Policy Optimization using Horizon Regularized Advantage to Improve Generalization in Reinforcement Learning

Nasik Muhammad Nafi
Kansas State University
Manhattan, KS, USA
nnafi@ksu.edu

Raja Farrukh Ali
Kansas State University
Manhattan, KS, USA
rfali@ksu.edu

William Hsu
Kansas State University
Manhattan, KS, USA
bshu@ksu.edu

Kevin Duong
Kansas State University
Manhattan, KS, USA
kduong@ksu.edu

Mason Vick
Kansas State University
Manhattan, KS, USA
mtvick@ksu.edu

ABSTRACT

In this work, we focus on improving the generalization performance of a reinforcement learning (RL) agent in diverse environments. We observe that in environments created under the Contextual Markov Decision Process (CMDP), where an environment's dynamics and attribute distribution change across contexts, the generated episodes are highly stochastic and unpredictable. To improve generalization in such scenarios, we present Horizon Regularized Advantage (HRA) estimation that enables robustness to the underlying uncertainty of episode duration. Using three challenging RL generalization benchmarks Procgen, Crafter, and Minigrid we demonstrate that our proposed approach outperforms the Proximal Policy Optimization (PPO) baseline that uses classical single exponential discounting-based advantage estimate. We also incorporate HRA into another generalization-specific approach (APDAC), and the results indicate further improvement in APDAC's generalization ability. This denotes the effectiveness of our approach as a generic component that can be incorporated into any policy gradient method to aid generalization.

KEYWORDS

reinforcement learning, generalization, discounting, advantage estimation, horizon regularization, policy optimization

ACM Reference Format:
Nasik Muhammad Nafi, Raja Farrukh Ali, William Hsu, Kevin Duong, and Mason Vick. 2024. Policy Optimization using Horizon Regularized Advantage to Improve Generalization in Reinforcement Learning. In *Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 9 pages.

1 INTRODUCTION

Deep neural networks have paved the way for the recent advances in machine learning and enabled powerful RL agents that can master games and real-world applications alike [5]. However, neural networks are sensitive to the underlying training data distribution and hence memorize the data on which they are trained (overfitting), as their objective is to minimize the empirical prediction error [50]. In RL, this manifests into agents learning training trajectories and being unable to *generalize* well, not only to unseen states but also across environments [16, 39]. Generalization in RL refers to the capability of an agent to perform well in similar but unseen environments and is currently seen as an active research challenge. Training deep RL algorithms is known to be data-intensive and given a sufficiently large set of samples, they can learn a specific skill [19, 33, 34] but tend to overfit even with large training samples [10, 13, 18, 27]. To facilitate research on this issue, newer benchmarks have been developed under the Contextual Markov Decision Process (CMDP) framework. In CMDP, different episodes correspond to different variations of environments where the variation can be identified by a context, however, the episodes share some basic properties and high-level goals. Procedurally generated (PCG) environments such as Procgen [10], Minigrid [9], etc, and set of similar robotics tasks [53] are examples of such benchmarks. CMDP enables the evaluation of generalization through held-out contexts used only in testing or generating an infinite number of contexts.

In this work, we look at generalization in CMDP from a new perspective of reward discounting that aims to avoid overfitting of the advantage estimate. RL algorithms generally specify a discount factor γ s.t. $0 \leq \gamma < 1$, that exponentially discounts the future reward r_t at step t as $\gamma^t r_t$ [49]. Such exponential discounting guarantees

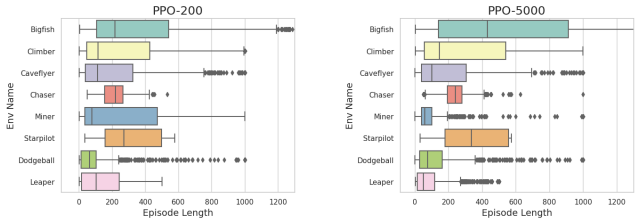

Figure 1: Distribution of episode lengths for a fully trained PPO agent estimated based on 1000 episodes randomly sampled from the test levels. Left: the agent is trained on 200 levels. Right: the agent is trained on 5000 levels; this denotes the inherent uncertainty and variation in completion time across levels (contexts) irrespective of the agent's expertise.

Proc. of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

the theoretical convergence of the value function and stabilizes optimization. The form and magnitude of the discount function itself establish strong priors over the solutions learned, and the magnitude of the discount factor sets a fixed *effective horizon* for the agent such that all rewards beyond that point are considered insignificant [28]. Exponential discounting of future rewards is consistent with the prior belief that there exists a known, *fixed* risk or hazard rate for the agent in the environment (hazard rate is defined as the per-time-step risk the agent incurs as it acts in the environment) [48][17]. We argue that this assumption of a fixed hazard rate (consequently, fixed episode length) in an environment does not hold in a CMDP setting.

In support of our claim, we identify that in a procedurally generated environment (an example of CMDP), where an environment's dynamics and attribute distribution change across levels, the generated levels are highly diverse. Different environment context indicates different degrees of uncertainty, yielding high variance in the observed episode length. To illustrate this uncertainty in the environments, in Figure 1 we plot the distribution of episode length measured over 1000 episodes sampled randomly from the test levels for 8 Procgen environments. The episodes were generated using a learned PPO [47] policy trained on 25M time steps. Even for a reasonably trained agent, we see that the episode length varies significantly for each environment. Raileanu and Fergus [42] show that even starting from semantically identical states, episode lengths can vary a lot due to variations in the level generation (contexts). The value estimate of a state highly depends on the length of the episode. For say, if the agent receives a reward of 10 at the end of the episode, then the short episode will have a higher expected value than the long episode due to the variable γ. Thus, a fixed effective horizon used as in the single exponential discounting may fail to better assess expected future rewards. Mandal et al. [31] also describes the role of an optimal effective horizon in the presence of episode length uncertainty. Additionally, we identify that the auxiliary task of learning over multiple horizons as presented in [17] collapses in the case of the actor-critic algorithm. This is due to the fact that in value-based algorithms the Q function directly defines the policy (*argmax*), while in actor-critic there are distinct value functions and policy functions. To address the uncertainty in the unseen environment context and achieve better generalization using an actor-critic algorithm, we propose to use a horizon-regularized advantage estimate that considers multiple discount factors (horizons). In summary, our contributions include:

- We show that CMDP indeed implies a high degree of uncertainty in the episode length or task completion time.
- We argue that a single discount factor introduces a very restricted inductive bias. To address this, we propose to mix advantage estimates from different discount factors to smooth the estimate so that it can better approximate the advantage in unseen scenarios with unknown episode lengths.
- We evaluate our approach on 3 different generalization benchmarks - Procgen, Crafter, and Minigrid; and our approach significantly outperforms PPO [46] and other baselines that use single exponential discounting.

