Exploring through Random Curiosity with General Value Functions

Aditya Ramesh[1] **Louis Kirsch**[1] **Sjoerd van Steenkiste**[1] **Jürgen Schmidhuber**[1,2]
[1]The Swiss AI Lab, IDSIA, University of Lugano (USI) & SUPSI, Lugano, Switzerland
[2]King Abdullah University of Science and Technology (KAUST), Thuwal, Saudi Arabia
{aditya, louis, sjoerd, juergen}@idsia.ch

Abstract

Exploration in reinforcement learning through intrinsic rewards has previously been addressed by approaches based on state novelty or artificial curiosity. In partially observable settings where observations look alike, state novelty can lead to intrinsic reward vanishing prematurely. On the other hand, curiosity-based approaches require modeling precise environment dynamics which are potentially quite complex. Here we propose random curiosity with general value functions (RC-GVF), an intrinsic reward function that connects state novelty and artificial curiosity. Instead of predicting the entire environment dynamics, RC-GVF predicts temporally extended values through general value functions (GVFs) and uses the prediction error as an intrinsic reward. In this way, our approach generalizes a popular approach called random network distillation (RND) by encouraging behavioral diversity and reduces the need for additional maximum entropy regularization. Our experiments on four procedurally generated partially observable environments indicate that our approach is competitive to RND and could be beneficial in environments that require behavioural exploration.

1 Introduction

Efficient exploration in reinforcement learning (RL) is a challenging problem, especially in high-dimensional observation spaces and sparse-reward environments [Thrun, 1992, Bellemare et al., 2016]. A common strategy to address this is by incorporating an *intrinsic* reward, in addition to the (sparse) extrinsic reward.

In the recent literature two main approaches have emerged: (1) approaches based on *state novelty*, where an intrinsic reward in the form of a 'novelty bonus' is awarded based on how often a state has been visited [Sutton, 1990, Barto and Singh, 1991, Bellemare et al., 2016, Burda et al., 2019a]; and (2) approaches based on *artificial curiosity*, where agents are rewarded based on the prediction error or information gain of a world model [Schmidhuber, 1991, Storck et al., 1995, Houthooft et al., 2016, Pathak et al., 2017].

A limitation of state novelty bonuses in partially observable settings is that observations may look alike. For example in random network distillation (RND) the agent simply receives a novelty bonus based on the error in making predictions about the observation after applying a fixed randomly initialized neural network. The success of the RND novelty bonus would therefore rely on the generalization properties of a (random) neural network's mapping of observations to features. This can be especially problematic if the error (and thus the intrinsic reward) vanishes before the agent has reliably discovered the source of extrinsic rewards in the environment [Raileanu and Rocktäschel, 2020, Flet-Berliac et al., 2021]. Similarly, curiosity-based approaches usually require modeling the complete environment dynamics, which are potentially very complex.

35th Deep RL workshop, Conference on Neural Information Processing Systems (NeurIPS 2021), Sydney, Australia.

In this paper we explore a connection between state novelty and artificial curiosity that may address both limitations. To that extent we propose *random curiosity with general value functions* (RC-GVF), a novel approach to generating intrinsic rewards based on general value functions (GVFs) [Sutton et al., 2011] and the prediction of abstract quantities about the environment [Schmidhuber, 1997]. We view GVFs as partially modeling the environment dynamics through predicting the temporally extended value of a quantity of interest (i.e. 'answering' a question) in the environment when following a given policy [Sutton, 1995]. This quantity of interest corresponds to a random observation-dependent function of pseudo-rewards, akin to the random target features of RND. We then minimize the TD-error of these GVFs while using the error as an intrinsic reward. We argue that this is similar to the intrinsic reward derived from curiosity-based approaches, but now only modeling a random subset of the environment dynamics.

RC-GVF includes RND as a special case where the discount factor of the GVF is set to zero. Unlike in RND, where predictions about random target features may be viewed as predicting pseudo-rewards, RC-GVF takes longer horizon prediction errors into account that might be better suited for exploration. In particular, by integrating the policy as part of the prediction problem, we reward the agent for altering its behavior, reducing the need for additional exploration mechanisms such as maximum entropy regularization that is typically used in the case of RND.

