

SHARPIE: A Modular Framework for Reinforcement Learning and Human-AI Interaction Experiments

Hüseyin Aydın^{1*}, Kevin Godin-Dubois^{2*}, Libio Goncalves Braz^{1*}, Floris den Hengst^{2*}, Kim Baraka², Mustafa Mert Çelikok³, Andreas Sauter², Shihan Wang¹, Frans A. Oliehoek³

¹Utrecht University, ²Vrije Universiteit Amsterdam, ³TU Delft

h.aydin@uu.nl, k.j.m.godin-dubois@vu.nl, l.b.goncalvesbraz@uu.nl, f.den.hengst@vu.nl, k.baraka@vu.nl, m.m.celikok@tudelft.nl, a.sauter@vu.nl, s.wang2@uu.nl, f.a.oliehoek@tudelft.nl

Abstract

Reinforcement learning (RL) offers a general approach for modeling and training AI agents, including human-AI interaction scenarios. In this paper, we propose SHARPIE (Shared Human-AI Reinforcement Learning Platform for Interactive Experiments) to address the need for a generic framework to support experiments with RL agents and humans. Its modular design consists of a versatile wrapper for RL environments and algorithm libraries, a participant-facing web interface, logging utilities, deployment on popular cloud and participant recruitment platforms. It empowers researchers to study a wide variety of research questions related to the interaction between humans and RL agents, including those related to interactive reward specification and learning, learning from human feedback, action delegation, preference elicitation, user-modeling, and human-AI teaming. The platform is based on a generic interface for human-RL interactions that aims to standardize the field of study on RL in human contexts.

Code — <https://github.com/libgoncalv/SHARPIE>

Introduction

Reinforcement learning (RL) refers to a family of algorithms in which agents learn from the consequences of their own interactions with an environment to try to maximize the long-term reward obtained (Sutton and Barto 1998). The available compute has seen an increase in the use of RL methods in a wide variety of problems, many of which involve humans in some sense (Den Hengst et al. 2020). RL agents interact with humans in a wide variety of ways: in some problem settings, RL agents observe humans to achieve a common goal or act on their behalf (Natarajan et al. 2010); in other settings, RL agents communicate with humans to provide services or support their decision making (Zhao et al. 2021); human feedback can also be used to evaluate the value of the agents’ actions in other settings (Knox and Stone 2009; Christiano et al. 2017).

These different types of interaction highlight the need to incorporate humans in the training and evaluation of RL

agents. However, their interaction patterns remain limited to basic and unidirectional. In contrast, hybrid human-AI intelligence, as envisioned by Akata et al. (2020) emphasizes the need for rich, diverse, and dynamic interaction patterns between humans and AI agents in order to effectively address problems which humans nor agents cannot solve independently. Although there are software packages to study the interaction of RL agents and their environments, little attention has been paid to training and evaluation of RL agents in diverse and dynamic interaction patterns that include both humans and other agents.

In particular, popular RL libraries such as Gymnasium (Brockman 2016; Towers et al. 2024), PettingZoo (Terry et al. 2021), JaxMARL (Rutherford et al. 2024), MO-Gymnasium (Alegre et al. 2022) or HIPPO-Gym (Taylor et al. 2023), provide a framework to train RL agents in benchmarks of particular formulations of the RL problem. These libraries also provide an interface to include a custom domain in the training. However, none of these libraries by themselves specifically tackles rich, diverse and dynamic interactions among humans and RL agents such as multi-modal communications for shared observations, delegated actions and common goals that we believe to be required for hybrid intelligence.

To address this research gap, we propose SHARPIE (Shared Human-AI Reinforcement-Learning Platform for Interactive Experiments) as a platform for studying how multiple humans and RL agents interact and collaborate effectively. This modular platform streamlines the design and execution of experiments with humans and interactive RL agents by:

- providing a versatile wrapper for popular RL, multi-agent RL, and multi-objective RL environments and algorithms
- supporting configurable communication channels between the human and the RL agents with various modalities
- offering logging services, deployment utilities, and participant recruitment platforms integration.

