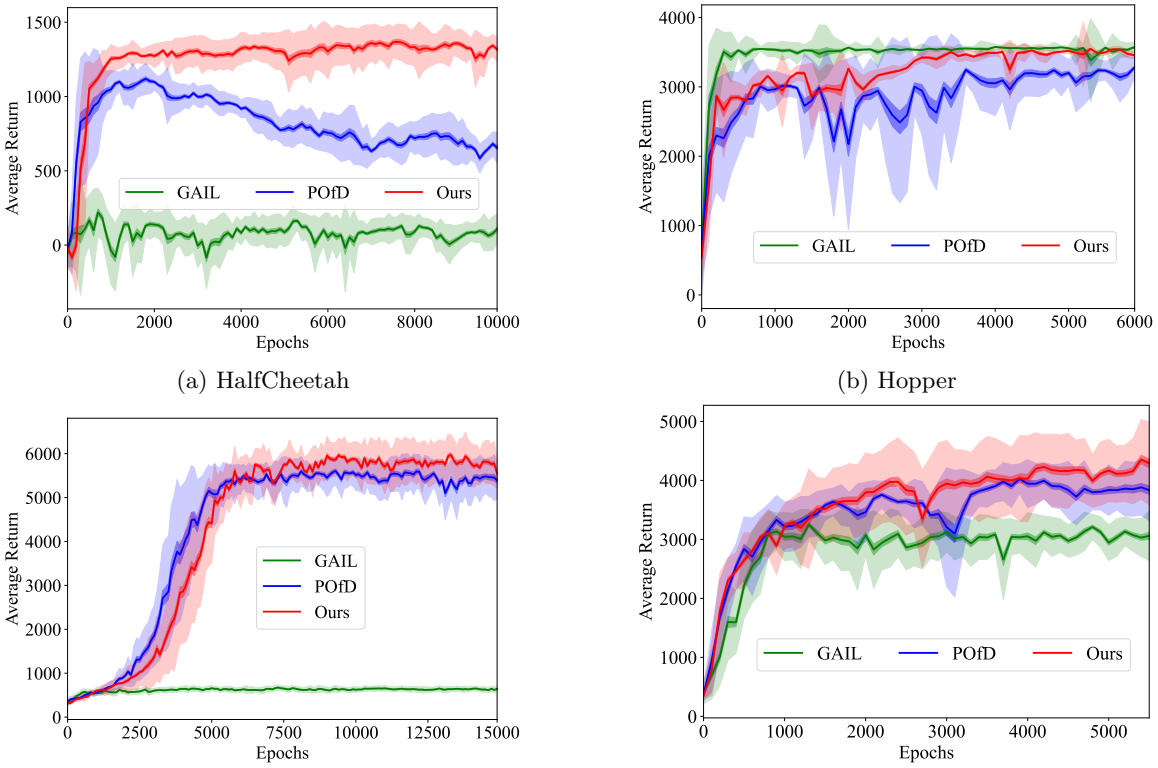
These environments are all characterized with continuous action space defined as a 6-dimensional array in HalfCheetah and Walker2D, a 3-dimensional array in Hopper and a 7-dimensional array in Humanoid. Each element within the action space represents for the torque applied in one hinge joint of the agent. The state space in these environments encompasses the kinematic information of the agent. The reward function employed in our experiments is sparse, which is intentionally set to provide a ground truth reward at a specified frequency α. For example, if α is set to 10, the agent receives an environment reward only every 10 timesteps during training.

The failed experiences are generated by modifying the actions from expert demonstrations into random values. The expert demonstrations are obtained from running Proximal Policy Optimization (PPO) (Schulman et al., 2017) with dense reward environments. It is worth noting that our method of collecting failed experiences can guarantee that the failed experiences are completely failed. While in practical, failed experiences are even easier to collect, for instance, collecting trajectories under a stochastic policy.

To evaluate the effectiveness of the proposed approach, we compare its performance against GAIL (Ho & Ermon, 2016) under dense reward environments, Policy Optimization from Demonstrations (POfD) where demonstrations are collected from expert under sparse reward environments (Kang et al., 2018). GAIL is also a minmax algorithm focusing on guiding the RL agent exploration by utilizing expert demonstration. As the success of GAIL is mostly observed in dense reward environments, we apply it in the same environments as other methods with dense rewards. POfD aims to solve the sparse reward problem by utilizing expert demonstrations to design a shaped reward function. The proposed method in this paper is labeled as 'Ours' in the results.