We evaluate RC-GVF on a benchmark of sparse reward partially observable environments. Compared to RND we observe improvements in mean return in the absence of entropy regularization. Introspection reveals that our approach reaches states that provide external reward more frequently when learning only from intrinsic rewards in environments that benefit from behavioral diversity. In environments that require visiting most states, performance is similar to RND.

2 Preliminaries

Reinforcement Learning We follow the standard POMDP formulation with time steps $t \in \mathbb{N}$, observations $o_t \in \mathcal{O}$, environment states $s_t \in \mathcal{S}$, actions $a_t \in \mathcal{A}$, extrinsic rewards $R_e(s_t)$, and policies $\pi(a_t|h_t)$ where $h_t = o_{1:t}$. The objective is to find the optimal policy π^* that maximizes the expected discounted return

$$J(\pi) = \mathbb{E}_\pi[\sum_{k=0}^{\infty} \gamma^k R_e(S_k)], \tag{1}$$

where $0 < \gamma < 1$ is the discount factor and upper case variables denote random variables.

Random Network Distillation In random network distillation (RND) the agent receives a state novelty reward proportional to the error of predicting features $\hat{z}(o_t)$ generated by a randomly initialized neural network $Z_\phi : \mathcal{O} \to \mathbb{R}^d$. The RND intrinsic reward for an observation is given by

$$R_i(o_t) = \|Z_\phi\left(o_t\right) - \hat{z}\left(o_t\right)\|_2. \tag{2}$$

Entropy Regularization To encourage additional exploration, many RL algorithms employ entropy regularization. The maximum entropy objective [Haarnoja et al., 2017] adjusts the RL objective from Equation 1 to

$$J_{\text{MaxEnt}}(\pi) = \mathbb{E}_\pi[\sum_{k=0}^{\infty} \gamma^k R_e(S_k) + \alpha\mathcal{H}(\pi(\cdot|H_k))], \tag{3}$$

where $\alpha \in \mathbb{R}_+$ is a hyper-parameter that trades off rewards and entropy regularization. RND also typically employs this regularization, despite introducing its own exploration mechanism.

General Value Functions A general value function (GVF) [Sutton et al., 2011, Schaul and Ring, 2013] is defined by a policy π, a cumulant or pseudo-reward function $Z : \mathcal{O} \to \mathbb{R}$, and a discount factor γ_z. It can be expressed as

$$v_{\pi,z}(o) = \mathbb{E}_\pi\left[\sum_{k=0}^{\infty} \gamma_z^k Z(O_{t+k})|O_t = o\right]. \tag{4}$$

General value functions extend the concept of predicting expected cumulative values to arbitrary signals beyond the reward. They can be viewed as answers to questions about such quantities under a particular policy.

3 Random Curiosity with General Value Functions

This section describes *random curiosity with general value functions* (RC-GVF), a novel approach to intrinsic rewards based on GVFs of random pseudo rewards.

Random Pseudo-Rewards and GVFs Similar to artificial curiosity [Schmidhuber, 1991] in RC-GVF we model the environment dynamics. Instead of modeling the entire dynamics, we ask questions about outcomes in the future under a policy π [Schmidhuber, 1997]. In this paper, these questions are random and represented by a randomly initialized neural network Z_ϕ that maps observations to features $Z_\phi : \mathcal{O} \to \mathbb{R}^d$, here referred to as pseudo-rewards. At time step t, the current observation o_t is mapped to the pseudo-rewards $z_{t+1} \in \mathbb{R}^d$. To capture the outcome of a question across time, we are interested in the discounted pseudo-return G_t^z for a sequence of pseudo-reward random variables $Z_{t+1}, Z_{t+2} \ldots$ given by

$$G_t^z = \sum_{k=0}^{\infty} \gamma_z^k \cdot Z_{t+k+1}, \tag{5}$$

where γ_z is a scalar discount factor. Pseudo-rewards and returns are random variables due to the stochasticity in the policy and/or environment dynamics. Similarly, the pseudo-value(s) of an observation under the policy π is given by

$$v_{\pi,z}(o) = \mathbb{E}_\pi \left[\sum_{k=0}^{\infty} \gamma_z^k Z_\phi(O_{t+k}) \mid O_t = o \right]. \tag{6}$$

This value function is known as a *general value function* [Sutton et al., 2011].