2 RELATED WORK

Generalization in Deep RL. Recent studies have highlighted the inability of RL agents to generalize to new scenarios [11, 16, 40] which has led to an increasing effort on developing intelligent agents that avoid overfitting and generalize well to unseen data [18, 27, 30, 44]. Methods that have been used with some success include regularization techniques like dropout [23], batch normalization [11, 22, 23], and data augmentation [12, 43, 51, 52, 55]. Raileanu and Fergus [42] use decoupled policy and value networks to improve generalization while Nafi et al. [36, 37] present the potential of partial decoupling to improve generalization. Cobbe et al. [13] introduce phase-wise training of decoupled actor-critic architecture that ensures better sample efficiency and generalization. Paischer et al. [41] show that storing compact abstraction of the observation history using language model allows generalization. Zhang et al. [54] and Agarwal et al. [1] use bisimulation metrics to measure similarity between states with the aim of learning task-relevant representations. Mazoure et al. [32] propose to predict the future states by maximizing the mutual information between its internal representation of successive time steps. Bengio et al. [7] investigate the link between interference and generalization in temporal difference (TD) learning and suggest that TD causes low interference that leads to under-generalizing parameters. A feature-swapping regularization technique to avoid observational overfitting is proposed in [8]. Generalist-specialist training framework, as introduced in [25], alternates between two model training phases - one that encourages the development of general skills and another that promotes specialization in specific tasks or sub-domains. Igl et al. [24] and Lyle et al. [30] leverage policy distillation to improve generalization. Recently, the importance of balanced exploration to find a generalizable policy has been demonstrated [26].

Discounting. A lower discount rate has been shown to have the effect of a regularizer that can improve generalization [4]. However, determining the appropriate exponential discount factor for a particular environment remains challenging [31]. Nafi et al. [38] presents a method that utilizes randomly generated discount factors to simulate augmented value targets and use them to reduce value function overfitting. Beyond the classical exponential discounting scheme, hyperbolic discounting has been studied in the fields of behavioral psychology, economics, neuroscience, and lately, to a limited extent, in reinforcement learning. Sozou [48] proposed a per-time-step death via the hazard rate, whereas Dasgupta and Maskin [14] proposed that uncertainty over the timing of rewards leads to preference reversals as exhibited in hyperbolic discounting. Alexander and Brown [3] proposed a hyperbolic discounting-based temporal difference (TD) learning method. Although TD learning relies on exponential discounting in its calculation, naive modification to discount hyperbolically has been shown to be inconsistent. Kurth-Nelson and Redish [29] proposed the modeling of hyperbolic discounting via distributed exponential discounting. Fedus et al. [17] extended this formulation to deep reinforcement learning by approximating hyperbolic discounting from exponential discounting and evaluated their approach using a value-based method, Rainbow [21], on the Arcade Learning Environment (ALE) [6] benchmark. However, in this work, we propose a horizon-regularized advantage estimate that leverages multiple discount factors.

3 BACKGROUNDS

3.1 Contextual Markov Decision Process

We consider a Contextual Markov Decision Process (CMDP) represented by $\mathcal{M} = (\mathcal{S}, \mathcal{A}, C, \mathcal{T}, r, \mu_C, \mu_S)$ where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, C is the context space, $\mathcal{T}(s'|s,a)$ is the transition function, r is the reward function, μ_C is the context distribution, and μ_S is the initial state distribution that depends on the selected context. The context for an episode is selected based on the distribution $c \sim \mu_C$. Following $s_0 \sim \mu(\cdot|c)$, an initial state is sampled for the selected context (episode). After that, the successive states for that episode given the selected context are sampled according to the distribution $s_{t+1} \sim \mathcal{T}(\cdot|s_t, a_t, c)$. Let d_π^c represent the state distribution when the agent acts according to the policy π under the context c. The agent has access to a small subset of contexts during training. However, the ultimate objective is to learn a policy π that maximizes, $\mathcal{G} = \mathbb{E}_{c\sim\mu_C, s\sim d_\pi^c, a\sim\pi(s)}[r(s,a)]$, the expected return over *all possible contexts*.

3.2 Proximal Policy Optimization

Proximal Policy Optimization (PPO) is the widely used policy gradient method [47], and for this work, we use PPO as a baseline and build our approach on top of PPO. While learning from high-dimensional image observation, as the policy and value function approximator, PPO generally leverages a shared neural network. Given that the network is parameterized by θ, PPO optimizes the following joint objective:

$$J_{PPO}(\theta) = J_\pi(\theta) - \alpha_v L_V(\theta) + \alpha_s S_\pi(\theta) \tag{1}$$

where $J_\pi(\theta)$ is the policy gradient objective, $L_V(\theta)$ is the value loss, α_v is the coefficient for value loss, $S_\pi(\theta)$ is the entropy bonus, and α_s is the coefficient for entropy bonus. PPO is primarily based on the Trust Region Policy Optimization (TRPO) [45] method. Both of them maximize the following surrogate policy objective:

$$J_\pi(\theta) = \hat{\mathbb{E}}_t[r_t(\theta)\hat{A}_t] \tag{2}$$

where $r_t(\theta) = \frac{\pi_{(\theta)}(a_t|s_t)}{\pi_{(\theta)_{old}}(a_t|s_t)}$ is the probability ratio between the new policy and the old policy, and $\hat{A}_t$ is the advantage estimate at timestep t. Unlike TRPO, PPO prevents excessively large policy updates by clipping the value of $r_t(\theta)$ to the intervals of $[1-\epsilon, 1+\epsilon]$. The minimum between this clipped value and the original value of $r_t(\theta)$ is then selected as $r_t(\theta)$. Thus, PPO optimizes the following clipped surrogate objective for policy optimization:

$$J_\pi(\theta) = \hat{\mathbb{E}}_t\Big[min\big(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t\big)\Big] \tag{3}$$

4 METHODOLOGY

In order to achieve a generalizable policy, we aim to learn a good advantage estimate to guide policy optimization. We identify that fixed exponential reward discounting results in an overfitted advantage estimate that restricts generalization. Thus, we propose to optimize the policy using a horizon regularized advantage estimate to achieve generalization.

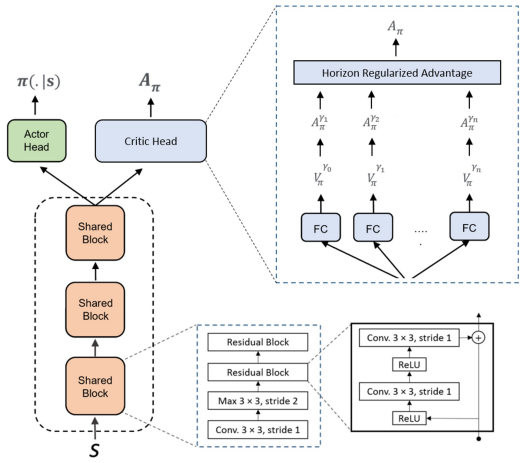

Figure 2: Details of our proposed architecture. Each *Shared Block* is identical to the IMPALA CNN architecture [15]. The *Critic Head* predicts state-values $V_\pi^{\gamma_i}$ for n_γ different discount factors. These exponentially discounted state-value predictions are then used to calculate the corresponding advantage estimates and the horizon regularized advantage function.