The platform aims to empower researchers to address research questions on the interaction of RL agents and humans, including those related to e.g. interpretability, in-

*These authors contributed equally.

teractive reward specification and learning, learning from human feedback, action delegation, preference elicitation, user-modeling, and human-AI teaming. As SHARPIE aims to facilitate human-agent collaboration research in the most generic way possible, it may also prove useful for simulating interactive robotic tasks prior to their physical implementation. Hence, we believe that SHARPIE will also lead to new research questions in the field of human-robot interaction (Goodrich, Schultz et al. 2008).

Apart from its potential impact on other fields, the main contribution of SHARPIE is to introduce a standard for RL-based human-agent interactions in a multi-agent setting in the same way that the Gymnasium API has become this standard for fully simulated RL environments. In this paper, we first present some motivating use cases to sketch the scope and intended outcomes of this project with tangible use cases, including their relation with cognitive science. Next, we describe the SHARPIE framework design principles, the challenges encountered, and the solutions implemented to tackle these. We close with a discussion of future ambitions.

Motivating Use Cases

Table 1 provides practical examples of target use cases that illustrate the wide range of problems that SHARPIE is designed to deal with. It details for each use case the roles taken on by human(s) and agent(s), an example environment and an envisioned interaction interface. These example use cases are not comprehensive and we will only implement some of these initially. We provide them here for illustrative purposes.

The first use case, entitled ‘reward annotation’, corresponds to a setting in which a human annotator supplies the agent with a dense reward signal based on observations of the agent interacting with an environment. This reward signal complements a sparse reward signal in a maze navigation task, and models a setting in which a human expert guides the agent in completing its task successfully. It requires a visual representation of the environment and human reward feedback components in the User Interface (UI). The second, ‘exploration’, use case targets the same maze solving task but suggests richer platform capabilities, as it requires the participant to be able to define desired maze layouts and select promising policies interactively (Godin-Dubois, Miras, and Kononova 2025). This use-case highlights the use of a person to specify the task the agent should solve, which can be found in myriad examples of humans interacting with RL agents (Chang et al. 2024).

The third ‘teaching’ use case targets a multi-agent setting in which a number of agents need to learn from human demonstrators the task of covering a number of landmarks in the environment. This task invites researchers to study how goals (i.e. target locations) of the other agents can be estimated in a straightforward unidirectional way (Hou et al. 2024). The ‘action delegation’ use case deals with the problem of handing over control between human and learner, and suggests the need for humans and learners to effectively model the other agents’ intended actions, goals and capabilities in a bidirectional setting.

The ‘task specification’ use case targets interactively guiding the agent towards a high-level goal by specifying (sub)tasks for sparse reward tasks or open-ended settings. This use case illustrates the need for communication channels between agents and humans to specify goals in e.g. free text or a formal representation of the task at hand (Icarte et al. 2022; Den Hengst et al. 2022). It allows studying the ability of humans to effectively express their (sub)tasks, as well as studying approaches to agree on the semantics of task specifications. The ‘human-AI teaming’ use case suggests the ability for human and RL agents to bidirectionally communicate their observations, intentions or goals (Zhu, Dastani, and Wang 2024). Through effective and efficient interactions among humans and agents, we can form AI-human teams to coordinate learning behaviors, for example to solve classical predator-prey tasks (Lowe et al. 2017).

Finally, we consider two multiobjective decision-making problems. The first, ‘utility elicitation’ use case addresses the need for humans to select from a set of optimal policies in a multi-objective setting. The agent learns the attainable trade-offs between the gathering of treasures and the amount of time spent under water in a sea environment. It then presents their trade-offs as a Pareto front of policies. The human stakeholder can then select a solution policy that optimizes the particular trade-off that maximizes *utility*.

The multiobjective setting is generalized to multiple stakeholders in the final use case, in which the agent controls the amount of water released by an energy-generating dam. The different stakeholders, i.e., the inhabitants of an upstream village, a downstream village, and the dam operators, need to agree on a policy that balances the objectives of meeting energy and water demand while minimizes downstream and upstream costs of flooding. The individual utilities for these stakeholders are mined separately and then used to compute a set of policies that for some trade-off to inform the stakeholders as part of a shared decision-making process using example trajectories.

