Do As You Teach: A Multi-Teacher Approach to Self-Play in Deep Reinforcement Learning

Chaitanya Kharyal
Microsoft
chaitanyajee@gmail.com

Tanmay Kumar Sinha
Microsoft Research
tanmaysinha18@gmail.com

Sai Krishna Gottipati
AI Redefined
sai@ai-r.com

Srijita Das
University of Alberta, Amii
srijita1@ualberta.ca

Matthew E. Taylor
University of Alberta, Alberta Machine Intelligence Institute (Amii)
matthew.e.taylor@ualberta.ca

Abstract

A long-running challenge in the reinforcement learning (RL) community has been to train a goal-conditioned agent in a sparse reward environment such that it could also generalize to other unseen goals. Empirical results in Fetch-Reach and a novel driving simulator demonstrate that our proposed algorithm, Multi-Teacher Asymmetric Self-Play, allows one agent (i.e., a *teacher*) to create a successful curriculum for another agent (i.e., the *student*). Surprisingly, results also show that training with multiple teachers actually helps the student learn faster. Our analysis shows that multiple teachers can provide better coverage of the state space, selecting diverse sets of goals, and better helping a student learn. Moreover, results show that completely new students can learn offline from the goals generated by teachers that trained with a previous student. This is crucial in the context of industrial robotics where repeatedly training a teacher agent is expensive and sometimes infeasible.

1 Introduction

The future of industrial automation is hinged on the ability of the industrial robots to precisely finish the tasks designated for them. These tasks are usually specified in terms of a state the robot is required to reach i.e, a goal state. Goal-conditioned RL Schaul et al. (2015); Pitis et al. (2020) is an emerging sub-field that trains policies with goal inputs. This enables the agent to generalize to new unseen goals, learn multiple tasks and acquire new skills along the way. This area is particularly relevant for complex industrial settings like cooking robots Bollini et al. (2013), manipulator hands Morrison et al. (2018), robotic surgeries Su et al. (2021) which requires the robot to learn multiple tasks and achieve a variety of goals. Some of these applications are very delicate and a minor mistake (e.g, in the case of a robotic surgery) can be fatal. Therefore, the agents are required to precisely reach the designated goal states. This necessity for precision prompts the RL practitioner to design the reward function in such a way that the agent gets rewarded only if it reaches the exact goal state, thus making the reward function sparse.

Training a goal conditioned RL agent in sparse reward environments that could generalize well to other unseen goals has been a long lasting challenge. While several exploration based methods are proposed Bellemare et al. (2016b); Houthooft et al. (2016); Burda et al. (2018), they all try to optimize for a specific objective (e.g, information theoretic, intrinsic reward, count based etc.). It remains unclear whether these align with the actual objective of the goal-conditioned agent and if the same algorithm can work on a wide range of environments. There are other curriculum learning based methods Campero et al. (2021b); Florensa et al. (2018) where a teacher generates a curriculum of goals for the student agent. The student learns to solve harder goals in a progressive way after solving easier goals. However, these methods are mostly limited to single teachers and hence, difficult to cover goals in the entire state-space especially for robotics.

