Autonomous Skill Acquisition for Robots Using Graduated Learning

Gautham Vasan
University of Alberta
Edmonton, Canada
vasan@ualberta.ca

ABSTRACT

Skill acquisition is among the most remarkable aspects of human intelligence. It involves discovering purposeful behavioural modules, retaining them as skills, honing them through practice, and applying them in unforeseen circumstances [11]. Skill acquisition underlies our ability to choose to spend time and energy on the mastery of particular tasks and draw upon previous experience to solve more complex problems over time with less cognitive effort [10]. If endowed with continual skill acquisition, robots can autonomously improve their skills over time, where learning at one stage of development is a foundation for future learning [23]. It could unlock new possibilities for physical automation with general-purpose robots, just as general-purpose computer processors ushered in the information age [24, 33]. In this work, we propose a novel approach called *Graduated Learning*, where we ask a robot to acquire new manipulation and locomotion skills repeatedly, using time-delineated experiences of attempts at those skills (i.e., episodes) and some store of previously acquired knowledge (e.g., weights of a neural network). Our proposed approach chooses the order in which an agent learns these skills since the progressive manner in which they are developed plays a vital role in developing a final skill set.

KEYWORDS

Reinforcement Learning, Robot Learning, Real-time learning

ACM Reference Format:
Gautham Vasan. 2024. Autonomous Skill Acquisition for Robots Using Graduated Learning. In *Proc.of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), Auckland, New Zealand, May 6 – 10, 2024*, IFAAMAS, 3 pages.

1 INTRODUCTION

Recent advances in Machine Learning have fueled a class of robots that can perform specialized tasks such as vacuum cleaning, assembly, welding and pick-and-place items in warehouses. However, most of these robots cannot accomplish much beyond what they are explicitly designed to do, often operating within a very narrow set of conditions. They require representations of the world in terms of objects, actions, and plans which are carefully hand-crafted or learned for the robot's task [25, 33]. Typical data-driven learning systems have distinct training and deployment phases. In order to adapt to changes in the real-world post-deployment (e.g., changes in lighting, wear and tear of hardware, and sensor calibration), the entire system is often rebuilt rather than making incremental changes to the model. The majority of commercially deployed robots rely heavily on human supervision and intervention when deployed out in the world [21, 25]. However, a general-purpose robot capable of solving a multitude of tasks cannot rely on representations hardwired before deployment because each new task may require a different representation. To build a truly general-purpose robot, there are four key challenges that need to be addressed.

Learning under real-time constraints. Firstly, general-purpose robots must have the ability to learn on-the-fly as they interact with the environment, commonly known as online learning. When online learning systems engage with the physical world, we call them real-time learning systems. Reinforcement Learning (RL) is a natural way of formulating real-time learning tasks [27]. Many deep RL methods, which use artificial neural networks with many layers, have been developed to solve complex motor control problems [1, 7, 26]. However, they do not easily extend to the real-time learning setting that operates under time and resource constraints, for example, in quadrotors and mobile robots [13]. While approaches including learning from demonstrations [6, 30], sim-to-real [4, 19], and offline RL [14] have been used to develop pre-trained agents, there has been relatively little interest in studying real-time learning in the real world.

How to specify a reinforcement learning task? Secondly, to solve unforeseen tasks, general-purpose robots cannot rely on a human instructor to provide domain knowledge beforehand. Many deep RL approaches rely on task-specific prior knowledge to carefully engineer a guiding reward signal, often biasing the solution that the agent can find [8, 22]. For many guiding reward tasks, there is a related sparse reward task (e.g., a reward of -1 for each time step until termination) that is much easier to specify and still captures the desired behaviour of agents. In addition, when the sparse reward formulation is used, the agent can discover novel and potentially superior solutions. However, this learning using sparse rewards is thought to be hard to solve [20]. Our interest lies in leveraging sparse reward formulations since they can provide a natural means to specify a sequence of tasks either given by humans at a high level or automatically generated for a continual learning agent.