4.1 Horizon Regularized Advantage

Generalized Advantage Estimate (GAE) [46] or simply advantage denotes the additional expected return that can be achieved by following a particular action compared to the state's absolute value. The advantage is more resistant to overfitting to environment idiosyncrasies than value estimates and less dependent on the number of remaining steps in the episode Raileanu and Fergus [42]. Advantage estimates guide the policy gradient in an actor-critic setup. Thus, we focus on advantage instead of value function.

A single, fixed discount factor in the case of exponential discounting imposes an effective horizon for the agent [28]. As a result, the agent's value function estimate relies on a prior belief about the length of the episode. The episode length can significantly change the value or advantage estimate of the earlier states in a trajectory based on the later reward. For example, the final reward of an episode will be highly discounted and perceived as small if the episode length is too long. However, if the episode length is small, then the same final reward will contribute much more to the advantage estimate. Thus, due to the fixed effective horizon, an agent may fail to correctly anticipate the worth of future rewards in case of highly-varied episode length (as depicted in Figure 1). As we can not restrict the length of an episode, we propose to relax this fixed effective horizon by considering an estimate that arises from multiple horizons. Thus, we need a value estimate that considers multiple discount factors γ_n while calculating the advantage estimate.

We introduce the notion of horizon regularized advantage (HRA) which can be considered a smoothed version of the actual advantage estimate. Formally, we define HRA as a function of multiple advantage estimates over multiple horizons. HRA prevents large policy changes corresponding to a single observed episode length that might destabilize the learning process. As the most simple

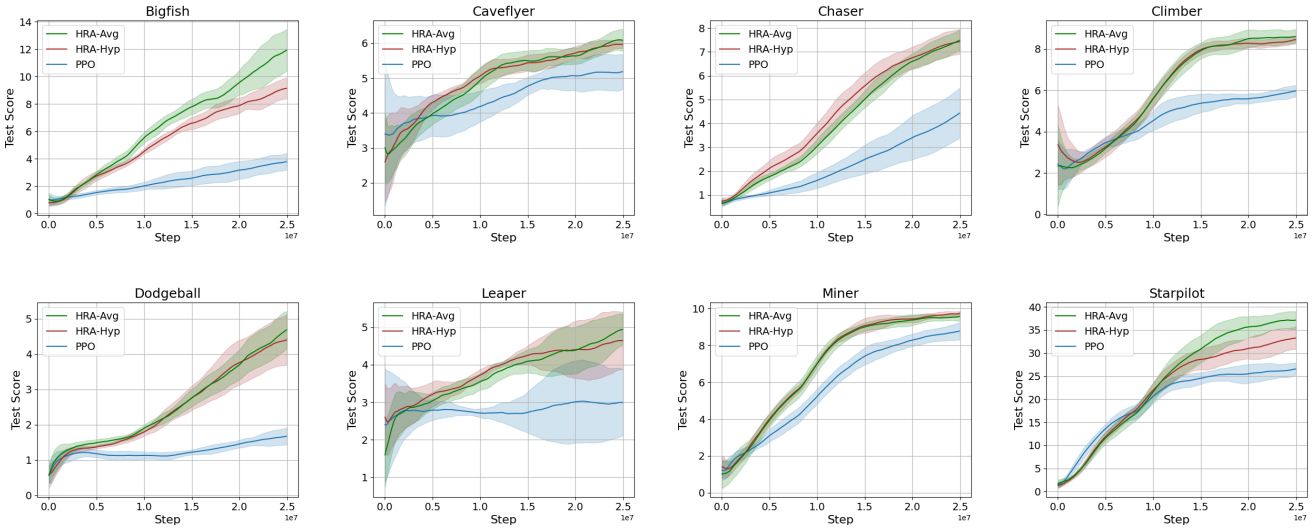

Figure 3: Test performance of proposed average advantage-based horizon regularization (HRA-Avg) and hyperbolic discounting-based horizon regularization (HRA-Hyp) against PPO on Procgen environments.

versions of such HRA, we propose the average of the advantages estimated for different discount factors. If the advantage is a GAE,

$$\hat{A}_{HRA} = \frac{1}{n_\gamma} \sum_{i=0}^{n_\gamma} \hat{A}_{GAE(\gamma_i,\lambda)} \tag{4}$$

where $\hat{A}_{GAE(\gamma_i,\lambda)} = \sum_{l=0}^{\infty} (\gamma_i\lambda)^l \delta_{t+l}^V$ and δ_t^V is the single step TD error. We use this HRA in the policy gradient objective of PPO [47] in Equation 3. Using a multi-head architecture (as shown in Figure 2), where each head corresponds to the value function for each γ_i, we minimize the average of the losses calculated for these multiple γ_i. The loss function corresponding to a γ_i is defined as:

$$L_v^{\gamma_i}(\theta) = \hat{\mathbb{E}}_t \left[\left(V_\theta^{\gamma_i}(s_t) - \hat{V}_{targ}^{\gamma_i} \right)^2 \right] \tag{5}$$

4.2 Regularization through Hyperbolic Approximation

Unseen levels in a procedurally generated environment imply an unknown hazard rate. Consider an episode sampled from a new level m has an associated hazard rate λ_m, where $\gamma_m = e^{-\lambda_m}$ [17]. This hazard doesn't necessarily mean only the chance of dying of an agent, but the variance or uncertainty in the completion time, which is analogous to survival time or how long the agent interacts with the environment, can also be modeled through hazard rate or the corresponding discount factor. Since an agent cannot accurately estimate the hazard or put simply the value of the uncertainty parameter γ_m for each new level in a model-free setup and hyperbolic discounting is better able to capture the uncertainty [17], thus to aid policy optimization we propose to regularize the advantage estimate through hyperbolic-discounting which is a weighted integral of the estimates obtained from multiple discount factors. Advantage is defined as $A(s_t, a_t) = Q(s_t, a_t) - V(s_t)$. Leveraging the hyperbolic function evaluation $\Gamma_k(t) = \frac{1}{1+kt} = \int_0^1 \gamma^{kt} d\gamma$ and motivated by the Q function estimation of [17], we propose to estimate hyperbolically-discounted advantage as follows:

$$A_{HRA}^{\Gamma_k}(s,a) = \int_0^1 A_{GAE}^{(\gamma^k)}(s,a) d\gamma \tag{6}$$

Based on the value function calculated over all the γ^k where $0 \le \gamma < 1$, we estimate the hyperbolically-discounted advantage. Note that the effective discount factor is γ^k, and not just original γ. From a practical perspective, following Fedus et al. [17], we consider a finite set of γ (consequently γ^k), say n_γ number of γs to approximate the advantage through Riemann sum,

$$A_{HRA}^{\Gamma_k}(s,a) \approx \sum_{\gamma_i} (\gamma_{i+1} - \gamma_i)\, A_{GAE}^{(\gamma_i)^k}(s,a) \tag{7}$$