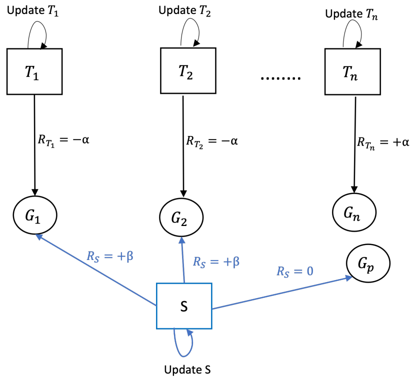


Figure 1: Framework for Multi Teacher Asymmetric Self-play

In this work, we propose a novel approach to generate an automatic curriculum of increasingly challenging goals by multiple teachers for a goal-conditioned student agent. All the agents in our proposed approach are learning agents starting from scratch. The intuition behind this idea is that multiple teachers would cover larger parts of the state-space, hence the goal-distribution for the student agent would be closer to the true goal distribution. Additionally, more coverage would lead to the teachers suggesting diverse goals. We hypothesize that these crucial factors would make the student learn faster and better in challenging robotics tasks as demonstrated by our experiments. The multiple teacher agents could further be deployed post training as teachers with specific skill-set providing assistance to other robotics tasks. While investigating the effectiveness of our proposed algorithm, we noticed a lack of moderately complex goal conditioned environments that can be used to thoroughly test for statistical significance with a reasonable amount of computation. We therefore propose a novel (open source) driving simulator environment that can be used to further the research in this area. We elaborate on the effectiveness of this environment in Section-4 and provide the corresponding experimental results validating our algorithm and simulator.

Contributions: The key contributions of this work include (1) introducing a novel multi-teacher self-play algorithm that outperforms the existing state-of-the-art methods (2) validating that the improvement in performance is directly proportional to the number of teachers and the diversity of goals generated by these multiple teachers (3) introducing a novel (open source) sparsely rewarded goal-conditioned environment that can be used to test several hypotheses without the industry-scale compute. Our empirical results on two robotics domains demonstrates the effectiveness of this novel approach.

2 Related Work

Sparse reward environments can be quite difficult for RL agents — intrinsic motivation has been shown to be one quite helpful heuristic Gottipati et al. (2019). For instance, methods use measures like the novelty of states visited Bellemare et al. (2016a); Lopes et al. (2012), empowerment Mohamed & Rezende (2015), uncertainty in environment dynamics Houthooft et al. (2016); Schmidhuber (1991); Pathak et al. (2017), or impact on the state representation Raileanu & Rocktäschel (2020) to guide the agent to explore the environment. As a result, the agent ends up exploring "interesting" parts of the state-space to gather useful information about the task. Another class of methods encourage the agent to explore parts of the state space with high uncertainty Osband et al. (2016); Janz et al. (2019); Metelli et al. (2019). Lastly, other methods Oh et al. (2018); Zha et al. (2021) use the agent's good experiences to build an imitation learning model to guide exploration.

A more recent approach is to use another agent (i.e, a teacher) to generate an automatic curriculum of intermediate goals for the student agent, thus giving intermediate reinforcements. Asymmetric Self-Play (ASP) Sukhbaatar et al. (2018) defines a minimax game between the teacher and student — the teacher demonstrates increasingly challenging goals and the student tries to reach these goals. In

ASP, both the teacher and the student are similarly parameterized and represented by neural networks. This setup was extended by Plappert et al OpenAI et al. (2021) by including a behaviour cloning loss for the student to imitate the teacher and thus solving complex robotics tasks. Du et al. Du et al. (2022) uses four players (two goal generators and two goal-conditioned students) in a mixed setting (friendly and adversarial) to find a productive balance between feasible and challenging goals for the goal-conditioned student.

Campero et al. Campero et al. (2021a) propose an adversarial setup for discrete domains by training a teacher to suggest challenging goals for the goal-conditioned student agent. Florensa et al. Florensa et al. (2018) used generative adversarial networks where a generator (teacher) generates goals of appropriate difficulty and a discriminator tries to identify the difficulty of the goals. A similar setter-solver framework was proposed by Racaniere et al. Racaniere et al. (2019), where an agent sets goals that are valid, feasible, and have high coverage for the goal-conditioned student.

There is a long literature of using multiple suboptimal teachers to improve the sample-efficiency of RL agent. Early approaches include aggregating multiple teacher models into a single mixture model Jacobs et al. (1991) and combining multiple independent teacher's action distributions Hinton (2002). There are methods in imitation learning that learns from multiple suboptimal teacher policies by framing it as a bi-level optimization of learning teacher expertise and learner policy Zhang et al. (2021) and reduction to online learning Cheng et al. (2020). Other methods employ ways to estimate the teacher's value function to identify the best teacher at every step in order to match the learner's performance Li et al. (2019a;b); Kurenkov et al. (2019). Model combination approaches Gimelfarb et al. (2018) have also been used in literature to learn from multiple teachers by combining their individual models.