We close this section with a brief discussion of how the proposed framework supports the study of RL, cognitive sciences, and their interconnection. Firstly, the ability to collect human demonstrations, reward annotations, and task descriptions (use cases 1,2,3 and 5) allows the study of human decision making, task success, and task definitions in humans. Furthermore, collaboration between RL agents and humans (use cases 4 and 6) allows the study of social learning, trust and reciprocity dynamics, and theory of mind in hybrid human-AI settings (Jara-Ettinger 2019; Alexandre S. Pires 2024). The impact of RL agents on group decision-making processes can be studied through use cases 6, 7 and 8 (Seo and Lee 2017), while the presence of communication channels (5) allows the study of language, communication, and grounding (Zhang et al. 2024). We believe that the number of possible (inter)connections between RL and cognitive science that can be studied with SHARPIE will continue to grow as suitable environments, algorithms and UI components are contributed to the SHARPIE framework.

		Human role(s)	Agent role(s)	Environment	Interface
1	Reward annotation	1 annotator	1 learner	AMaze (discrete) ¹	Video output with timed feedback
2	Exploration	1 task specifier	n learners	AMaze (hybrid) ²	Maze layout & policy selection
3	Teaching	n teachers	n students	Simple Spread ³	Demonstrations
4	Action delegation	1 controller	1 learner	Mountain Car ⁴	Environment interactions
5	Task specification	1 task specifier	1 learner	MineCraft ⁵	Text communication channel
6	Human-AI Teaming	n collaborators		Simple Tag ³	Environment interactions, intentions
7	Utility elicitation	1 stakeholder	1 learner	Deep Sea Treasure ⁶	Example policies
8	Shared decision support	n stakeholders	1 advisor	Water-reservoir ⁷	Environment interactions

Table 1: Use case examples. References are: 1- Godin-Dubois, Miras, and Kononova (2024); 2- Godin-Dubois, Miras, and Kononova (2025); 3- Lowe et al. (2017); 4- Moore (1990); 5- Johnson et al. (2016); 6- Vamplew et al. (2011); 7- Pianosi, Castelletti, and Restelli (2013).

Related Work

Several successful software packages and platforms for the development of RL algorithms and evaluation environments have been made over the past decades, emphasizing a different number of objectives, different numbers of agents, different kinds of tasks and different programming languages, etc. We do not intend to review all software packages in RL and here we focus only on the related work that particularly deals with RL platforms that explicitly target the study of RL in a context that involves humans.

HIPPO-GYM (Taylor et al. 2023), provides a framework that involves human interactions in a RL setting. However, this framework is limited to single-agent tasks and provides only specific interactions in those tasks (i.e. humans teaching RL agents). Knierim et al. (2024a) presents a framework in human teaching RL agents setting with audio communication. Their codebase also facilitates a Wizard-of-Oz approach (Green and Wei-Haas 1985), where a participant can play the role of a teacher while a researcher plays the role of an artificial agent. Although an alternative modality is proposed in this study, the interactions are designed to be one-way, with human teachers guiding RL-agents.

Another line of work presents Reinforcement Learning from Human Feedback (RLHF) platforms (Christiano et al. 2017; Yuan et al. 2024). However, the interfaces for human interaction provided by these platforms are only capable of handling elicited feedback for RLHF.

There are several software platforms for conducting behavioral experiments (Peirce 2007; De Leeuw 2015). None of these offers any infrastructure for any type of artificially intelligent entities, whether learning or not.

Various platforms and software packages have been developed for the deployment of RL (Gauci et al. 2019; Albers, Neerinx, and Brinkman 2022; Zhu et al. 2024). However, these are not focused on controlled experiments on the interaction between RL agents and humans.

Previous studies in multi-agent learning such as MAgent (Zheng et al. 2018) and MOMAland (Felten et al. 2024) are capable of handling multi-agent experiments. MAgent supports a large population of agents, ranging from hundreds to millions, whereas it is possible to train the agents for multi-

ple objectives in MOMAland. However, these packages do not yet support human interaction in any way. SHARPIE specifically targets the gap of easily setting up experiments with (multiple) human participants and RL agents by interfacing with existing environment and algorithm libraries.

One of the most related work to our study is Interactive Gym (McDonald 2024) which is a framework in early stage of development that enables human-AI interaction in multi-agent RL settings. This framework currently provides two different setups: single human with multiple agents in a browser based execution and any number of human and AI agents in a client-server based execution of experiments. It is difficult to assess the exact differences between this framework and ours, as both are under development. However, with its current structure, Interactive Gym seems to be specific to the Gymnasium environments where human(s) can take the control of RL agents in a game setup which can require more effort to deal with a diverse set of scenarios including various type of human-RL agent interactions. SHARPIE, on the other hand, aims to provide a more generic and extendable design that is able to efficiently facilitate such scenarios.