TD-Error as Intrinsic Curiosity Reward Our exploration mechanism rewards the agent for taking actions that generate unknown outcomes under fixed random questions. To that end, we train a separate (recurrent) neural network, which we call the *predictor*, to predict these pseudo-values. Concretely, the predictor $\hat{v}_{\pi,z} : \mathcal{H} \to \mathbb{R}^d$ maps histories of observations $\mathcal{H}$ to values. We focus on a predictor that is trained on-policy. One motivating factor for this is that it couples the prediction task to the current policy, which creates an incentive to vary the policy for additional exploration. As a target we use the (truncated) λ-return, which can be recursively expressed as

$$G_t^z(\lambda) = Z_{t+1} + \gamma_z(1 - \lambda_z)\hat{v}_{\pi,z}(H_{t+1}) + \gamma_z \lambda G_{t+1}^z. \tag{7}$$

Here, $\lambda_z \in [0, 1]$ is the usual parameter that allows us to balance the bias-variance trade off by interpolating between TD(0) and Monte Carlo estimates of the pseudo-return [Sutton, 1988]. The intrinsic reward of RC-GVF at time step t is then defined as the error between the output of the predictor and the truncated λ-pseudo-return

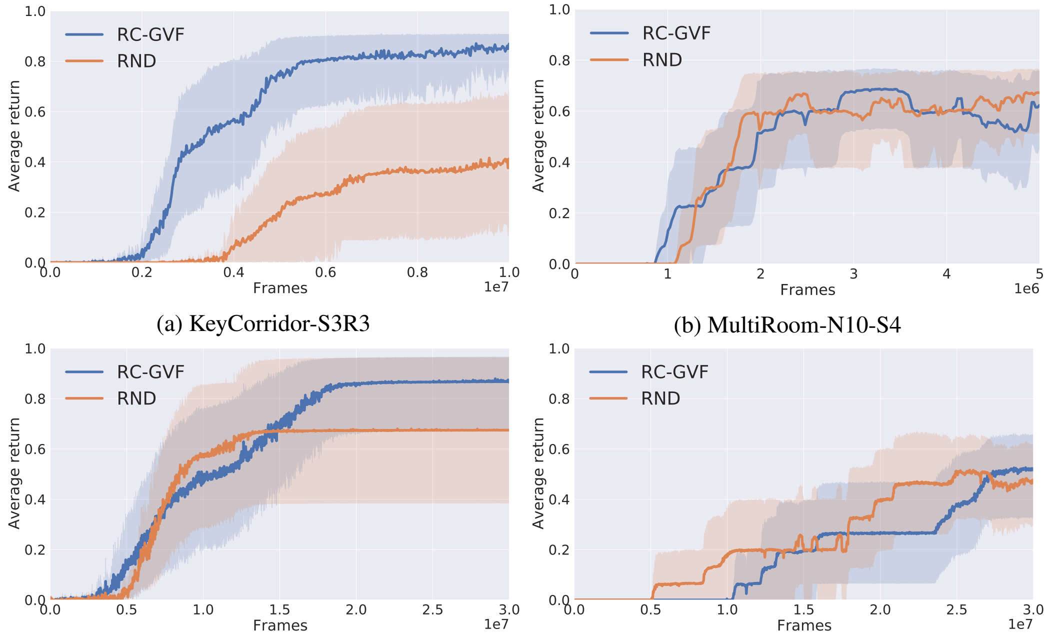
$$R_i(o_t) = \|G_t^z(\lambda) - \hat{v}_{\pi,z}(h_t)\|_2, \tag{8}$$

where $h_t = o_{1:t}$. Using $G_t^z(\lambda)$ as the target links the prediction task and the intrinsic reward to the agent's policy, thereby encouraging *behavioral exploration*.