5 EXPERIMENTS AND RESULTS

5.1 Network Architecture and Training

Following previous works, we use the IMPALA-CNN architecture as the actor-critic model for the PPO baseline which employs generalized advantage estimate [10]. Figure 2 shows our implemented architecture. This CNN architecture has three identical blocks, shared by the actor and the critic, and each block has 5 convolutional layers. To implement our proposed approach HRA, we augment the same architecture with five value heads corresponding to five different γ values. We then calculate the advantage value for each of the value predictions. Finally, we take an average of all advantages for the average advantage-based regularization or integrate the advantage values to obtain the hyperbolic advantage. We use ADAM as the optimizer with a learning rate of 0.0005. For PPO, we experiment with two sets of discount factors $\gamma = [0.85, 0.90, 0.95, 0.975, 0.999]$ and $\gamma = [0.90, 0.95, 0.97, 0.98, 0.99]$. The latter demonstrates better performance and we observe using too low values of γ negatively impacts the performance. For APDAC, we use comparatively larger

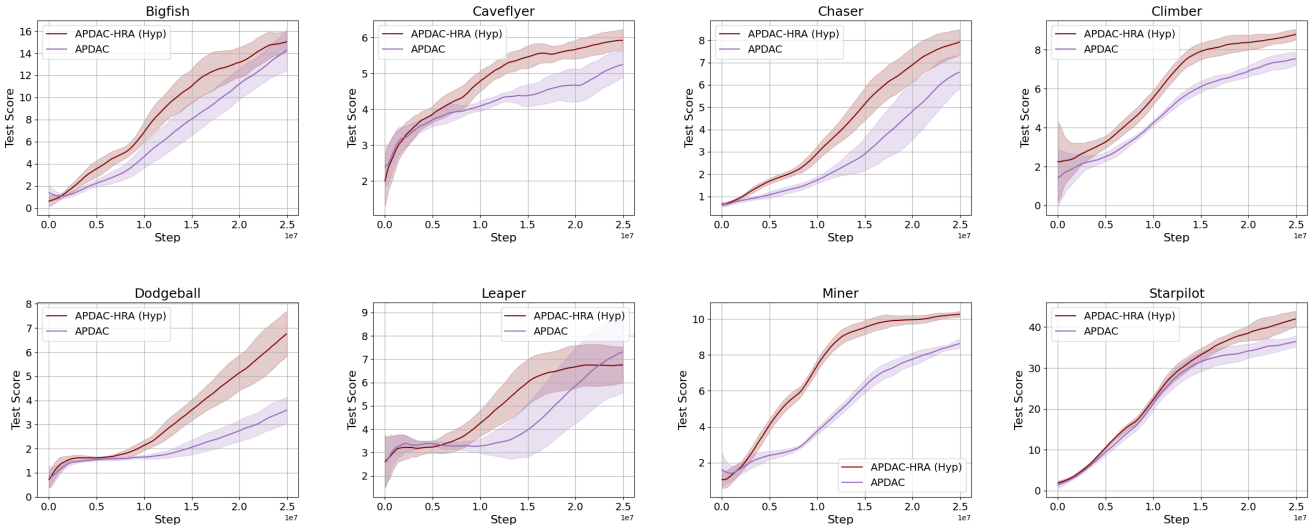

Figure 4: Test performance of APDAC and its counterpart that uses horizon regularized advantage through hyperbolic discounting. Means and standard deviations are calculated over 5 trials.

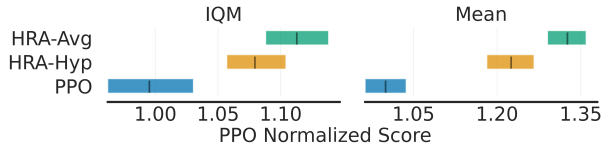

Figure 5: Performance comparison of proposed HRA-Avg and HRA-Hyp with PPO across all 16 environments using IQM and Mean of PPO normalized score.

values for the effective discount factors with $k = 0.025$, while for PPO we use $k = 0.1$. Our code is publicly available.[1]

5.2 Evaluation Benchmarks

We evaluate our approach on three procedurally generated environments including Procgen [10], Crafter [20], and Minigrid [9]. In all the environments, the agent aims to learn the optimal action probability based on the input image observation and obtained reward. Procgen offers an infinite number of diverse procedurally generated levels that make it suitable to investigate the generalization capability of a trained agent. We train the model for 25M time steps with the difficulty mode set to *easy*. Unless mentioned otherwise, we train the agent on 200 levels and test on the full distribution of the levels going beyond the training ones. Crafter offers a highly diverse environment with multiple achievement targets. Crafter is like an open-world survival game where each episode gets configured with a different sequence of resources, terrain types, and creatures, thus requiring generalization and long-term reasoning to perform better in any new episodes. For Minigrid, we train the agent on *Multiroom* and *Fourroom* environments. The agent needs to navigate to the randomly generated goal location from the random initial positions. Thus, the chosen Minigrid environments can create episodes with varied lengths. We train the agent for 1M timesteps for Crafter and Minigrid.

5.3 Generalization Performance on Test Distribution

Figure 3 shows the experimental results on the test distribution of levels for 8 environments from Procgen and presents rolling mean test scores and standard deviations calculated over 5 trials. The results indicate that the proposed two horizon regularized versions of PPO, (i) that uses an average of multiple horizon advantage (HRA-Avg) and (ii) that achieves horizon regularization through hyperbolically discounted advantage (HRA-Hyp), significantly outperform the PPO baseline on the test levels. Since PPO was not specifically designed for generalization, we further incorporated our approach with APDAC [37], a recent generalization-specific approach, to get an understanding of the benefit of our method while integrating it with other existing solutions to generalization. Figure 4 shows that our proposed hyperbolic advantage-based counterpart, APDAC-HRA (Hyp), performs better than APDAC on the test distribution of most environments from Procgen. Figure 6 and 7 show that HRA achieves significantly better performance than PPO on Crafter and Minigrid benchmarks respectively.