SHARPIE Framework

The SHARPIE library is a Python-based web framework that is currently under active development and that aims to provide a versatile wrapper around popular Reinforcement Learning a) environments, b) algorithms, and c) methodologies (see Figure 1). For the first part, it can encapsulate any existing environment that follows the conventional Gymnasium API (reset, step, render, etc.) which encompasses most of the existing RL platforms (see Appendix A). Most importantly, the ambition of SHARPIE is not to be tightly integrated with any one particular environment or library, but rather to be compatible with many.

Furthermore, one of its core components is a customizable and easily deployable front-end UI which is primarily web-based and multi-modal. This allows human users to interact with the environment and other (RL or human) agents. Additionally, RL agents may be able to request action delegation, explicitly or as prompted by their learning algorithm,

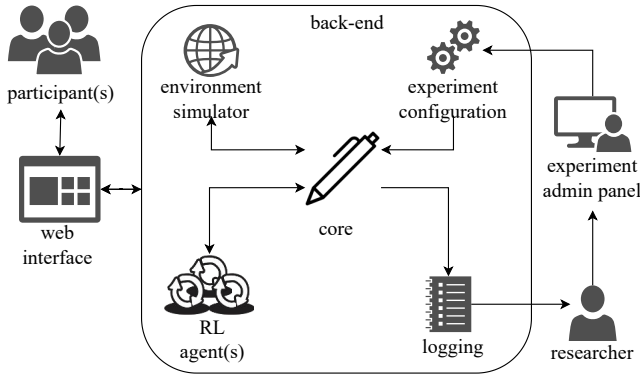


Figure 1: High-level SHARPIE architecture.

to further increase the range of interaction scenarios that SHARPIE can help streamline.

The front-end also provides various complementary utilities to further smoothen out the experimental processes: from (a)synchronous evaluations of a learning agent to scheduling and management on long-term data storage. In the first case, this takes many forms, such as a preference elicitation module to visualize and rank trajectories, actions, and policies. In the second case, this includes all the logging facilities required in a large-scale RL experiment, especially one involving human participants. Indeed, it is of paramount importance to be able to securely and robustly store any relevant data, as restarting such an experiment from scratch might be, at best, long and costly and, at worse, practically impossible.

In addition, SHARPIE is designed to support multi-modal communication channels to allow communication between agents (again, RL or human). Depending on said agents, these channels may be used for e.g. coordination, teaching, or overriding. While initially designed for short semantic content, such as predefined sets of symbols, these communication mediums will be expanded upon to include any and all relevant formats (natural text, images, audio, etc.).

Finally, the library is designed with utilities to deploy to a cloud server, a private machine, or a local host. Given the widespread support for Python, this encompasses any desktop operating system (e.g. Linux, Mac, Windows) with Python 3 installed, including remote web-servers or platforms such as Amazon Web Services (AWS). This provides an abstraction between the machines that the researchers and participants are using and where the experiment is actually running. Such a feature is essential in studies involving numerous participants with varying locations and hardware.

Discussion & Future Work

We have presented and motivated the design and implementation of a framework to accelerate human-AI interaction studies, with a particular focus on studies for multi-agent tasks involving humans. As stated, the scope of this project is not to implement alternative environment and algorithms for such studies, but rather to integrate with existing solutions seamlessly and provide utilities to ease exper-

imentation with participants. This is done by empowering researchers through a modular software interface relying, in part, on the de facto standard set by Gymnasium. With SHARPIE we aim to provide an easily integrable framework that researchers can use to painlessly set up experiments involving both human and artificial agents. Our hope is that in turn such an architecture lays the foundation for a standard for the interaction between human and artificial agents.