Effective Horizon and RND as a Special Case The effective horizon over which predictions are considered depends on the choice of the discount factor γ_z. We obtain an intrinsic reward matching RND (Equation 2) for the special case of $\gamma_z = 0$, which yields $G_t^z = Z_{t+1}$. The larger γ_z, the more pronounced is the contribution of the current policy to the prediction problem.

4 Experiments

We compare RC-GVF and RND on four procedurally generated environments from MiniGrid [Chevalier-Boisvert et al., 2018]: KeyCorridor-S3R3, ObstructedMaze-2Dl, MultiRoom-N7-S8, and MultiRoom-N10-S4. Exploration in these environments is particularly challenging due to partial



(a) KeyCorridor-S3R3

(b) MultiRoom-N10-S4

(c) ObstructedMaze-2Dl

(d) MultiRoom-N7-S8

Figure 1: Average return of RC-GVF and RND on the selected Minigrid environments (with no entropy regularization).

observability, sparse rewards, and the procedural generation of mazes and objects. Additional details about the environments are available in Appendix A.

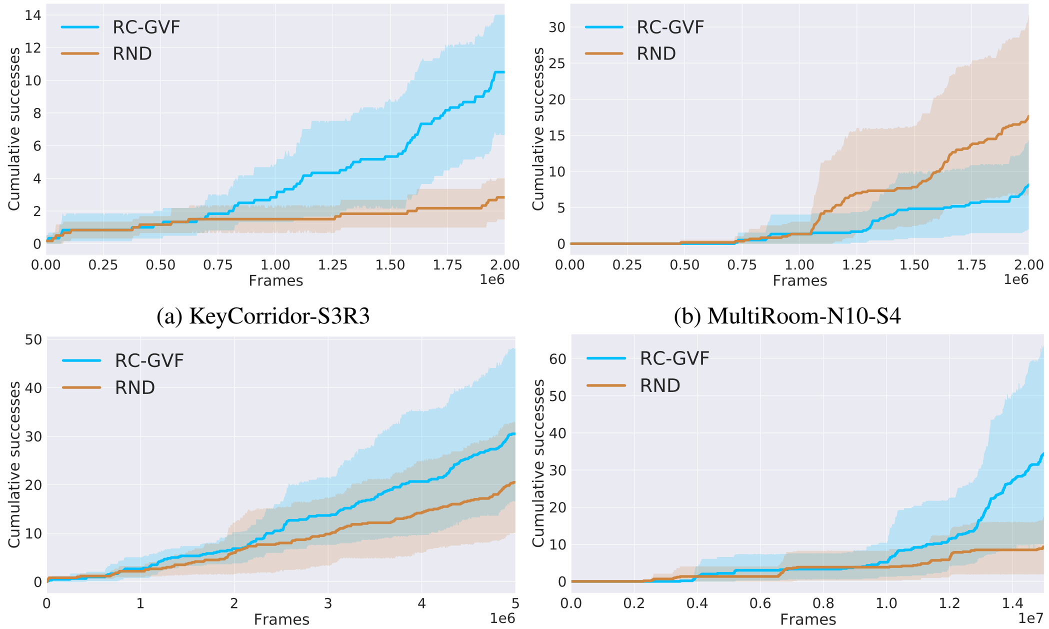
To avoid confounding factors from combining different exploration strategies, we mainly focus on the exploration behaviour in the absence of entropy regularization. For comparison with previous works, we also include an analysis with entropy regularization. Moreover, to study the benefit of RC-GVF in isolation, we also consider a setting where the agent only receives intrinsic rewards to guide its behaviour. Finally, we conduct an experiment to study the influence of the RC-GVF discount factor (γ_z) on the agent's performance.