The novel method we propose in the next section can be thought of as using multiple teachers to assist with exploration. It is most related to ASP, but we use multiple teachers. These teachers are also demonstrators — their trajectories can help the student learn faster. Finally, our method does not assume the teachers have any expertise and must learn to propose goals on the fly, influenced by the student's learning performance.

3 MULTI-TEACHER CURRICULUM LEARNING

Our approach is built upon the framework of a standard markov decision process (MDP) defined by $(\mathcal{S}, \mathcal{A}, R, \mathcal{P}, \gamma)$where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, R is the reward function , $\mathcal{P}$ is the transition probability and $\gamma \in [0, 1)$ is the discount factor . At any time step t of the episode, the agent is in state $s_t \in \mathcal{S}$, computes an action $a_t \in \mathcal{A}$ based on its policy π and current observation o_t i.e, $a_t \sim \pi(\cdot|o_t)$. In the case of a goal conditioned agent, the policy is also a function of the desired goal state g i.e, $a_t \sim \pi(\cdot|o_t, g)$. The agent then goes to the next state $s_{t+1} \in \mathcal{S}$ based on the transition probability function $\mathcal{P}$ and receives a reward $r_t \in R$. The goal of an RL algorithm is to maximize the expected discounted sum of rewards by optimizing its policy π.

We train two types of agents — *teacher(s)* and a *student*. In each teacher-student rollout, we sample a starting state $s_0 \in \mathcal{S}$. From this state, the teacher's policy $\pi_T(a \mid s)$ selects actions over time and eventually reaches its final state, $g \in \mathcal{S}$. Again, starting from the same state s_0, the student, with its goal conditioned policy $\pi_S(a \mid s, g)$, interacts with the environment and tries to reach the goal state set by the teacher, g_t. The teacher should aim to generate increasingly difficult goals that form a curriculum so that the student learns to reach a wide variety of difficult goal states. When successful, this will allow the student to reach any arbitrary goal thrown at it. We also train a third type of agent — *intern* that learns based on the goals set by a teacher while the teacher is training its corresponding student agent. It doesn't have access to the teacher's demonstrations.

There are two reasons why this approach may be helpful for student learning. Firstly, we know by construction that g_t is reachable from the starting state s_0. Moreover, the teacher provides a valid demonstration from s_0 to g_t that can be used to enhance the student's learning through behavioural cloning. Secondly, when the teacher's reward is set to be a function of the student's ability, the teacher learns to set goals that are incrementally difficult for the student.

While traditional curriculum learning methods typically have either a fixed curriculum or a single teacher agent, we consider the case with multiple teachers. In our experiments, we observe that a single teacher agent does not provide as diverse a set of goals as multiple teachers. Thus, a student trained with multiple teachers is able to better generalize to unseen goals (compared to a single teacher, or no teacher at all). In each rollout, one of the teacher agents sets the goal and the student tries to reach that goal. We repeat this until each teacher agent has set m goals. Once the rollout data is collected, we update the model parameters for each of the agents.

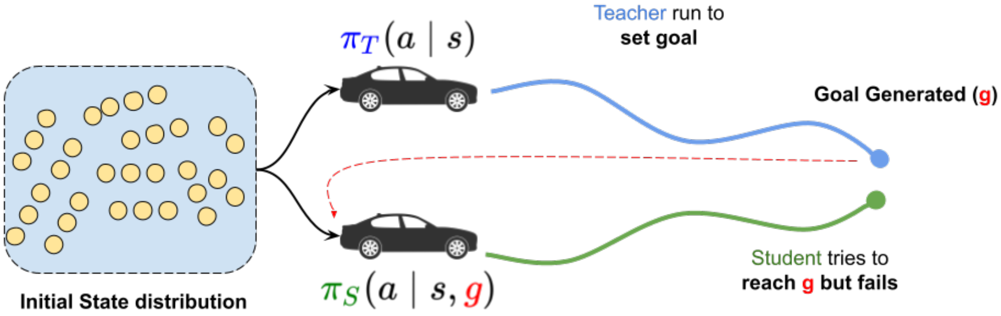


Figure 2: One teacher-student rollout of our algorithm.