This modular approach also allows for numerous directions of improvement in terms of interoperability and features. In the former case, we aim to provide an increasing number of ready-made, supported plugins to handle a large part of the existing work on environments, libraries, and deployment options. In the latter case, we plan to widen the scope of possible human-agent interactions by incorporating additional modalities such as audio or video (Christofi and Baraka 2024; Knierim et al. 2024b). The resulting real-time and multimodal communication between users and agents would allow the study of rich and fine-grained communication protocols, and support experiments involving pitch and tone, nonverbal cues, etc. Finally, we envision a hosted version of SHARPIE that can be used for outreach, education and user literacy purposes.

Acknowledgements

This research was funded by the Hybrid Intelligence Center, a 10-year programme funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research, <https://hybridintelligence-centre.nl>, grant number 024.004.022.

References

- Akata, Z.; Balliet, D.; De Rijke, M.; Dignum, F.; Dignum, V.; Eiben, G.; Fokkens, A.; Grossi, D.; Hindriks, K.; Hoos, H.; et al. 2020. A research agenda for hybrid intelligence: augmenting human intellect with collaborative, adaptive, responsible, and explainable artificial intelligence. *Computer*, 53(8): 18–28.
- Albers, N.; Neerincx, M. A.; and Brinkman, W.-P. 2022. Addressing people’s current and future states in a reinforcement learning algorithm for persuading to quit smoking and to be physically active. *Plos one*, 17(12): e0277295.
- Alegre, L. N.; Felten, F.; Talbi, E.-G.; Danoy, G.; Nowé, A.; Bazzan, A. L.; and da Silva, B. C. 2022. MO-Gym: A library of multi-objective reinforcement learning environments. In *Proceedings of the 34th Benelux Conference on Artificial Intelligence BNAIC/Benelearn*, volume 2022, 2.
- Alexandre S. Pires, F. P. S. 2024. Artificial Agents Facilitate Human Cooperation Through Indirect Reciprocity. In *Frontiers in Artificial Intelligence and Applications*, Volume 392: *ECAI 2024*. IOS Press.
- Brockman, G. 2016. OpenAI Gym. *arXiv preprint arXiv:1606.01540*.
- Chang, W.-T.; Wang, S.; Kramer, S.; Oey, M.; and Ben Al-louch, S. 2024. Human-Centered AI for Dementia Care: Using Reinforcement Learning for Personalized Interventions Support in Eating and Drinking Scenarios. In *HHAI 2024*:

Hybrid Human AI Systems for the Social Good, 84–93. IOS Press.

Christiano, P. F.; Leike, J.; Brown, T.; Martic, M.; Legg, S.; and Amodei, D. 2017. Deep reinforcement learning from human preferences. *Advances in neural information processing systems*, 30.

Christofi, K.; and Baraka, K. 2024. Uncovering Patterns in Humans that Teach Robots through Demonstrations and Feedback. In *Companion of the 2024 ACM/IEEE International Conference on Human-Robot Interaction*, 332–336.

De Leeuw, J. R. 2015. jsPsych: A JavaScript library for creating behavioral experiments in a Web browser. *Behavior research methods*, 47: 1–12.

DeepMind; Babuschkin, I.; Baumli, K.; Bell, A.; Bhupatiraju, S.; Bruce, J.; Buchlovsky, P.; Budden, D.; Cai, T.; Clark, A.; Danihelka, I.; Dedieu, A.; Fantacci, C.; Godwin, J.; Jones, C.; Hemsley, R.; Hennigan, T.; Hessel, M.; Hou, S.; Kapturowski, S.; Keck, T.; Kemaev, I.; King, M.; Kunesch, M.; Martens, L.; Merzic, H.; Mikulik, V.; Norman, T.; Papamakarios, G.; Quan, J.; Ring, R.; Ruiz, F.; Sanchez, A.; Sartran, L.; Schneider, R.; Sezener, E.; Spencer, S.; Srinivasan, S.; Stanojević, M.; Stokowiec, W.; Wang, L.; Zhou, G.; and Viola, F. 2020. The DeepMind JAX Ecosystem.

Den Hengst, F.; François-Lavet, V.; Hoogendoorn, M.; and van Harmelen, F. 2022. Reinforcement learning with option machines. In *31st International Joint Conference on Artificial Intelligence, IJCAI 2022*, 2909–2915. International Joint Conferences on Artificial Intelligence Organization.

Den Hengst, F.; Grua, E. M.; el Hassouni, A.; and Hoogendoorn, M. 2020. Reinforcement learning for personalization: A systematic literature review. *Data Science*, 3(2): 107–147.