Implementation We use Proximal Policy Optimization (PPO) [Schulman et al., 2017] as our base agent. This agent is trained to maximize the expected sum of a weighted combination of intrinsic and extrinsic rewards. At each time step t, the agent receives a reward $R_t = R_e(s_t) + \beta R_i(o_t)$. Here $R_e(s_t)$ is the extrinsic reward from the environment, $R_i(o_t)$ is the intrinsic reward generated from either RND or RC-GVF, and $\beta \in \mathbb{R}_+$ is a hyperparameter to balance the weighting of intrinsic and extrinsic rewards.

The agent consists of an actor and a critic, both share convolution layers followed by an LSTM [Hochreiter and Schmidhuber, 1997], with separate multi-layer perceptron (MLP) heads for the actor and critic. The pseudo-reward generator is implemented as a convolutional neural network whose output is flattened to a vector. As per the original implementation of RND [Burda et al., 2019a], the predictor has the same architecture as that of the pseudo-reward generator. In the case of RC-GVF, the predictor is recurrent, consisting of three convolutional layers followed by an LSTM with a linear output layer. More details about the neural architectures and implementation are available in Appendix B.

Evaluation We present the results averaged over 10 independent runs for RC-GVF and RND on each problem. The best hyper-parameter configuration for each approach and environment was identified by a grid search with three seeds per configuration (see Appendix C).

Unless mentioned otherwise, all figures report the mean as a solid line and the shading indicates 95% bootstrapped confidence intervals for the 10 seeds.



(a) KeyCorridor-S3R3

(b) MultiRoom-N10-S4

(c) ObstructedMaze-2Dl

(d) MultiRoom-N7-S8

Figure 2: Cumulative number of times the agent successfully solves the environment, while only receiving intrinsic rewards. Shading indicates 95% bootstrapped confidence intervals over 6 seeds.

Results Figure 1 presents our results on the MiniGrid environments when no entropy regularization is used. We observe that RC-GVF appears more sample efficient than RND on the KeyCorridor environment. In the ObstructedMaze environment we see that our approach is more stable across independent runs. Both approaches are comparable and slightly unstable in the Multi-Room environments (N10-S4 and N7-S8). We note that in a previous empirical comparison with IMPALA [Espeholt et al., 2018] instead of PPO as the base agent (i.e. as in this paper), RND was not able to reach any extrinsic reward states on these MultiRoom tasks [Raileanu and Rocktäschel, 2020].

One explanation for the findings in Figure 1 could be that the tasks in the Multi-Room environments are inherently suited to RND's intrinsic reward of covering as many states as possible. The extrinsic reward in these environments requires visiting most states. Similarly, the better performance of RC-GVF in KeyCorridor and ObstructedMaze could be due to a greater need to try out *different* behaviours (finding the key, unlocking doors, picking up objects). Indeed, this variation in behaviors is encouraged by the policy dependence in RC-GVF.

Figure 2 presents an analysis of the number of episodes in which the agent successfully completes a task (i.e. it receives any extrinsic reward). To measure the exploration effect independent of extrinsic reward maximization, in this setting the agent receives only intrinsic rewards to guide its behaviour. It can be observed how RC-GVF completes the task more often on the KeyCorridor and ObstructedMaze environments, again indicating that RC-GVF could be better suited to explore the kinds of environments which benefit from policy diversity and higher entropy.

In the MultiRoom environments, we observe mixed results. It can be seen how an agent trained solely with the RND bonus reaches the goal state more frequently in the N10-S4 environment as compared to an agent trained with our bonus. We believe that the reason behind the reduced performance of RC-GVF is tied to the exploration mechanism driven by altering the policy. In these environments the goal state (with non-zero rewards) has a visually unique appearance (see Figure 7) that potentially produces large intrinsic rewards in the case of RND, whereas RC-GVF is also incentivized by *behavioral* exploration, reducing the tendency to follow the precise policy to revisit that state.