6 ANALYSIS AND ABLATIONS

6.1 Analyzing the Computational Overhead

To implement HRA, we need to add n number of different value heads for different value functions corresponding to each γ. We only add one fully connected layer for each γ. Hence, there is an increase in the number of parameters. However, we observe this introduces a very minimal difference in computational cost compared to the

[1] https://github.com/nasiknafi/horizon-regularized-advantage

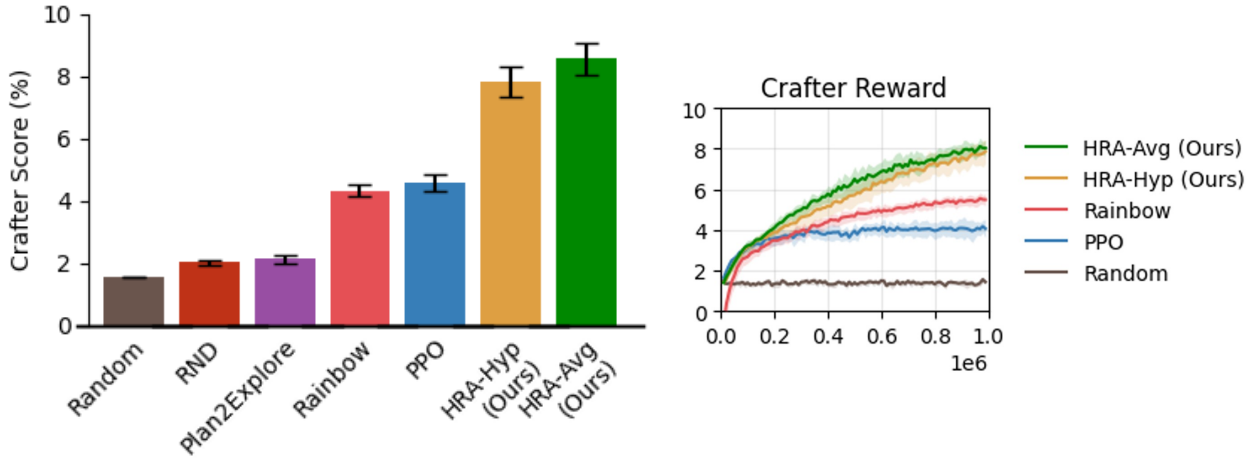

Figure 6: (Left) Crafter score of our HRA-Avg and HRA-Hyp compared to standard PPO and Rainbow along with other approaches; (right) Comparison of the reward achieved by each agent during 1M timestep. In terms of both metrics, both versions of our proposed approach perform significantly better.

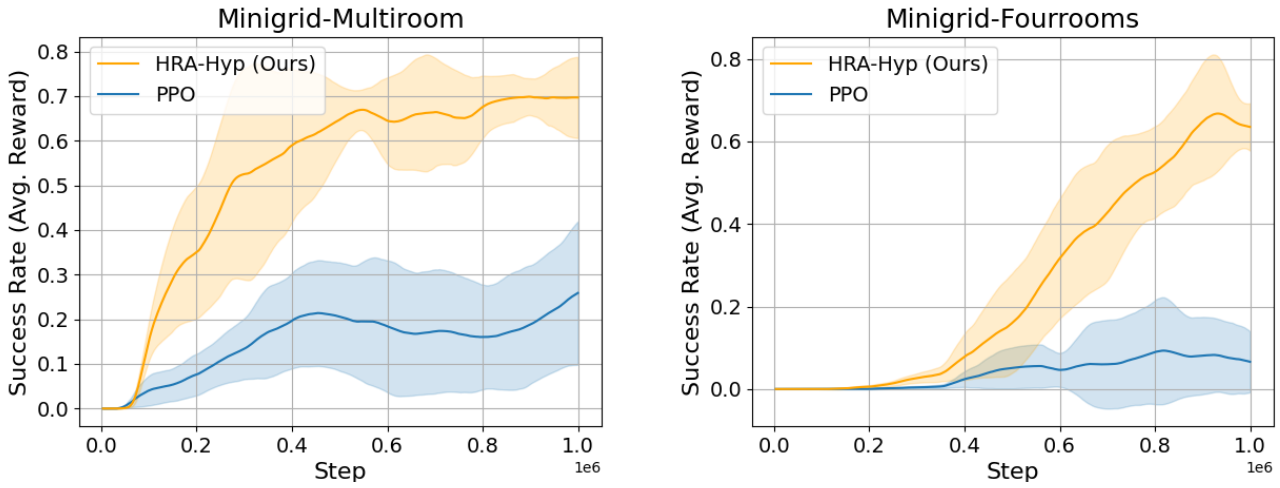

Figure 7: Evaluation of PPO and proposed HRA on two Minigrid environments. HRA significantly outperforms standard PPO.

standard single-head version (PPO/APDAC with a single discount factor). This is mainly because the convolutional layers are all shared across different value heads. We notice an approximately 20%-25% increase in computing time for Procgen environments and a 10% increase in the case of Crafter and Minigrid. Indeed, this is marginal compared to the performance gain because generalization-specific approaches such as [13, 35, 42] require 100%-200% increased computation time compared to basic PPO.

6.2 Comparing Statistical Uncertainty

Figure 5 shows the experimental results on the full distribution of levels in terms of the aggregate metrics considered across all 16 environments as proposed by [2]. Figure 5 indicates that both HRA-Avg and HRA-Hyp achieve higher mean PPO normalized scores. 95% bootstrap confidence intervals do not overlap with each other. We also present Interquartile Mean (IQM) which is a more robust metric than the generic mean considering statistical uncertainty. For IQM, while the confidence intervals for the proposed two regularization methods overlap, they are far apart from the PPO.

6.3 Assessing the Generalization Gap

We compare the train-test performance gap of the baseline PPO with the two horizon regularization methods. Figure 8 presents the train and test performance of all three methods across four

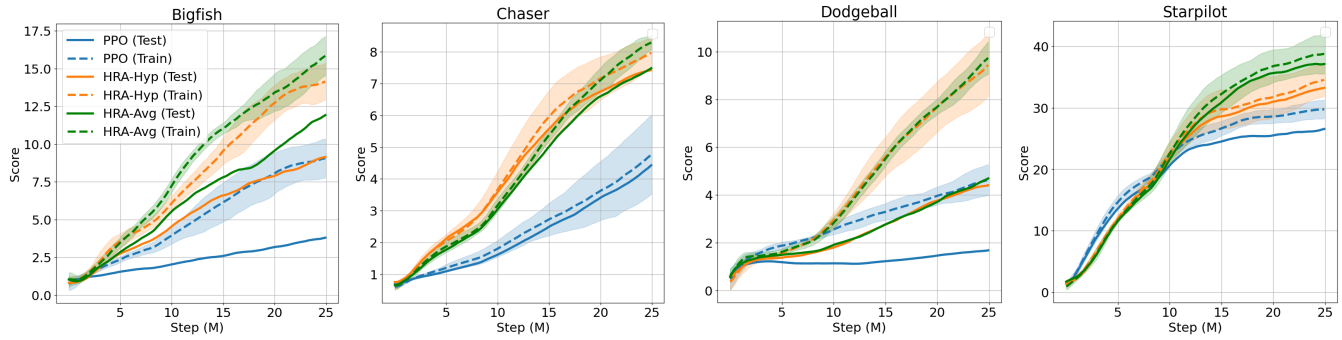

Figure 8: Train and test performance of PPO, HRA-Hyp, and HRA-Avg for four Procgen environments. Means and standard deviations are calculated over 5 trials, each with a different seed.