Felten, F.; Alegre, L. N.; Nowé, A.; Bazzan, A. L. C.; Talbi, E. G.; Danoy, G.; and Silva, B. C. d. 2023. A Toolkit for Reliable Benchmarking and Research in Multi-Objective Reinforcement Learning. In *Proceedings of the 37th Conference on Neural Information Processing Systems (NeurIPS 2023)*.

Felten, F.; Ucak, U.; Azmani, H.; Peng, G.; Röpke, W.; Baier, H.; Mannion, P.; Roijers, D. M.; Terry, J. K.; Talbi, E.-G.; et al. 2024. MOMAland: A Set of Benchmarks for Multi-Objective Multi-Agent Reinforcement Learning. In *Multi-objective Decision Making Workshop at ECAI 2024*.

Gauci, J.; Conti, E.; Liang, Y.; Virochsiri, K.; He, Y.; Kaden, Z.; Narayanan, V.; Ye, X.; Chen, Z.; and Fujimoto, S. 2019. Horizon: Facebook’s open source applied reinforcement learning platform. In *ICML 2019 Workshop on RL4RealLife*.

Godin-Dubois, K.; Miras, K.; and Kononova, A. 2025. AMaze: An Intuitive Benchmark Generator for Fast Prototyping of Generalizable Agents. *Frontiers in Artificial Intelligence*, in press.

Godin-Dubois, K.; Miras, K.; and Kononova, A. V. 2024. AMaze: A Benchmark Generator for Sighted Maze-Navigating Agents. *Journal of Open Source Software*, in press.

Goodrich, M. A.; Schultz, A. C.; et al. 2008. Human–robot interaction: a survey. *Foundations and Trends® in Human–Computer Interaction*, 1(3): 203–275.

Green, P.; and Wei-Haas, L. 1985. The Rapid Development of User Interfaces: Experience with the Wizard of OZ Method. *Proceedings of the Human Factors Society Annual Meeting*, 29(5): 470–474.

Hou, M.; Hindriks, K.; Eiben, G.; and Baraka, K. 2024. “Give Me an Example Like This”: Episodic Active Reinforcement Learning from Demonstrations. In *Proceedings of the 12th International Conference on Human-Agent Interaction, HAI ’24*, 287–295. New York, NY, USA: Association for Computing Machinery. ISBN 9798400711787.

Huang, S.; Dossa, R. F. J.; Ye, C.; Braga, J.; Chakraborty, D.; Mehta, K.; and Araújo, J. G. 2022. CleanRL: High-quality Single-file Implementations of Deep Reinforcement Learning Algorithms. *Journal of Machine Learning Research*, 23(274): 1–18.

Icarte, R. T.; Klassen, T. Q.; Valenzano, R.; and McIlraith, S. A. 2022. Reward machines: Exploiting reward function structure in reinforcement learning. *Journal of Artificial Intelligence Research*, 73: 173–208.

Jara-Ettinger, J. 2019. Theory of mind as inverse reinforcement learning. *Current Opinion in Behavioral Sciences*, 29: 105–110.

Johnson, M.; Hofmann, K.; Hutton, T.; and Bignell, D. 2016. The Malmo Platform for Artificial Intelligence Experimentation. In *Ijcai*, volume 16, 4246–4247.

Knierim, M.; Jain, S.; Aydoğar, M. H.; Mitra, K.; Desai, K.; Saran, A.; and Baraka, K. 2024a. Prosody as a Teaching Signal for Agent Learning: Exploratory Studies and Algorithmic Implications. *arXiv preprint arXiv:2410.23554*.

Knierim, M.; Jain, S.; Aydoğar, M. H.; Mitra, K. D.; Desai, K.; Saran, A.; and Baraka, K. 2024b. Leveraging Prosody as an Informative Teaching Signal for Agent Learning: Exploratory Studies and Algorithmic Implications. In *Proceedings of the 26th International Conference on Multimodal Interaction*, 95–123.

Knox, W. B.; and Stone, P. 2009. Interactively shaping agents via human reinforcement: The TAMER framework. In *Proceedings of the fifth international conference on Knowledge capture*, 9–16.