Developing efficient, scalable policy learning methods. Thirdly, deep RL methods can be prohibitively expensive in terms of computational resource requirements, especially on complex vision-based control tasks. As onboard computing is a scarce resource in mobile robots, learning systems must be designed such that the computational platform can meet real-time requirements [13]. While state-of-the-art deep RL methods achieve effective performance, they

Proc.of the 23rd International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2024), N. Alechina, V. Dignum, M. Dastani, J.S. Sichman (eds.), May 6 – 10, 2024, Auckland, New Zealand. © 2024 Association for Computing Machinery.

require computationally intensive, large batch gradient updates, which are not suitable for real-time learning [21, 34]. In contrast, incremental RL methods are computationally cheap and perfectly suited for real-time learning [5, 29]. However, their learning performance is not adequate to solve complex control tasks. We deem a learning method scalable if the computation or memory usage can be increased or decreased depending on the availability of resources without affecting the final performance after learning for a sufficient amount of time. For onboard learning, we require novel, computationally cheap learning methods that may learn slower than existing methods, but they could catch up with sufficient time.

Skill discovery. Fourthly, we lack methods to effectively discover real-life skills by learning from scratch in complex vision-based robot tasks. There are several methods for discovering skills in simple, simulated environments [2, 3, 9, 11, 17, 31], but they cannot be extended to the real-time learning setting. While [15] learns diverse primitive skills on a Daisy hexapod robot, they parameterize the primitives as simple cyclic movements and solve only a simple, non-visual locomotion task.

2 THE GRADUATED LEARNING FRAMEWORK

I present the idea of graduated learning, where we repeatedly ask a robot to acquire new manipulation and locomotion skills, using time-delineated experiences of attempts at those skills (i.e., episodes) and some store of previously acquired knowledge (e.g., weights of a neural network). Such a learning agent will solve only minimum-time problems. Based on its current behaviour, the agent determines which events are not so frequent or rare, called flow events. The agent makes flow events goal states and learns to reach those goal states as soon as possible. Once the agent learns to effectively reach a flow goal state, we store the policy network parameters for future use. As the agent solves more such tasks, newer events become flow events to the agent. The agent could also learn a high-level controller that can reuse previously learned skills to solve a task.

3 EXPERIMENTAL APPROACH

We will use the iRobot Create2 vacuum robot, Anki Vector, Franka Emika Panda Arm and UR5 industrial robot arm as physical testbeds given our extensive experience prototyping applications and publishing research papers with them [12, 16, 29, 32]. Since experiments are costly to perform on real robots, we will also use simulated experiments on Gazebo and MuJoCo [28]. Simulated experiments would only serve as litmus tests to rule out ineffective approaches.

Milestone 1: Real-time reinforcement learning. Recently, we developed a real-time learning system called the Remote-Local Distributed (ReLoD) system to make resource-intensive RL algorithms tractable on resource-limited computers [32]. A common setup for a robotic agent is to have two different computers simultaneously: a resource-limited local computer tethered to the robot and a powerful remote computer connected wirelessly. Given such a setup, it is unclear to what extent the performance of a learning system can be affected by resource limitations and how to efficiently use the wirelessly connected powerful computer to compensate for any performance loss. In this paper, we implemented a real-time learning system to distribute computations of two deep reinforcement learning (RL) algorithms, Soft Actor-Critic (SAC) and Proximal Policy Optimization (PPO), between a local and a remote computer.

Milestone 2: Task Specification. In a recent submission, we re-evaluated sparse rewards for task specification in RL, which are often overlooked due to their perceived difficulty and lack of informativeness compared to dense rewards. Sparse rewards, however, offer simplicity in specification, for example, for a sequence of tasks in continual learning. Our studies contrasted the two reward paradigms, revealing that sparse rewards not only facilitate learning higher-quality policies but also surpass dense-reward-based policies on their own performance metrics. We demonstrated that sparse-reward formulations can lead to effective learning in a reasonable timeframe on benchmark tasks. Crucially, we identified the goal-hit rate of the initial policy as a robust early indicator for learning success in sparse-reward settings. By applying our insights to vision-based reaching tasks on four distinct robotic platforms, we show that robots can learn without pre-training and from raw pixels in two to three hours with sparse rewards. Our findings advocate for a re-evaluation of sparse rewards to simplify reward design and achieve higher-quality policies in real robot learning. Our video demo can be found here: https://youtu.be/rrbqbbdmvkM

Milestone 3: A novel, efficacious incremental learning method. As the agent will learn thousands of tasks or even more over time, we need to develop lightweight computationally and memory-efficient agents that can maintain all these tasks in our computers. I'm working on an effective, novel incremental learning method based on the entropy regularized RL objective proposed by [35]. It can be used to learn multiple control policies in parallel, and all the relevant data structures can be stored with ease. This work will be submitted to NeurIPS 2024.