Additional Analysis We carry out an experiment to study the influence of the GVF discount factor γ_z on RC-GVF's performance in the KeyCorridor-S3R3 environment. Here γ_z can be viewed as interpolating between RND ($\gamma_z = 0$) and variations of RC-GVF with increasingly more emphasis

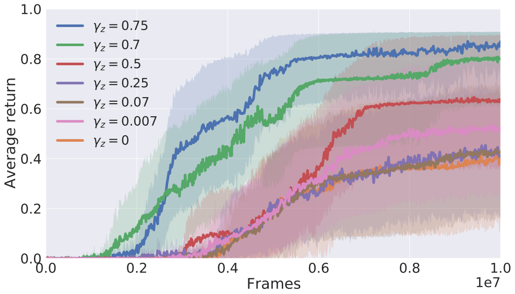


Figure 3: Performance of RC-GVF with different discount factors (γ_z) on the KeyCorridor-S3R3 environment (with no entropy regularization).

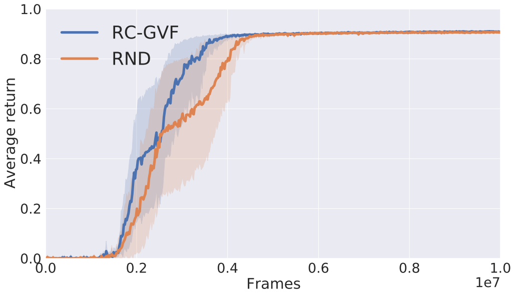


Figure 4: We see that with the entropy regularization, both methods improve and are comparable on the KeyCorridor-S3R3 environment.

on longer horizon predictions. Since the choice of GVF discount factor impacts the magnitude of the pseudo-returns, the best intrinsic reward coefficient (β) was selected separately for each value of γ_z through a grid search (see Appendix C). Figure 3 presents the results of this experiment. We observe that the lower discount factors $\gamma_z \in \{0.007, 0.07, 0.25\}$ generally perform similarly to RND (shown as $\gamma_z = 0$), with only $\gamma_z = 0.007$ performing slightly better than expected. Increasing the GVF discount factor–and therefore the horizon considered for the value function predictions–appears to have a positive impact on the agent's performance in this setting.

In the previous experiments we have focused on settings without entropy regularization, to avoid confounding multiple exploration mechanisms. To compare with previous works, we conduct an experiment on KeyCorridor-S3R3 with entropy regularization. In Figure 4, we see that the performance of both approaches improves with the introduction of the entropy term. RND's improvement is more pronounced in comparison to RC-GVF. This suggests that RC-GVF requires less entropy regularization due to promoted behavioral diversity.

5 Conclusion

We developed an intrinsic reward approach to exploration inspired by ideas from state novelty bonuses and artificial curiosity. We introduced RC-GVF, based on general value functions, which derives intrinsic rewards from the long term prediction error of random questions under the current policy. The discount factor of these general value functions allows us to control the horizon over which predictions are considered. Our approach includes RND as a special case when the discount factor is zero.

Our experiments on four procedurally generated partially observable environments indicate that our approach could be beneficial in environments that require behavioural exploration.

Limitations While the incorporation of the current policy into the prediction task can have benefits in behavioral exploration, it can also lead to over-exploration. Alternatives to this on-policy version of RC-GVF include an off-policy variant or policy-conditioned GVFs [Harb et al., 2020, Faccio et al., 2021]. Another potential improvement can come from a distributional perspective to obtain intrinsic rewards. This may be important as RC-GVF does not account for the inherent variance in the pseudo-return even for a fixed policy. Furthermore, in comparison to RND, where the predictor belongs to same model class as the pseudo-reward generator, we need to select a predictor of appropriate complexity.

Future work In the future, we aim to compare RC-GVF with transition dynamics based curiosity approaches [Burda et al., 2019b]. Further generalizations are possible; such as moving beyond random pseudo-rewards, general value functions under different policies, and introducing time or state dependent discounting.

Acknowledgements

We would like to thank Kenny Young, Francesco Faccio, and Anand Gopalakrishnan for their valuable comments. This research was supported by the ERC Advanced Grant (742870), the Swiss National Science Foundation grant (200021_192356), and by the Swiss National Supercomputing Centre (CSCS project s1090).