Lancot, M.; Lockhart, E.; Lespiau, J.-B.; Zambaldi, V.; Upadhyay, S.; Pérolat, J.; Srinivasan, S.; Timbers, F.; Tuyls, K.; Omidshafiei, S.; Hennes, D.; Morrill, D.; Muller, P.; Ewalds, T.; Faulkner, R.; Kramár, J.; Vylder, B. D.; Saeta, B.; Bradbury, J.; Ding, D.; Borgeaud, S.; Lai, M.; Schrittwieser, J.; Anthony, T.; Hughes, E.; Danihelka, I.; and Ryan-Davis, J. 2019. OpenSpiel: A Framework for Reinforcement Learning in Games. *CoRR*, abs/1908.09453.

Liang, E.; Liaw, R.; Nishihara, R.; Moritz, P.; Fox, R.; Goldberg, K.; Gonzalez, J.; Jordan, M.; and Stoica, I. 2018. RLlib: Abstractions for Distributed Reinforcement Learning. In Dy, J.; and Krause, A., eds., *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, 3053–3062. PMLR.

Lowe, R.; Wu, Y.; Tamar, A.; Harb, J.; Abbeel, P.; and Mordatch, I. 2017. Multi-Agent Actor-Critic for Mixed

- Cooperative-Competitive Environments. *Neural Information Processing Systems (NIPS)*.
- McDonald, C. 2024. Interactive Gym. <https://github.com/hasemcd/interactive-gym>.
- Moore, A. W. 1990. Efficient memory-based learning for robot control. Technical report, University of Cambridge, Computer Laboratory.
- Natarajan, S.; Kunapuli, G.; Judah, K.; Tadepalli, P.; Kersting, K.; and Shavlik, J. 2010. Multi-agent inverse reinforcement learning. In *2010 ninth international conference on machine learning and applications*, 395–400. IEEE.
- Peirce, J. W. 2007. PsychoPy—psychophysics software in Python. *Journal of neuroscience methods*, 162(1-2): 8–13.
- Pianosi, F.; Castelletti, A.; and Restelli, M. 2013. Tree-based fitted Q-iteration for multi-objective Markov decision processes in water resource management. *Journal of Hydroinformatics*, 15(2): 258–270.
- Raffin, A.; Hill, A.; Gleave, A.; Kanervisto, A.; Ernestus, M.; and Dormann, N. 2021. Stable-Baselines3: Reliable Reinforcement Learning Implementations. *Journal of Machine Learning Research*, 22(268): 1–8.
- Rutherford, A.; Ellis, B.; Gallici, M.; Cook, J.; Lupu, A.; Ingvarsson, G.; Willi, T.; Khan, A.; Schroeder de Witt, C.; Souly, A.; Bandyopadhyay, S.; Samvelyan, M.; Jiang, M.; Lange, R.; Whiteson, S.; Lacerda, B.; Hawes, N.; Rocktäschel, T.; Lu, C.; and Foerster, J. 2024. Jax-MARL: Multi-Agent RL Environments and Algorithms in JAX. In *Proceedings of the 23rd International Conference on Autonomous Agents and Multiagent Systems, AAMAS '24*, 2444–2446. Richland, SC: International Foundation for Autonomous Agents and Multiagent Systems. ISBN 9798400704864.
- Seo, H.; and Lee, D. 2017. Reinforcement learning and strategic reasoning during social decision-making. In *Decision Neuroscience*, 225–231. Elsevier.
- Sutton, R. S.; and Barto, A. G. 1998. *Reinforcement Learning: An Introduction*. MIT Press.
- Taylor, M. E.; Nissen, N.; Wang, Y.; and Navidi, N. 2023. Improving reinforcement learning with human assistance: an argument for human subject studies with HIPPO Gym. *Neural Computing and Applications*, 35(32): 23429–23439.
- Terry, J.; Black, B.; Grammel, N.; Jayakumar, M.; Hari, A.; Sullivan, R.; Santos, L. S.; Dieffendahl, C.; Horsch, C.; Perez-Vicente, R.; et al. 2021. Pettingzoo: Gym for multi-agent reinforcement learning. *Advances in Neural Information Processing Systems*, 34: 15032–15043.
- Towers, M.; Kwiatkowski, A.; Terry, J.; Balis, J. U.; De Cola, G.; Deleu, T.; Goulao, M.; Kallinteris, A.; Krimmel, M.; KG, A.; et al. 2024. Gymnasium: A standard interface for reinforcement learning environments. *arXiv preprint arXiv:2407.17032*.
- Vamplew, P.; Dazeley, R.; Berry, A.; Issabekov, R.; and Dekker, E. 2011. Empirical evaluation methods for multiobjective reinforcement learning algorithms. *Machine learning*, 84: 51–80.
- Yuan, Y.; Hao, J.; Ma, Y.; Dong, Z.; Liang, H.; Liu, J.; Feng, Z.; Zhao, K.; and Zheng, Y. 2024. Uni-RLHF: Universal Platform and Benchmark Suite for Reinforcement Learning with Diverse Human Feedback. In *The Twelfth International Conference on Learning Representations, ICLR*.
- Zhang, Y.; Yang, S.; Bai, C.; Wu, F.; Li, X.; Wang, Z.; and Li, X. 2024. Towards efficient llm grounding for embodied multi-agent collaboration. *arXiv preprint arXiv:2405.14314*.
- Zhao, Y.; Wang, Z.; Zhu, C.; and Wang, S. 2021. Efficient dialogue complementary policy learning via deep q-network policy and episodic memory policy. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, 4311–4323.
- Zheng, L.; Yang, J.; Cai, H.; Zhou, M.; Zhang, W.; Wang, J.; and Yu, Y. 2018. Magent: A many-agent reinforcement learning platform for artificial collective intelligence. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32.
- Zhu, C.; Dastani, M.; and Wang, S. 2024. A survey of multi-agent deep reinforcement learning with communication. *Autonomous Agents and Multi-Agent Systems*, 38(1): 4.
- Zhu, Z.; de Salvo Braz, R.; Bhandari, J.; Jiang, D.; Wan, Y.; Efroni, Y.; Wang, L.; Xu, R.; Guo, H.; Nikulkov, A.; et al. 2024. Pearl: A Production-Ready Reinforcement Learning Agent. *Journal of Machine Learning Research*, 25(273): 1–30.