Milestone 4: Detecting flow events. How to detect flow events which are necessary to generate a task and choose a task among available flow tasks that best helps solve the overarching task? I will survey the literature for effective, automatic curriculum generation [18] and option discovery [2, 3] to answer this question. Subsequently, I'll draw upon these ideas to build agents capable of detecting flow events and choosing the most promising flow event to help solve the target task.

Milestone 5: Composition of learned skills. We will assess the effectiveness of a high-level controller that can select among learned flow event policies to generate abstract motor behaviours to solve complex vision-based tasks on multiple robots. Here, we can draw inspiration from [8, 17, 22].

4 IMPACT

Autonomous skill acquisition is crucial for building general-purpose robots that can operate in unstructured environments and safely share spaces with humans. The ability to learn versatile skills and adapt to unseen situations would help robots function independently in hazardous environments, assist in healthcare, and reduce the impact of labour shortages around the world. My research will push the boundaries of robot learning, thus bringing us a step closer towards building truly intelligent machines.

REFERENCES

[1] Abbas Abdolmaleki, Jost Tobias Springenberg, Yuval Tassa, Remi Munos, Nicolas Heess, and Martin Riedmiller. 2018. Maximum a posteriori policy optimisation. *arXiv preprint arXiv:1806.06920* (2018).

[2] Pierre-Luc Bacon. 2018. *Temporal Representation Learning*. McGill University (Canada).

[3] Akhil Bagaria and George Konidaris. 2019. Option discovery using deep skill chaining. In *International Conference on Learning Representations*.

[4] Konstantinos Bousmalis, Alex Irpan, Paul Wohlhart, Yunfei Bai, Matthew Kelcey, Mrinal Kalakrishnan, Laura Downs, Julian Ibarz, Peter Pastor, Kurt Konolige, et al. 2018. Using simulation and domain adaptation to improve efficiency of deep robotic grasping. In *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 4243–4250.

[5] Thomas Degris, Patrick M Pilarski, and Richard S Sutton. 2012. Model-free reinforcement learning with continuous action in practice. In *2012 American Control Conference (ACC)*. IEEE, 2177–2182.

[6] Abhishek Gupta, Clemens Eppner, Sergey Levine, and Pieter Abbeel. 2016. Learning dexterous manipulation for a soft robotic hand from human demonstrations. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 3786–3793.

[7] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. 2018. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*. PMLR, 1861–1870.

[8] Tim Hertweck, Martin Riedmiller, Michael Bloesch, Jost Tobias Springenberg, Noah Siegel, Markus Wulfmeier, Roland Hafner, and Nicolas Heess. 2020. Simple sensor intentions for exploration. *arXiv preprint arXiv:2005.07541* (2020).

[9] Ryan Julian, Eric Heiden, Zhanpeng He, Hejia Zhang, Stefan Schaal, Joseph Lim, Gaurav Sukhatme, and Karol Hausman. 2020. Scaling simulation-to-real transfer by learning composable robot skills. In *Proceedings of the 2018 International Symposium on Experimental Robotics*. Springer, 267–279.

[10] Andrej Karpathy and Michiel Van De Panne. 2012. Curriculum learning for motor skills. In *Advances in Artificial Intelligence: 25th Canadian Conference on Artificial Intelligence, Canadian AI 2012, Toronto, ON, Canada, May 28-30, 2012. Proceedings 25*. Springer, 325–330.

[11] George Dimitri Konidaris. 2011. *Autonomous robot skill acquisition*. University of Massachusetts Amherst.

[12] Dmytro Korenkevych, A Rupam Mahmood, Gautham Vasan, and James Bergstra. 2019. Autoregressive policies for continuous control deep reinforcement learning. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. 2754–2762.

[13] Srivatsan Krishnan, Behzad Boroujerdian, William Fu, Aleksandra Faust, and Vijay Janapa Reddi. 2021. Air learning: a deep reinforcement learning gym for autonomous aerial robot visual navigation. *Machine Learning* 110 (2021), 2501–2540.

[14] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. 2020. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643* (2020).

[15] Tianyu Li, Nathan Lambert, Roberto Calandra, Franziska Meier, and Akshara Rai. 2020. Learning generalizable locomotion skills with hierarchical reinforcement learning. In *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 413–419.

