EDAP70 - Proposed projects

Davide Tateo

March 2026

1 Software engineering for an Open Source RL library

In this project, the objective is to work with the MushroomRL [4] library (`https://github.com/MushroomRL/mushroom-rl`), a Torch-based reinforcement learning library designed specifically for research and higher education.

The library has been in development for many years, and recently we have been developing version 2.0. The new version of the library supports parallel environments and recurrent policies. Unfortunately, the new setup is causing memory issues for the DQN algorithm [8] and its variants, specifically when dealing with images as input and when using histories of multiple observations. This happens because the new MushroomRL implementation doesn't support the concept of LazyFrames anymore, which allowed to reduce the memory requirements by avoiding keeping in memory multiple copies of the same images.

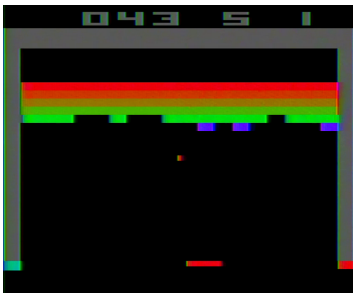

Figure 1: The breakout Atari game, serving as the main benchmark for DQN

The main objective of the project is to reduce the memory usage of the current implementation for DQN and variants. Ideally, we should look at either a novel solution for efficient data usage using the current data structures of MushroomRL, or use existing solutions (LazyFrames or other approaches from the literature) to solve the issue. While this work gives you experience in Reinforcement Learning, the focus is mostly on the software engineering part. Specifically, the **minimal objectives** are:

- Implementation of efficient memory management of a sequence of image frames
- Test of the DQN algorithm in the Atari games setting
- Documentation and testing of the code

The **ideal objectives** are:

- Fix of the replay memory length bug (1 sample error) and test updates.
- Refactoring of the regressor interface, which needs to be simplified, specifically for DQN
- Full implementation of the list dataset backed, to support arbitrary input data types
- In-depth benchmarking of the DQN algorithms in Atari and/or other image-based tasks

The **requirements** for this project are:

- Good knowledge of Python
- Software engineering skills
- Willingness to contribute to the Open Source community. The student agrees to release the code under the MIT license.
- An Nvidia GPU able to perform scientific computations, ideally an RTX

If the code quality is high, the student may merge the code into the MushroomRL repository with a pull request. This will also make the student appear as a contributor to the library on GitHub.

2 Safe Reinforcement Learning of dynamic bimanual manipulation policies

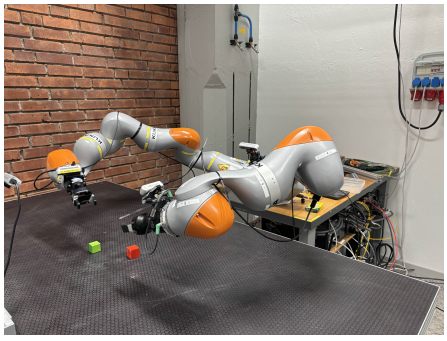

Figure 2: The real-world Kuka Iiwa bimanual setup, which must be replicated in simulation within this project

Safety is a fundamental property in robotics. In Reinforcement Learning, safety is even more fundamental, as it involves exploration and policy updates. Thus, in this setting, safety is not a static property to be analyzed; it becomes a dynamic problem that must be considered during both training and deployment. For this reason, the framework of Constrained Markov Decision Processes [2] has been designed, which extends the Reinforcement Learning setup to consider constraints and not only reward functions. While the reward functions must be maximized, constraints must be fulfilled either at the end of training (Safe Reinforcement Learning) or during the whole exploration process (Safe Exploration).

Among all the robotic tasks, dynamic bimanual manipulation tasks are particularly challenging, as it requires to coordinate two arms in the same workspace under fast motions. This makes learning under constraints in this setting particularly relevant and challenging.

The objective of this project is to explore this edge-case area of research. In the context of the project, the **minimal objectives** will be:

- Implementation of a digital twin of our bimanual manipulation setup (Two Kuka Iiwa robots) either in Isaac Sim, MJX, or other parallel simulators using the MushroomRL library [4].
- Design of a simple bimanual manipulation task (bimanual peg insertion)
- design of the reward function and safety constraints for the task.
- Implementation of the Lagrangian SAC [6] and of the D-ATACOM [5] approaches. The existing codebase will be provided.
- Testing of the task in the unconstrained setting and using the two abovementioned RL approaches.

The **ideal objectives** for the projects are:

- Implementation of a dynamic task (this may include throwing and catching objects, interaction with people, or fast motions such as serving a tennis ball with a racket)
- Full benchmarking with safe reinforcement learning [1, 9, 7, 10, 11] and safety filters [3] approaches in both settings.
- Introduction of torque-based safety constraints to allow contacts during manipulation.

The proposed project is extremely challenging, and the project load may be heavy, as it is pushing at the edge of current research on RL for manipulation. Therefore, consider carefully if you really want to apply to this project. The **requirements** for this project are:

- Good knowledge of the Python Language.
- Willingness to dive deep into the Reinforcement learning and safe Reinforcement Learning literature.
- Academic attitude/drive.

The project may be extended into a master's thesis and/or a scientific publication.

References

[1] Joshua Achiam, David Held, Aviv Tamar, and Pieter Abbeel. Constrained Policy Optimization. In *International Conference on Machine Learning (ICML)*, 2017.

[2] Eitan Altman. Constrained Markov Decision Processes with Total Cost Criteria: Lagrangian Approach and Dual Linear Program. *Mathematical methods of operations research*, 48(3):387–417, 1998.

[3] Gal Dalal, Krishnamurthy Dvijotham, Matej Vecerik, Todd Hester, Cosmin Paduraru, and Yuval Tassa. Safe exploration in continuous action spaces. *arXiv preprint arXiv:1801.08757*, 2018.

[4] Carlo D'Eramo, Davide Tateo, Andrea Bonarini, Marcello Restelli, and Jan Peters. Mushroomrl: Simplifying reinforcement learning research. *Journal of Machine Learning Research*, 22(131):1–5, 2021.

[5] Jonas Günster, Puze Liu, Jan Peters, and Davide Tateo. Handling long-term safety and uncertainty in safe reinforcement learning. In *Conference on Robot Learning (CoRL)*, 2024.

[6] Sehoon Ha, Peng Xu, Zhenyu Tan, Sergey Levine, and Jie Tan. Learning to walk in the real world with minimal human effort. In *Proceedings of the 2020 Conference on Robot Learning*, volume 155 of *Proceedings of Machine Learning Research*, pages 1110–1120. PMLR, 16–18 Nov 2021.

[7] Yongshuai Liu, Jiaxin Ding, and Xin Liu. Ipo: Interior-point policy optimization under constraints. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 34(04), pages 4940–4947, 2020.

[8] 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. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[9] Alex Ray, Joshua Achiam, and Dario Amodei. Benchmarking safe exploration in deep reinforcement learning.(2019). *URL https://cdn. openai. com/safexp-short. pdf*, pages 1–25, 2019.

[10] Adam Stooke, Joshua Achiam, and Pieter Abbeel. Responsive Safety in Reinforcement Learning by PID Lagrangian Methods. In *International Conference on Machine Learning (ICML)*, 2020.

[11] Qisong Yang, Thiago D Simão, Simon H Tindemans, and Matthijs TJ Spaan. Wcsac: Worst-case soft actor critic for safety-constrained reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35 (12), pages 10639–10646, 2021.