Encore Abstract: A Research Agenda for Usability and Generalisation in Reinforcement Learning

Dennis J.N.J. Soemers[1], Spyridon Samothrakis[2], Kurt Driessens[1], and Mark H.M. Winands[1]

[1] Department of Advanced Computing Sciences, Maastricht University
{dennis.soemers, kurt.driessens, m.winands}@maastrichtuniversity.nl
[2] Institute for Analytics and Data Science, University of Essex
ssamot@essex.ac.uk

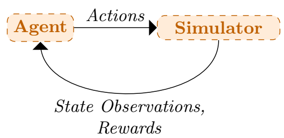
This document is an encore abstract of the position paper entitled "A Research Agenda for Usability and Generalisation in Reinforcement Learning" [26].

Introduction. Reinforcement learning (RL) [27] researchers have largely converged on common APIs for the development of benchmarks used to evaluate the performance of RL algorithms. New environments are customarily written in general-purpose programming languages such as `C++` or `Python`, and implement a Gym-based [5] API for learning algorithms to interface with environments.

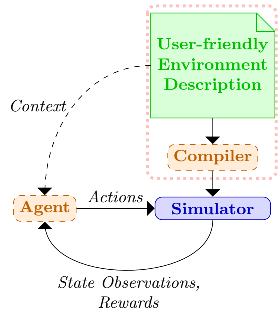
Discussions on experimental methodologies and statistical analyses of empirical evaluations in RL have been on the rise within the RL research community [13,1,15,22,14,29], but we see little to no discussion on *how* or *by whom* tasks or environments are described or implemented. In our position paper, we argue that the assumption that environments can be implemented in general-purpose programming languages, by engineers familiar with machine learning, (i) poses a challenge to widespread adoption of RL for real-world use cases, and (ii) also leads the research community to miss out on interesting research directions with respect to generalisation and transfer in RL.

We posit that more widespread application of RL will be greatly aided if users can express their tasks in user-friendly domain-specific languages (DSLs) [20,2], or even in natural language (see Fig. 1). Once we adopt a methodology in which environments are represented in explicit forms that can be provided as inputs to an agent (e.g., DSL or natural language snippets), we can also explore new forms of generalisation or transfer in RL, where effective generalisation or zero-shot transfer to unseen environments may become feasible given sufficient understanding of the task descriptions.

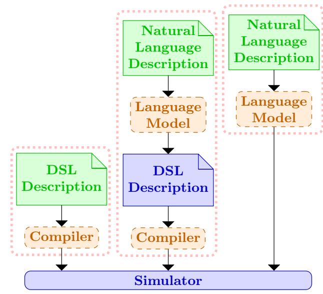
Environment Description Languages. The standard practice is to implement environments in general-purpose programming languages, or more advanced toolkits such as CUDA or JAX [6,9,18,3,17,10,12,16,25,4,24,28,8]. This practice requires engineering and RL expertise, therefore forming a barrier to entry. Provided that they are sufficiently user-friendly for domain experts, DSLs can lower this barrier. An example of such a DSL is the DSL of Ludii [23], which



(a) A simulator of the environment is implemented in a general-purpose programming language. This is the most common setting in current research.



(b) The environment is described in a user-friendly language, and compiled into a simulator. The description may serve as additional input to the agent.



(c) A DSL description can be converted into a simulator by a compiler. A natural language description can first be translated into an intermediate DSL description by a language model, or can be translated directly with a language model serving as compiler.

Fig. 1: Orange boxes with dashed lines represent components that require substantial engineering or RL expertise. The green components can be provided by users with little to no engineering expertise. **(a)** A depiction of the customary setting in RL research. **(b)** The approach for which we posit that increased research attention is warranted. **(c)** User-friendly environment descriptions may be written in a DSL, or in a natural language, where the latter approach may or may not generate an intermediate DSL description.

boasts an official collection of over 1400 different board games described in the same DSL. Such a large collection would have been impractical to implement in a general-purpose programming language. Natural languages would arguably be more user-friendly if they could be interpreted correctly, but until large language models become more consistently reliable, it likely remains advisable to at least use human-verifiable DSLs as intermediary [7,21,30].