References

Sebastian B Thrun. Efficient exploration in reinforcement learning. 1992.

Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. *Advances in neural information processing systems*, 29:1471–1479, 2016.

Richard S Sutton. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Machine learning proceedings 1990*, pages 216–224. Elsevier, 1990.

Andrew G Barto and Satinder Pal Singh. On the computational economics of reinforcement learning. In *Connectionist Models*, pages 35–44. Elsevier, 1991.

Yuri Burda, Harrison Edwards, Amos J. Storkey, and Oleg Klimov. Exploration by random network distillation. In *7th International Conference on Learning Representations*, 2019a.

Jürgen Schmidhuber. A possibility for implementing curiosity and boredom in model-building neural controllers. In *Proc. of the international conference on simulation of adaptive behavior: From animals to animats*, pages 222–227, 1991.

Jan Storck, Sepp Hochreiter, Jürgen Schmidhuber, et al. Reinforcement driven information acquisition in non-deterministic environments. In *Proceedings of the international conference on artificial neural networks, Paris*, volume 2, pages 159–164. Citeseer, 1995.

Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. *Advances in Neural Information Processing Systems*, 29:1109–1117, 2016.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR, 2017.

Roberta Raileanu and Tim Rocktäschel. RIDE: rewarding impact-driven exploration for procedurally-generated environments. In *8th International Conference on Learning Representations*, 2020.

Yannis Flet-Berliac, Johan Ferret, Olivier Pietquin, Philippe Preux, and Matthieu Geist. Adversarially guided actor-critic. In *9th International Conference on Learning Representations*, 2021.

Richard S. Sutton, Joseph Modayil, Michael Delp, Thomas Degris, Patrick M. Pilarski, Adam White, and Doina Precup. Horde: a scalable real-time architecture for learning knowledge from unsupervised sensorimotor interaction. In Liz Sonenberg, Peter Stone, Kagan Tumer, and Pinar Yolum, editors, *10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2011), Taipei, Taiwan, May 2-6, 2011, Volume 1-3*, pages 761–768, 2011.

J. Schmidhuber. What's interesting? Technical Report IDSIA-35-97, IDSIA, 1997. ftp://ftp.idsia.ch/pub/juergen/interest.ps.gz; extended abstract in Proc. Snowbird'98, Utah, 1998; see also Schmidhuber [2002].

Richard S Sutton. Td models: Modeling the world at a mixture of time scales. In *Machine Learning Proceedings 1995*, pages 531–539. Elsevier, 1995.

Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *International Conference on Machine Learning*, pages 1352–1361. PMLR, 2017.

Tom Schaul and Mark Ring. Better generalization with forecasts. In *Twenty-Third International Joint Conference on Artificial Intelligence*, 2013.

Richard S Sutton. Learning to predict by the methods of temporal differences. *Machine learning*, 3 (1):9–44, 1988.

Maxime Chevalier-Boisvert, Lucas Willems, and Suman Pal. Minimalistic gridworld environment for openai gym. `https://github.com/maximecb/gym-minigrid`, 2018.

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

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

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

Jean Harb, Tom Schaul, Doina Precup, and Pierre-Luc Bacon. Policy evaluation networks. *arXiv preprint arXiv:2002.11833*, 2020.

Francesco Faccio, Louis Kirsch, and Jürgen Schmidhuber. Parameter-based value functions. In *9th International Conference on Learning Representations*, 2021.

Yuri Burda, Harrison Edwards, Deepak Pathak, Amos J. Storkey, Trevor Darrell, and Alexei A. Efros. Large-scale study of curiosity-driven learning. In *7th International Conference on Learning Representations*, 2019b.

J. Schmidhuber. Exploring the predictable. In A. Ghosh and S. Tsuitsui, editors, *Advances in Evolutionary Computing*, pages 579–612. Springer, 2002.

Lucas Willems. Rl starter files. `https://github.com/lcswillems/rl-starter-files`, 2017.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, 2015.