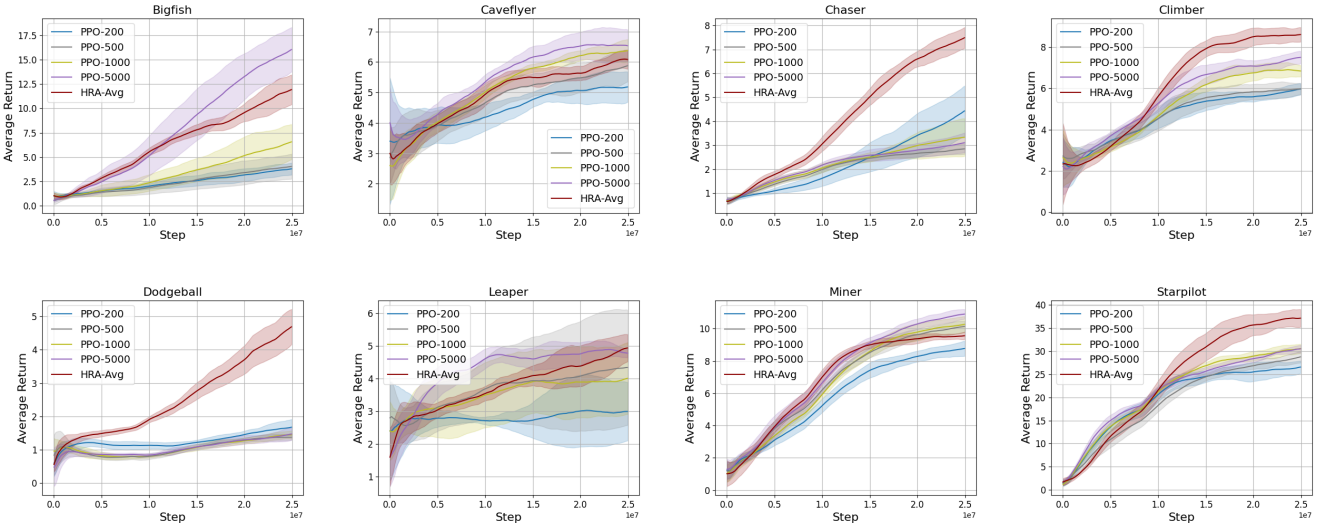

Figure 9: Test rewards of proposed HRA-Avg compared to traditional exponential discounting-based PPO trained on the increased number of levels. HRA-Avg has been trained on only 200 levels.

Procgen games. HRA-Avg and HRA-Hyp outperform the PPO on both train and test level performance. It also shows that our approaches perform competitively with the baseline to reduce the train-test gap in many cases, however, there are few exceptions. Figure 9 shows that, in most of the environments, our proposed method HRA-Avg can achieve test rewards that are higher than the rewards of the PPO trained on more levels (more training levels inherently improve generalization). It is evident that training on only 200 levels, horizon regularization can achieve performance gain that is even higher than PPO trained on 5000 levels of Procgen.

6.4 Analyzing the Auxiliary Task of Multi-horizon Learning

[17] shows the potential of learning over multiple horizons, which serves as an auxiliary task. Figure 10 shows the test performance of our hyperbolic advantage-based approach (HRA-Hyp) against a PPO implementation that learns over multiple horizons e.g. for five different discount factors using exponential reward discounting and it calculates the advantage based on the value corresponding to the largest gamma. From the experimental results, it is evident that learning value functions over multiple horizons as an auxiliary task and estimating the advantage according to only the max gamma (largest gamma) value does not perform well. Even adding such an auxiliary task may degrade the performance. Instead, an advantage value estimated over multiple horizons (e.g. hyperbolically discounted advantage) leads to a more generalizable policy.

6.5 Comparison with Different Gamma

To further analyze the benefit of our proposed horizon regularized advantage estimation in the procedurally generated environments, in Figure 11, we present a comprehensive comparison against multiple PPO models each having a distinct single γ value. We select

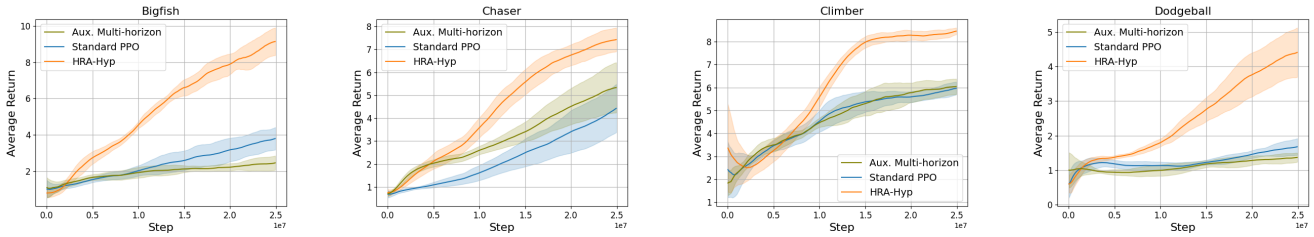

Figure 10: Test performance of the auxiliary task of learning over multiple horizons that calculates the advantage only with the largest gamma vs. our hyperbolic advantage estimation that combines multiple advantage estimates.

