# Extended Abstract: Hyperparameters in Reinforcement Learning and How To Tune Them

**Theresa Eimer***
Institute for Artificial Intelligence
Leibniz University Hannover
t.eimer@ai.uni-hannover.de

**Marius Lindauer**
Institute for Artificial Intelligence
Leibniz University Hannover
m.lindauer@ai.uni-hannover.de

**Roberta Raileanu**
Meta AI Research
raileanu@meta.com

*This is an Extended Abstract of a Paper accepted for publication at ICML 2023.*

## 1 Motivation

Deep Reinforcement Learning (RL) has been adopting better scientific practices in order to improve reproducibility such as standardized evaluation metrics and reporting as well as greater attention to implementation details and design decisions [13, 10, 14, 1, 16]. However, the process of hyperparameter optimization still varies widely across papers with inefficient grid searches being most commonly used [18, 5, 3, 12]. This makes fair comparisons between RL algorithms challenging. In this paper, we show that hyperparameter choices in RL can significantly affect the agent's final performance and sample efficiency, and that the hyperparameter landscape can strongly depend on the tuning seed which might lead to overfitting to single seeds. We therefore propose adopting established best practices from AutoML, such as the separation of tuning and testing seeds, as well as principled hyperparameter optimization (HPO) across a broad search space [9, 15]. We support this by comparing multiple state-of-the-art HPO tools on a range of RL algorithms and environments to their hand-tuned counterparts, demonstrating that HPO approaches often have higher performance and lower compute overhead (see Figure 1).

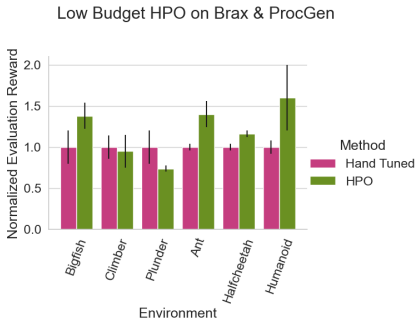


Figure 1: **Comparison of Hyperparameter Tuning Approaches:** state-of-the-art hyperparameter tuning tools, in this case DEHB, match or outperform hand tuning via grid search, while using less than 1/12 of the budget.

As a result of our findings, we recommend a set of best practices for the RL community going forward, which should result in stronger empirical results with fewer computational costs, better reproducibility, and thus faster progress in RL. In order to encourage the adoption of these practices, we provide plug-and-play implementations of the tuning algorithms used in this paper at `https://github.com/facebookresearch/how-to-autorl`.

*Work done during an Internship at Meta

16th European Workshop on Reinforcement Learning (EWRL 2023).


Figure 2: Tuning Results for PPO on Brax (top) and IDAAC on PrcGen (bottom). Shown is the mean evaluation reward across 10 episodes for 3 tuning runs as well as the 98% confidence interval across tuning runs.

## 2 Exploring the RL Hyperparameter Landscape

In order to gain insights into the sensitivity of RL algorithms to hyperparameters, we utilize uniform sweeps across several hyperparameters of DQN, SAC, and PPO on two MiniGrid environments [7], three Brax environments (namely Ant, Halfcheetah, and Humanoid) [11], as well as two classic control environments (namely Pendulum and Acrobot) [6]. For each environment, we find that these algorithms are highly sensitive to the majority of their hyperparameters. Their interactions, at least on the classic control environments, are relatively simple in our experiments. We also find that fairly wide ranges of hyperparameter values are likely to work well for any given algorithm and environment. A major challenging factor for Hyperparameter Optimization in RL, however, is the performance instability of the same configuration between different random seeds which can make the comparison betwen different hyperparameter settings unreliable.

## 3 Tuning Reinforcement Learning Algorithms

To validate our results on the RL hyperparameter landscape, we compare different categories of Hyperparameter Optimization algorithms on classic control environments as well as on challenging tasks from Procgen [8] and Brax. We find that DEHB [2], performed best overall even with small budgets of only 16 or 64 full runs (see Figure 2). Even Random Search [4] proved to be able to outperform large Grid Searches, though showed less reliable scaling properties. The dynamic tuning approaches PB2 [17] and BGT [19] were able to provide well-performing hyperparameter configurations during training, but overall failed to generalize to new test seeds. In all these experiments, we saw significant performance discrepancies between tuning and test seeds. Since neither are usually reported in RL, fair comparisons between algorithms currently depend on whether researchers reproducing results by chance choose the test seeds on all methods. We therefore recommend to adapt best practices for tuning and reporting hyperparameters in RL to ensure better comparisons and therefore faster progress in the field.