To ensure the reproducibility of our experiments, we run each algorithm for 5 times with different seeds and show the results as the mean and corresponding standard deviation in the form of standard error. In particular, we set λ_1 to 0.1 in the proposed objective function across all environments. To provide a fair comparison, we use accumulative rewards under environments' original reward setting

(dense reward) as a metric to evaluate POfD and our proposed method trained in sparse reward setting. It can be observed in Fig. 1 that the proposed method can achieve higher accumulative rewards than both GAIL and POfD in HalfCheetah, Humanoid and Walker2D. However, it converges at a slower rate than GAIL in Hopper. This underscores that POfD is very sensitive to both quality and quantity of the expert demonstrations. Furthermore, GAIL can achieve a good performance when the complexity of the environment is low, e.g., Hopper. Moreover, it is difficult for the RL agent to outperform the expert demonstrations with POfD as expert demonstrations are the only guidance for the exploration in sparse reward environments. These results also show the value and impact of the proposed method to learn from failed experiences and leverage a diverse range of unsuccessful experiences.



(a) HalfCheetah

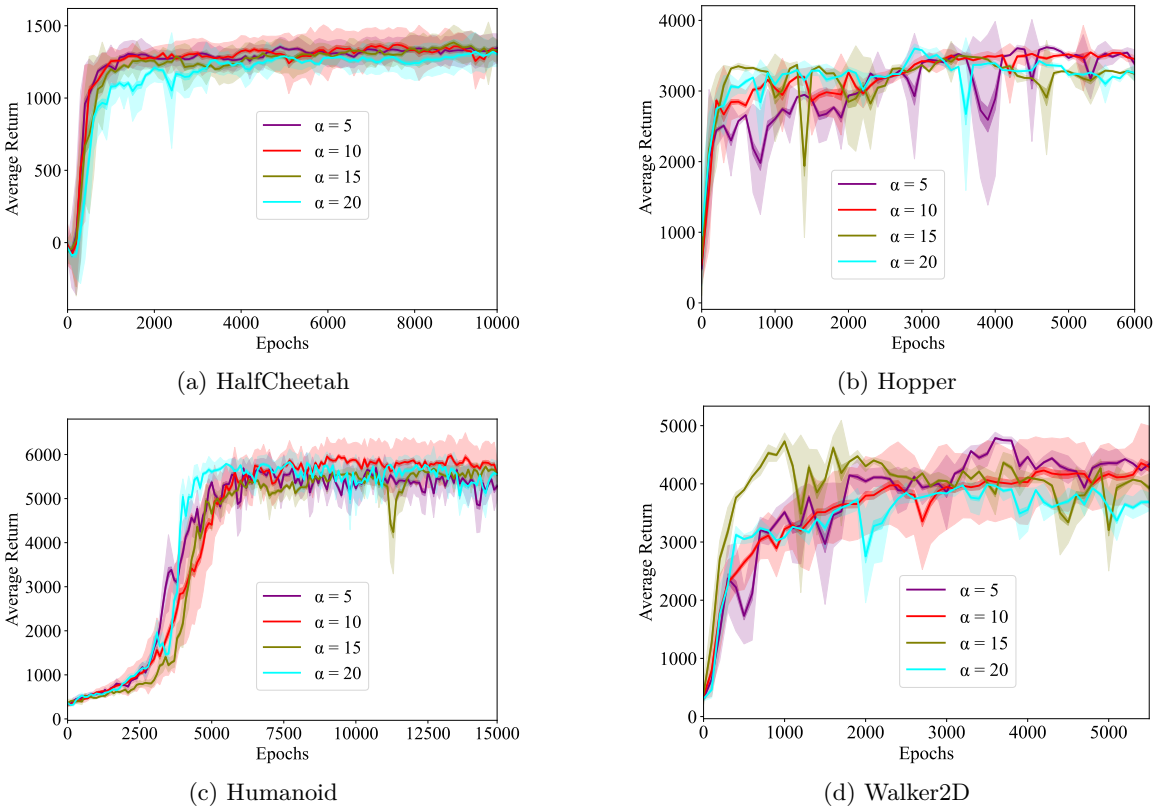
(b) Hopper

(c) Humanoid

(d) Walker2D

Figure 1: Learning curves for four tasks using different algorithms. The plots show mean (solid line) ± standard error (dark shaded area) and standard deviation (light shaded area) over 5 runs.

In addition, we further test the case when different α, namely frequencies to provide true environment reward, are used in the proposed approach. We select α as $5, 10, 15$, and 20 respectively in these four environments to evaluate the impact of α. Fig. 2 shows the comparison among different α, which indicates that a higher α may lead to a slightly worse performance. In other words, increased sparsity may negatively impact the performance. However, the performance degradation is rather limited, which shows the effectiveness and impact of the proposed learning from failure approach. By comparing Fig. 2 and Fig. 1, it can be observed that the proposed approach under different α still outperforms POfD and GAIL in most cases, further showing the need and importance of leveraging failure in the policy learning process.