Appendix A: Popular RL packages and their APIs

Name	Description	Language	Target	API
Amaze ¹	Maze navigation environments	Python	Research	Gymnasium+
CleanRL ²	Algorithm library	Python/Torch	Research/Production	Gymnasium
Gymnasium ³	De-facto standard environments	Python	Research	Gymnasium
HIPPO-Gym ⁴	Human Input Parsing	Python	Research	Gymnasium+
Interactive-Gym	Interactive web-based experiment platform	Python	Research	Gymnasium+
JaxMARL ⁵	Multi-agent environments	JAX	Research	Gymnasium+
Momaland ⁶	Multi-agent multi-objective environments	Python	Research	Gymnasium
MO-Gymnasium ⁷	Multi-objective environments	Python	Research	Gymnasium+
MORL Baselines ⁸	Multi-objective algorithm library	Python	Research	Gymnasium+
OpenSpiel ⁹	Game (theory) environments/algorithms	Python	Research	Gymnasium
Pearl ¹⁰	Algorithm library	Python	Production	Gymnasium
PettingZoo ¹¹	Multi-agent environments	Python	Research	Gymnasium+
Ray RLlib ¹²	Algorithm library	Python/Torch	Production	Gymnasium
RLax ¹³	Algorithm library	JAX	Research	Gymnasium
Stable-baselines3 ¹⁴	Algorithm library	Python/Torch	Research/Production	Gymnasium
Uni-RLHF ¹⁵	Annotation tool	Python	Annotation	Gymnasium+

Table 2: Comparison of RL Packages, APIs, and Applications. + indicates minor alterations to the Gymnasium API such as a vectorial reward for multi-objective, or vectorial action for multi-agent RL. References are: 1- Godin-Dubois, Miras, and Kononova (2024); 2- Huang et al. (2022); 3- Brockman (2016); 4- Taylor et al. (2023); 5- Rutherford et al. (2024); 6-Felten et al. (2024); 7- Alegre et al. (2022); 8- Felten et al. (2023); 9- Lanctot et al. (2019); 10- Zhu et al. (2024); 11- Terry et al. (2021); 12- Liang et al. (2018); 13- DeepMind et al. (2020); 14- Raffin et al. (2021); 15-Yuan et al. (2024).