Environment Descriptions as Context for Generalisation. If RL agents are not provided with sufficient context to disambiguate between different environments (or tasks or goals), generalisation to unseen environments may not be possible without strong assumptions on their similarity to previously seen ones [19,11]. We argue that precise, complete, and unambiguous descriptions of environments—complete in the sense that they carry sufficient information that they could be compiled into a correct simulator—are ideal candidates to serve as such context, and enable progress in zero-shot generalisation in RL.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Agarwal, R., Schwarzer, M., Castro, P.S., Courville, A., Bellemare, M.G.: Deep reinforcement learning at the edge of the statistical precipice. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems. vol. 34, pp. 29304–29320. Curran Associates, Inc. (2021)
2. Aram, M., Neumann, G.: Multilayered analysis of co-development of business information systems. Journal of Internet Services and Applications **6**(13) (2015)
3. Bettini, M., Kortvelesy, R., Blumenkamp, J., Prorok, A.: VMAS: A vectorized multi-agent simulator for collective robot learning. In: Proceedings of the 16th International Symposium on Distributed Autonomous Robotic Systems. DARS '22, Springer (2022)
4. Bonnet, C., Luo, D., Byrne, D., Surana, S., Abramowitz, S., Duckworth, P., Coyette, V., Midgley, L.I., Tegegn, E., Kalloniatis, T., Mahjoub, O., Macfarlane, M., Smit, A.P., Grinsztajn, N., Bolge, R., Waters, C.N., Mimouni, M.A., Sob, U.A.M., de Kock, R., Singh, S., Furelos-Blanco, D., Le, V., Pretorius, A., Laterre, A.: Jumanji: a diverse suite of scalable reinforcement learning environments in JAX. In: Proceedings of the International Conference on Learning Representations (2024)
5. Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., Zaremba, W.: OpenAI gym. https://arxiv.org/abs/1606.01540 (2016)
6. Dalton, S., Frosio, I.: Accelerating reinforcement learning through GPU Atari emulation. In: Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., Lin, H. (eds.) Advances in Neural Information Processing Systems. vol. 33, pp. 19773–19782. Curran Associates, Inc. (2020)
7. Desai, A., Gulwani, S., Hingorani, V., Jain, N., Karkare, A., Marron, M., R, S., Roy, S.: Program synthesis using natural language. In: Proceedings of the 38th International Conference on Software Engineering. p. 345–356. Association for Computing Machinery (2016)
8. Earle, S., Todd, G., Li, Y., Khalifa, A., Jiang, Z., Nasir, M.U., Banburski-Fahey, A., Togelius, J.: PuzzleJAX: A benchmark for reasoning and learning. https://arxiv.org/abs/2508.16821 (2025)
9. Freeman, C.D., Frey, E., Raichuk, A., Girgin, S., Mordatch, I., Bachem, O.: Brax - a differentiable physics engine for large scale rigid body simulation (2021), http://github.com/google/brax
10. Frey, S., Li, K., Nagy, P., Sapora, S., Lu, C., Zohren, S., Foerster, J., Calinescu, A.: JAX-LOB: A GPU-accelerated limit order book simulator to unlock large scale reinforcement learning for trading. In: ICAIF '23: Proceedings of the Fourth ACM International Conference on AI in Finance. pp. 583–591 (2023)
11. Ghosh, D., Rahme, J., Kumar, A., Zhang, A., Adams, R.P., Levine, S.: Why generalization in rl is difficult: Epistemic pomdps and implicit partial observability. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems. vol. 34, pp. 25502–25515. Curran Associates, Inc. (2021)
12. Gulino, C., Fu, J., Luo, W., Tucker, G., Bronstein, E., Lu, Y., Harb, J., Pan, X., Wang, Y., Chen, X., Co-Reyes, J.D., Agarwal, R., Roelofs, R., Lu, Y., Montali, N., Mougin, P., Yang, Z., B, W., Faust, A., McAllister, R., Anguelov, D., Sapp, B.: Waymax: An accelerated, data-driven simulator for large-scale autonomous driving research. In: Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks (2023)