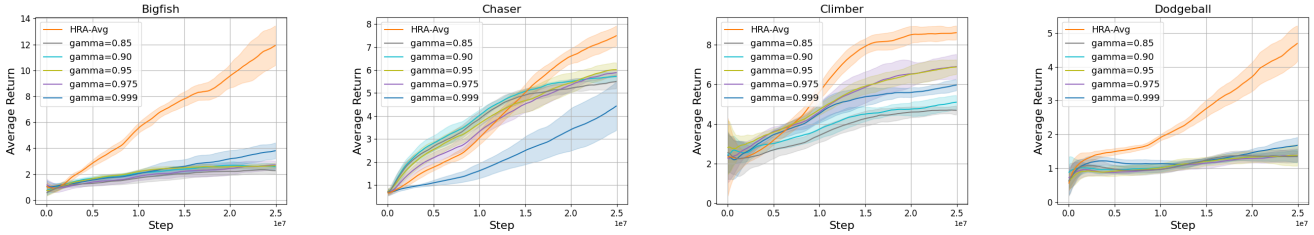

Figure 11: Analysis of test scores for exponential discounting using different discount factors γ vs. horizon regularization.

the gamma values such that they correspond to the gamma values that have been used collectively to approximate the advantage. It is crucial to observe from the figure that there is no common single gamma or discount factor across the environments that perform better in general. This demonstrates that every environment has its hazard distribution and that uncertainty about the hazard cannot be modeled using a single discount factor. Our proposed HRA, leveraging the average of multiple advantage estimates from multiple discount factors, achieves consistently better rewards than any single exponential discounting-based ones. Thus our results are consistent with our claim that estimating the advantage by combining multiple horizons helps the agents to address the uncertainty.

7 CONCLUSION

This work presents an actor-critic method that uses horizon regularized advantage with the specific aim of improving the generalization ability of an agent; and evaluates it for the generalization tasks. We argue that since the task completion time in a procedurally generated environment is more uncertain, having an agent that considers a regularized advantage over multiple discount factors or effective horizon would perform better on unseen levels. Throughout the training, the agent learns the value estimate simultaneously over multiple horizons through the exponential discount factors $\gamma_0, \gamma_1, ..., \gamma_n$, then combines the advantages resulting from those value functions. We present two schemes for horizon regularization-based advantage estimates - simple arithmetic mean of multiple GAE and hyperbolic approximation of GAE from multiple discount factors. The introduction of horizon regularized advantage enables the effective use of multiple discount factors with policy gradient (actor-critic) methods. We evaluate our proposed method using PPO and APDAC, and the results show that the modified agent performs well on most of the tasks from the Procgen benchmark. Further, our approach significantly outperforms the PPO baseline on the Crafter and Minigrid tasks. We plan on extending this work by investigating more complex discounting schemes for generalization.

ACKNOWLEDGMENTS

We would like to thank Ethan Khoury for his help in setting up the experiments for Crafter. We would also like to thank the anonymous reviewers for their feedback to strengthen this work.

REFERENCES

[1] Rishabh Agarwal, Marlos C Machado, Pablo Samuel Castro, and Marc G Bellemare. 2021. Contrastive behavioral similarity embeddings for generalization in reinforcement learning. *arXiv preprint arXiv:2101.05265* (2021).
[2] Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G Bellemare. 2021. Deep Reinforcement Learning at the Edge of the Statistical Precipice. *Advances in Neural Information Processing Systems* (2021).
[3] William H Alexander and Joshua W Brown. 2010. Hyperbolically discounted temporal difference learning. *Neural computation* 22, 6 (2010), 1511–1527.
[4] Ron Amit, Ron Meir, and Kamil Ciosek. 2020. Discount factor as a regularizer in reinforcement learning. In *International conference on machine learning*. PMLR, 269–278.
[5] Marc G Bellemare, Salvatore Candido, Pablo Samuel Castro, Jun Gong, Marlos C Machado, Subhodeep Moitra, Sameera S Ponda, and Ziyu Wang. 2020. Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature* 588, 7836 (2020), 77–82.
[6] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. 2013. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research* 47 (2013), 253–279.
[7] Emmanuel Bengio, Joelle Pineau, and Doina Precup. 2020. Interference and generalization in temporal difference learning. In *International Conference on Machine Learning*. PMLR, 767–777.
[8] David Bertoin and Emmanuel Rachelson. 2022. Local Feature Swapping for Generalization in Reinforcement Learning. In *International Conference on Learning Representations*.

[9] Maxime Chevalier-Boisvert, Lucas Willems, and Suman Pal. 2018. *Minimalistic Gridworld Environment for Gymnasium.* https://github.com/Farama-Foundation/Minigrid

[10] Karl Cobbe, Chris Hesse, Jacob Hilton, and John Schulman. 2020. Leveraging procedural generation to benchmark reinforcement learning. In *International conference on machine learning*. PMLR, 2048–2056.

[11] Karl Cobbe, Oleg Klimov, Chris Hesse, Taehoon Kim, and John Schulman. 2019. Quantifying Generalization in Reinforcement Learning. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 1282–1289. https://proceedings.mlr.press/v97/cobbe19a.html

[12] Karl Cobbe, Oleg Klimov, Chris Hesse, Taehoon Kim, and John Schulman. 2019. Quantifying generalization in reinforcement learning. In *International Conference on Machine Learning*. PMLR, 1282–1289.

[13] Karl W Cobbe, Jacob Hilton, Oleg Klimov, and John Schulman. 2021. Phasic policy gradient. In *International Conference on Machine Learning*. PMLR, 2020–2027.

[14] Partha Dasgupta and Eric Maskin. 2005. Uncertainty and hyperbolic discounting. *American Economic Review* 95, 4 (2005), 1290–1299.

[15] Lasse Espeholt, Hubert Soyer, Remi Munos, Karen Simonyan, Vlad Mnih, Tom Ward, Yotam Doron, Vlad Firoiu, Tim Harley, Iain Dunning, et al. 2018. Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures. In *International Conference on Machine Learning*. PMLR, 1407–1416.

[16] Jesse Farebrother, Marlos C Machado, and Michael Bowling. 2018. Generalization and regularization in DQN. *arXiv preprint arXiv:1810.00123* (2018).

[17] William Fedus, Carles Gelada, Yoshua Bengio, Marc G Bellemare, and Hugo Larochelle. 2019. Hyperbolic discounting and learning over multiple horizons. *arXiv preprint arXiv:1902.06865* (2019).

[18] Jake Grigsby and Yanjun Qi. 2020. Measuring Visual Generalization in Continuous Control from Pixels. *CoRR* abs/2010.06740 (2020). arXiv:2010.06740 https://arxiv.org/abs/2010.06740

[19] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*. PMLR, 1861–1870.

[20] Danijar Hafner. 2021. Benchmarking the spectrum of agent capabilities. *arXiv preprint arXiv:2109.06780* (2021).

[21] Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. 2018. Rainbow: Combining improvements in deep reinforcement learning. In *Thirty-second AAAI conference on artificial intelligence*.

[22] Tianyang Hu, Wenjia Wang, Cong Lin, and Guang Cheng. 2021. Regularization Matters: A Nonparametric Perspective on Overparametrized Neural Network. In *International Conference on Artificial Intelligence and Statistics*. PMLR, 829–837.

[23] Maximilian Igl, Kamil Ciosek, Yingzhen Li, Sebastian Tschiatschek, Cheng Zhang, Sam Devlin, and Katja Hofmann. 2019. Generalization in reinforcement learning with selective noise injection and information bottleneck. *Advances in neural information processing systems* 32 (2019).