[16] A Rupam Mahmood, Dmytro Korenkevych, Gautham Vasan, William Ma, and James Bergstra. 2018. Benchmarking reinforcement learning algorithms on real-world robots. In *Conference on robot learning*. PMLR, 561–591.

[17] Josh Merel, Saran Tunyasuvunakool, Arun Ahuja, Yuval Tassa, Leonard Hasenclever, Vu Pham, Tom Erez, Greg Wayne, and Nicolas Heess. 2020. Catch & carry: reusable neural controllers for vision-guided whole-body tasks. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 39–1.

[18] Sanmit Narvekar, Bei Peng, Matteo Leonetti, Jivko Sinapov, Matthew E Taylor, and Peter Stone. 2020. Curriculum learning for reinforcement learning domains: A framework and survey. *The Journal of Machine Learning Research* 21, 1 (2020), 7382–7431.

[19] Xue Bin Peng, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. 2018. Sim-to-real transfer of robotic control with dynamics randomization. In *2018 IEEE international conference on robotics and automation (ICRA)*. IEEE, 3803–3810.

[20] Matthias Plappert, Marcin Andrychowicz, Alex Ray, Bob McGrew, Bowen Baker, Glenn Powell, Jonas Schneider, Josh Tobin, Maciek Chociej, Peter Welinder, et al. 2018. Multi-goal reinforcement learning: Challenging robotics environments and request for research. *arXiv preprint arXiv:1802.09464* (2018).

[21] Aravind Rajeswaran, Chelsea Finn, Sham M Kakade, and Sergey Levine. 2019. Meta-learning with implicit gradients. *Advances in neural information processing systems* 32 (2019).

[22] Martin Riedmiller, Roland Hafner, Thomas Lampe, Michael Neunert, Jonas Degrave, Tom Wiele, Vlad Mnih, Nicolas Heess, and Jost Tobias Springenberg. 2018. Learning by playing solving sparse reward tasks from scratch. In *International conference on machine learning*. PMLR, 4344–4353.

[23] Mark Bishop Ring et al. 1994. Continual learning in reinforcement environments. (1994).

[24] Geordie Rose. 2022. The Case for General-Purpose Robots Over Special-Purpose Robots. https://sanctuary.ai/resources/news/the-case-for-general-purpose-robots-over-special-purpose-robots/

[25] Nicholas Roy, Ingmar Posner, Tim Barfoot, Philippe Beaudoin, Yoshua Bengio, Jeannette Bohg, Oliver Brock, Isabelle Depatie, Dieter Fox, Dan Koditschek, et al. 2021. From machine learning to robotics: challenges and opportunities for embodied intelligence. *arXiv preprint arXiv:2110.15245* (2021).

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

[27] Richard S Sutton and Andrew G Barto. 2018. *Reinforcement learning: An introduction*. MIT press.

[28] Emanuel Todorov, Tom Erez, and Yuval Tassa. 2012. Mujoco: A physics engine for model-based control. In *2012 IEEE/RSJ international conference on intelligent robots and systems*. IEEE, 5026–5033.

[29] Gautham Vasan. 2017. Teaching a powered prosthetic arm with an intact arm using reinforcement learning. (2017).

[30] Gautham Vasan and Patrick M Pilarski. 2017. Learning from demonstration: Teaching a myoelectric prosthesis with an intact limb via reinforcement learning. In *2017 International Conference on Rehabilitation Robotics (ICORR)*. IEEE, 1457–1464.

[31] Yi Wan and Richard S Sutton. 2022. Toward Discovering Options that Achieve Faster Planning. *arXiv preprint arXiv:2205.12515* (2022).

[32] Yan Wang, Gautham Vasan, and A Rupam Mahmood. 2023. Real-time reinforcement learning for vision-based robotics utilizing local and remote computers. In *2023 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 9435–9441.

[33] KR Zentner, Ryan Julian, Ujjwal Puri, Yulun Zhang, and Gaurav S Sukhatme. 2021. A simple approach to continual learning by transferring skill parameters. *arXiv preprint arXiv:2110.10255* (2021).

[34] Henry Zhu, Justin Yu, Abhishek Gupta, Dhruv Shah, Kristian Hartikainen, Avi Singh, Vikash Kumar, and Sergey Levine. 2020. The ingredients of real-world robotic reinforcement learning. *arXiv preprint arXiv:2004.12570* (2020).

[35] Brian D Ziebart. 2010. *Modeling purposeful adaptive behavior with the principle of maximum causal entropy*. Carnegie Mellon University.