13. Henderson, P., Islam, R., Bachman, P., Pineau, J., Precup, D., Meger, D.: Deep reinforcement learning that matters. In: Proceedings of the 32nd AAAI Conference on Artificial Intelligence. pp. 3207–3214. AAAI (2018)
14. Jordan, S.M., White, A., da Silva, B.C., White, M., Thomas, P.S.: Position: Benchmarking is limited in reinforcement learning research. In: Salakhutdinov, R., Kolter, Z., Heller, K., Weller, A., Oliver, N., Scarlett, J., Berkenkamp, F. (eds.) Proceedings of the 41st International Conference on Machine Learning. Proceedings of Machine Learning Research, vol. 235, pp. 22551–22569. PMLR (2024)
15. Jordan, S.M.: Scientific experiments in reinforcement learning. https://nips.cc/virtual/2022/63891 (2022), opinion talk contributed to the Deep Reinforcement Learning Workshop at NeurIPS 2022
16. Koyamada, S., Okano, S., Nishimori, S., Murata, Y., Habara, K., Kita, H., Ishii, S.: Pgx: Hardware-accelerated parallel game simulators for reinforcement learning. In: Advances in Neural Information Processing Systems (2023)
17. Lange, R.T.: gymnax: A JAX-based reinforcement learning environment library (2022), http://github.com/RobertTLange/gymnax
18. Makoviychuk, V., Wawrzyniak, L., Guo, Y., Lu, M., Storey, K., Macklin, M., Hoeller, D., Rudin, N., Allshire, A., Handa, A., State, G.: Isaac gym: High performance GPU based physics simulation for robot learning. In: Vanschoren, J., Yeung, S. (eds.) Proceedings of the Neural Information Processing Systems Track on Datasets and Benchmarks. vol. 1. Curran (2021)
19. Malik, D., Li, Y., Ravikumar, P.: When is generalizable reinforcement learning tractable? In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P., Vaughan, J.W. (eds.) Advances in Neural Information Processing Systems. vol. 34, pp. 8032–8045. Curran Associates, Inc. (2021)
20. Mernik, M., Heering, J., Sloane, A.M.: When and how to develop domain-specific languages. ACM Computing Surveys **37**(4), 316–344 (2005)
21. Oswald, J., Srinivas, K., Kokel, H., Lee, J., Katz, M., Sohrabi, S.: Large language models as planning domain generators. In: Proceedings of the International Conference on Automated Planning and Scheduling. vol. 34, pp. 423–431 (2024)
22. Patterson, A., Neumann, S., White, M., White, A.: Empirical design in reinforcement learning. https://arxiv.org/abs/2304.01315 (2023)
23. Piette, É., Soemers, D.J.N.J., Stephenson, M., Sironi, C.F., Winands, M.H.M., Browne, C.: Ludii – the ludemic general game system. In: Giacomo, G.D., Catala, A., Dilkina, B., Milano, M., Barro, S., Bugarín, A., Lang, J. (eds.) Proceedings of the 24th European Conference on Artificial Intelligence (ECAI 2020). Frontiers in Artificial Intelligence and Applications, vol. 325, pp. 411–418. IOS Press (2020)
24. Ponse, K., Kleuker, J.F., Moerland, T.M., Plaat, A.: Chargax: A JAX accelerated EV charging simulator. In: Reinforcement Learning Conference (2025), to appear
25. Rutherford, A., Ellis, B., Gallici, M., Cook, J., Lupu, A., Ingvarsson, G., Willi, T., Khan, A., de Witt, C.S., Souly, A., Bandyopadhyay, S., Samvelyan, M., Jiang, M., Lange, R.T., Whiteson, S., Lacerda, B., Hawes, N., Rocktäschel, T., Lu, C., Foerster, J.N.: JaxMARL: Multi-agent RL environments and algorithms in JAX. https://arxiv.org/abs/2311.10090 (2023)
26. Soemers, D.J.N.J., Samothrakis, S., Driessens, K., Winands, M.H.M.: A research agenda for usability and generalisation in reinforcement learning. Revised Selected Papers for ICAART 2025 (2025), to appear
27. Sutton, R.S., Barto, A.G.: Reinforcement Learning: An Introduction. MIT Press, Cambridge, MA, 2 edn. (2018)

28. Todd, G., Padula, A.G., Soemers, D.J.N.J., Togelius, J.: Ludax: A GPU-accelerated domain specific language for board games. https://arxiv.org/abs/2506.22609 (2025)
29. Voelcker, C., Hussing, M., Eaton, E.: Can we hop in general? A discussion of benchmark selection and design using the Hopper environment. In: Finding the Frame workshop @ Reinforcement Learning Conference (2024)
30. Zuo, M., Velez, F.P., Li, X., Littman, M.L., Bach, S.H.: Planetarium: A rigorous benchmark for translating text to structured planning languages. https://arxiv.org/abs/2407.03321 (2024)