[24] Maximilian Igl, Gregory Farquhar, Jelena Luketina, Wendelin Boehmer, and Shimon Whiteson. 2020. The impact of non-stationarity on generalisation in deep reinforcement learning. *arXiv preprint arXiv:2006.05826* (2020).

[25] Zhiwei Jia, Xuanlin Li, Zhan Ling, Shuang Liu, Yiran Wu, and Hao Su. 2022. Improving Policy Optimization with Generalist-Specialist Learning. In *International Conference on Machine Learning*. PMLR, 10104–10119.

[26] Yiding Jiang, J Zico Kolter, and Roberta Raileanu. 2024. On the importance of exploration for generalization in reinforcement learning. *Advances in Neural Information Processing Systems* 36 (2024).

[27] Niels Justesen, Ruben Rodriguez Torrado, Philip Bontrager, Ahmed Khalifa, Julian Togelius, and Sebastian Risi. 2018. Illuminating generalization in deep reinforcement learning through procedural level generation. *arXiv preprint arXiv:1806.10729* (2018).

[28] Michael Kearns and Satinder Singh. 2002. Near-optimal reinforcement learning in polynomial time. *Machine learning* 49, 2 (2002), 209–232.

[29] Zeb Kurth-Nelson and A David Redish. 2009. Temporal-difference reinforcement learning with distributed representations. *PLoS One* 4, 10 (2009), e7362.

[30] Clare Lyle, Mark Rowland, Will Dabney, Marta Kwiatkowska, and Yarin Gal. 2022. Learning dynamics and generalization in deep reinforcement learning. In *International Conference on Machine Learning*. PMLR, 14560–14581.

[31] Debmalya Mandal, Goran Radanovic, Jiarui Gan, Adish Singla, and Rupak Majumdar. 2023. Online Reinforcement Learning with Uncertain Episode Lengths. In *37th AAAI Conference on Artificial Intelligence*. AAAI.

[32] Bogdan Mazoure, Remi Tachet des Combes, Thang Long Doan, Philip Bachman, and R Devon Hjelm. 2020. Deep reinforcement and infomax learning. *Advances in Neural Information Processing Systems* 33 (2020), 3686–3698.

[33] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. 2016. Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*. PMLR, 1928–1937.

[34] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. 2015. Human-level control through deep reinforcement learning. *nature* 518, 7540 (2015), 529–533.

[35] Seungyong Moon, JunYeong Lee, and Hyun Oh Song. 2022. Rethinking Value Function Learning for Generalization in Reinforcement Learning. *arXiv preprint arXiv:2210.09960* (2022).

[36] Nasik Muhammad Nafi, Raja Farrukh Ali, and William Hsu. 2023. Analyzing the Sensitivity to Policy-Value Decoupling in Deep Reinforcement Learning Generalization. In *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems* (London, United Kingdom) *(AAMAS '23)*. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 2625–2627.

[37] Nasik Muhammad Nafi, Creighton Glasscock, and William Hsu. 2022. Attention-based Partial Decoupling of Policy and Value for Generalization in Reinforcement Learning. In *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*. 15–22. https://doi.org/10.1109/ICMLA55696.2022.00011

[38] Nasik Muhammad Nafi, Giovanni Poggi-Corradini, and William Hsu. 2023. Policy Optimization with Augmented Value Targets for Generalization in Reinforcement Learning. In *2023 International Joint Conference on Neural Networks (IJCNN)*. 1–8. https://doi.org/10.1109/IJCNN54540.2023.10191507

[39] Alex Nichol, Vicki Pfau, Christopher Hesse, Oleg Klimov, and John Schulman. 2018. Gotta Learn Fast: A New Benchmark for Generalization in RL. arXiv:1804.03720 [cs.LG]

[40] Charles Packer, Katelyn Gao, Jernej Kos, Philipp Krähenbühl, Vladlen Koltun, and Dawn Song. 2018. Assessing generalization in deep reinforcement learning. *arXiv preprint arXiv:1810.12282* (2018).

[41] Fabian Paischer, Thomas Adler, Vihang Patil, Angela Bitto-Nemling, Markus Holzleitner, Sebastian Lehner, Hamid Eghbal-Zadeh, and Sepp Hochreiter. 2022. History compression via language models in reinforcement learning. In *International Conference on Machine Learning*. PMLR, 17156–17185.

[42] Roberta Raileanu and Rob Fergus. 2021. Decoupling value and policy for generalization in reinforcement learning. In *International Conference on Machine Learning*. PMLR, 8787–8798.

[43] Roberta Raileanu, Max Goldstein, Denis Yarats, Ilya Kostrikov, and Rob Fergus. 2020. Automatic data augmentation for generalization in deep reinforcement learning. *arXiv preprint arXiv:2006.12862* (2020).

[44] Aravind Rajeswaran, Kendall Lowrey, Emanuel Todorov, and Sham Kakade. 2017. Towards generalization and simplicity in continuous control. *arXiv preprint arXiv:1703.02660* (2017).

[45] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. 2015. Trust region policy optimization. In *International conference on machine learning*. PMLR, 1889–1897.

[46] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. 2015. High-dimensional continuous control using generalized advantage estimation. *arXiv preprint arXiv:1506.02438* (2015).

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

[48] Peter D Sozou. 1998. On hyperbolic discounting and uncertain hazard rates. *Proceedings of the Royal Society of London. Series B: Biological Sciences* 265, 1409 (1998), 2015–2020.

[49] Richard S. Sutton and Andrew G. Barto. 1998. *Reinforcement Learning: An Introduction.* MIT press Cambridge.

[50] Vladimir Vapnik. 1991. Principles of risk minimization for learning theory. *Advances in neural information processing systems* 4 (1991).

[51] Kaixin Wang, Bingyi Kang, Jie Shao, and Jiashi Feng. 2020. Improving generalization in reinforcement learning with mixture regularization. *Advances in Neural Information Processing Systems* 33 (2020), 7968–7978.

[52] Denis Yarats, Ilya Kostrikov, and Rob Fergus. 2020. Image augmentation is all you need: Regularizing deep reinforcement learning from pixels. In *International Conference on Learning Representations*.

[53] Tianhe Yu, Deirdre Quillen, Zhanpeng He, Ryan Julian, Karol Hausman, Chelsea Finn, and Sergey Levine. 2020. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Conference on robot learning*. PMLR, 1094–1100.

[54] Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. 2020. Learning invariant representations for reinforcement learning without reconstruction. *arXiv preprint arXiv:2006.10742* (2020).

[55] Hanping Zhang and Yuhong Guo. 2021. Generalization of reinforcement learning with policy-aware adversarial data augmentation. *arXiv preprint arXiv:2106.15587* (2021).