## 4 Concluding Remarks

We find that RL algorithms can benefit immensely from using established hyperparameter tuning methods, often producing better results at much lower budgets than grid searches. Thus adopting

state of the art HPO methods would increase both the efficiency and accessibility of the field. Additionally adopting the reporting standards in Algorithm Configuration can furthermore prevent unfair comparisons between RL algorithms, which is currently a widespread phenomenon due to underreporting of tuning methods and seeds.

## References


[1] M. Andrychowicz, A. Raichuk, P. Stanczyk, M. Orsini, S. Girgin, R. Marinier, L. Hussenot, M. Geist, O. Pietquin, M. Michalski, S. Gelly, and O. Bachem. What matters for on-policy deep actor-critic methods? A large-scale study. In *9th International Conference on Learning Representations, ICLR*. OpenReview.net, 2021.

[2] N. Awad, N. Mallik, and F. Hutter. DEHB: evolutionary hyberband for scalable, robust and efficient hyperparameter optimization. In Z. Zhou, editor, *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI*, pages 2147–2153. ijcai.org, 2021.

[3] A. Badia, B. Piot, S. Kapturowski, P. Sprechmann, A. Vitvitskyi, Z. Guo, and C. Blundell. Agent57: Outperforming the atari human benchmark. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 507–517. PMLR, 2020.

[4] J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. 13:281–305, 2012.

[5] C. Berner, G. Brockman, B. Chan, V. Cheung, P. Debiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, R. Józefowicz, S. Gray, C. Olsson, J. Pachocki, M. Petrov, H. de Oliveira Pinto, J. Raiman, T. Salimans, J. Schlatter, J. Schneider, S. Sidor, I. Sutskever, J. Tang, F. Wolski, and S. Zhang. Dota 2 with large scale deep reinforcement learning. *CoRR*, abs/1912.06680, 2019.

[6] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. Openai gym, 2016.

[7] M. Chevalier-Boisvert, L. Willems, and S. Pal. Minimalistic gridworld environment for gymnasium, 2018.

[8] K. Cobbe, C. Hesse, J. Hilton, and J. Schulman. Leveraging procedural generation to benchmark reinforcement learning. In *Proceedings of the 37th International Conference on Machine Learning, ICML*, volume 119 of *Proceedings of Machine Learning Research*, pages 2048–2056. PMLR, 2020.

[9] K. Eggensperger, M. Lindauer, and F. Hutter. Pitfalls and best practices in algorithm configuration. pages 861–893, 2019.

[10] L. Engstrom, A. Ilyas, S. Santurkar, D. Tsipras, F. Janoos, L. Rudolph, and A. Madry. Implementation matters in deep RL: A case study on PPO and TRPO. In *8th International Conference on Learning Representations, ICLR*. OpenReview.net, 2020.

[11] C. Freeman, E. Frey, A. Raichuk, S. Girgin, I. Mordatch, and O. Bachem. Brax - A differentiable physics engine for large scale rigid body simulation. In J. Vanschoren and S. Yeung, editors, *Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks 1, NeurIPS Datasets and Benchmarks 2021*, 2021.

[12] E. Hambro, R. Raileanu, D. Rothermel, V. Mella, T. Rocktäschel, H. Küttler, and N. Murray. Dungeons and data: A large-scale nethack dataset. 2022.

[13] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, and D. Meger. Deep reinforcement learning that matters. In S. McIlraith and K. Weinberger, editors, *Proceedings of the Conference on Artificial Intelligence (AAAI'18)*. AAAI Press, 2018.

[14] C. Hsu, C. Mendler-Dünner, and M. Hardt. Revisiting design choices in proximal policy optimization. *CoRR*, abs/2009.10897, 2020.

[15] M. Lindauer and F. Hutter. Best practices for scientific research on neural architecture search. *Journal of Machine Learning Research*, 21:1–18, 2020.

[16] J. Obando-Ceron and P. Castro. Revisiting rainbow: Promoting more insightful and inclusive deep reinforcement learning research. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML*, volume 139 of *Proceedings of Machine Learning Research*, pages 1373–1383. PMLR, 2021.

[17] J. Parker-Holder, V. Nguyen, and S. Roberts. Provably efficient online hyperparameter optimization with population-based bandits. In H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS*, 2020.

[18] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv:1707.06347 [cs.LG]*, 2017.

[19] X. Wan, C. Lu, J. Parker-Holder, P. Ball, V. Nguyen, B. Ru, and M. Osborne. Bayesian generational population-based training. In I. Guyon, M. Lindauer, M. van der Schaar, F. Hutter, and R. Garnett, editors, *International Conference on Automated Machine Learning, AutoML*, volume 188 of *Proceedings of Machine Learning Research*, pages 14/1–27. PMLR, 2022.