(a) HalfCheetah (b) Hopper

(c) Humanoid (d) Walker2D

Figure 2: Learning curves for four environments using different α. The plots show mean (solid line) $\pm$ standard error (dark shaded area) and standard deviation (light shaded area) over 5 runs.

5 Conclusion and Future Work

In this paper, we proposed a novel reinforcement learning approach that utilizes only failed experiences to learn control policies effectively in sparse reward environments. The proposed approach is based on the design of a shaped reward function which involves a trained discriminator that indicates the dissimilarity between the state-action pairs generated and sampled from the current policy and the failed experiences respectively. We further conducted experiments to compare the proposed approach with two relevant state-of-art methods, namely GAIL and POfD. The results show that our proposed method can outperform both GAIL and POfD in most environments. One limitation of the proposed method is that the shaped reward function relies heavily on the performance of the discriminator. As failed experiences can be very board, the discriminator can be less accurate when the environments are complex. Considering the aforementioned limitation, one future research direction is to design a general discriminator that can cover a broad range of failed experiences by extracting the feature of failed experiences.

References

Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in Neural Information Processing Systems*, 30, 2017.

Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47: 253–279, 2013.

Thomas Cederborg, Ishaan Grover, Charles L Isbell Jr, and Andrea Lockerd Thomaz. Policy shaping with human teachers. In *IJCAI*, pp. 3366–3372, 2015.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks. *Communications of the ACM*, 63(11):139–144, 2020.

Shane Griffith, Kaushik Subramanian, Jonathan Scholz, Charles L Isbell, and Andrea L Thomaz. Policy shaping: Integrating human feedback with reinforcement learning. *Advances in Neural Information Processing Systems*, 26, 2013.

Jun Han and Claudio Moraga. The influence of the sigmoid function parameters on the speed of backpropagation learning. In *International Workshop on Artificial Neural Networks*, pp. 195–201. Springer, 1995.

Christian Harteis, Johannes Bauer, and Hans Gruber. The culture of learning from mistakes: How employees handle mistakes in everyday work. *International Journal of Educational Research*, 47 (4):223–231, 2008.

Jonathan Ho and Stefano Ermon. Generative adversarial imitation learning. *Advances in Neural Information Processing Systems*, 29, 2016.

Bingyi Kang, Zequn Jie, and Jiashi Feng. Policy optimization with demonstrations. In *International Conference on Machine Learning*, pp. 2469–2478. PMLR, 2018.

Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

Anqi Mao, Mehryar Mohri, and Yutao Zhong. Cross-entropy loss functions: Theoretical analysis and applications. In *International Conference on Machine Learning*, pp. 23803–23828. PMLR, 2023.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

Ashvin Nair, Bob McGrew, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Overcoming exploration in reinforcement learning with demonstrations. In *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6292–6299. IEEE, 2018.

Matthias Plappert, Marcin Andrychowicz, Alex Ray, Bob McGrew, Bowen Baker, Glenn Powell, Jonas Schneider, Josh Tobin, Maciek Chociej, Peter Welinder, et al. Multi-goal reinforcement learning: Challenging robotics environments and request for research. *arXiv preprint arXiv:1802.09464*, 2018.

Warren B Powell. Ai, or and control theory: A rosetta stone for stochastic optimization. *Princeton University*, pp. 12, 2012.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

Kyriacos Shiarlis, Joao Messias, and Shimon Whiteson. Inverse reinforcement learning from failure. 2016.

Umer Siddique, Abhinav Sinha, and Yongcan Cao. Fairness in preference-based reinforcement learning. *arXiv preprint arXiv:2306.09995*, 2023.

Umar Syed and Robert E Schapire. A game-theoretic approach to apprenticeship learning. *Advances in Neural Information Processing Systems*, 20, 2007.

Mark Towers, Jordan K. Terry, Ariel Kwiatkowski, John U. Balis, Gianluca de Cola, Tristan Deleu, Manuel Goulão, Andreas Kallinteris, Arjun KG, Markus Krimmel, Rodrigo Perez-Vicente, Andrea Pierré, Sander Schulhoff, Jun Jet Tai, Andrew Tan Jin Shen, and Omar G. Younis. Gymnasium, March 2023. URL https://zenodo.org/record/8127025.

Devin White, Mingkang Wu, Ellen Novoseller, Vernon J Lawhern, Nicholas Waytowich, and Yongcan Cao. Rating-based reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 10207–10215, 2024.

Mingkang Wu, Feng Tao, and Yongcan Cao. Value of potential field in reward specification for robotic control via deep reinforcement learning. In *AIAA SCITECH 2023 Forum*, pp. 0505, 2023.

Xu Xie, Changyang Li, Chi Zhang, Yixin Zhu, and Song-Chun Zhu. Learning virtual grasp with failed demonstrations via bayesian inverse reinforcement learning. In *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1812–1817. IEEE, 2019.