Reward: We assign rewards to both the teacher and student agents denoted by R_T and R_S respectively, based on whether the student is able to reach the goal set by a teacher. If the student reaches the goal set by a teacher, that teacher gets a single negative reward and the student gets a single positive reward. Otherwise, when the student does not reach the goal, the teacher gets a positive reward and the student gets a reward of 0. Furthermore, the notion of invalid goals can be added to this reward structure by giving the teacher a large negative reward for setting an invalid goal. We have tested this invalid goal hypothesis on the fetch reach environment, and the results suggest that the student doesn't need any other training signal to avoid the invalid states.

$$R_T = \begin{cases} -\alpha & \text{if } g_t \text{ is valid and student reaches } g_t \\ +\alpha & \text{if } g_t \text{ is valid and student does not reach } g_t \\ -\gamma & \text{if } g_t \text{ is invalid} \end{cases}$$

$$R_S = \begin{cases} +\beta & \text{if student reaches } g_t \\ 0 & \text{if student does not reach } g_t \end{cases}$$

where, $\alpha, \beta, \gamma > 0$ and $\gamma > \alpha$.

Loss function: To assist student learning, we incorporate a behavioural cloning loss ($\mathcal{L}_{BC}$) for the student, in addition to the actor and critic loss functions used in a typical RL algorithm like TD3 Fujimoto et al. (2018)

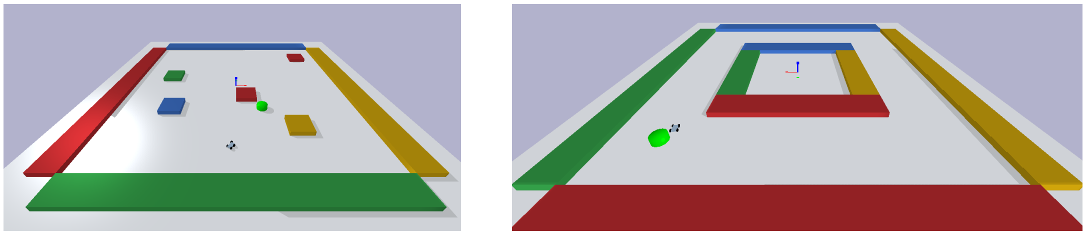
$$\mathcal{L}_{BC} = \mathbb{E}_{(s_t,g_t)\sim D_S}\left[\|\pi_S(a \mid s_t, g_t) - \pi_T(a \mid s_t)\|^2\right]$$

where D_S refers to the student's minibatch sampled during training.

Algorithm: The proposed multi-teacher algorithm is detailed in Algorithm 1. We denote the n teacher agents as T_1, T_2,... T_n and the student agent as S. Consequently we represent the parameters of actor and critic networks of teacher agents with $\theta_{T_1}, \cdots, \theta_{T_n}$ and that of the student agent with θ_S. In every "episode," m times for every teacher agent, we do a rollout of the teacher agent wherein the teacher agent interacts with the environment to set a goal (line 5), followed by a rollout of the student agent, where the student attempts to reach the goal set by the previous teacher (line 7). After $(n \times m)$ student-teacher rollouts (as shown in Figure-1) we update the parameters of each teacher agent based on the actor and critic loss functions, like in a standard RL algorithm like TD3 and the student agent is updated with both a behavior cloning loss and the standard TD3 losses (lines 10-11). We repeat this loop for a fixed number of episodes.

Algorithm 1: Multi Teacher Asymmetric Self-play

```
Data: $N, m$ ;                              //Number of teacher agents, multiplier
Data: $\theta_{T_1}, \cdots, \theta_{T_n}, \theta_S$ ;                    //Parameters for the agents
for episode = $1, 2, \cdots$ do
    for trial = $1, 2, \cdots, N \cdot m$ ;                                  //Rollouts
    do
        $k = \lfloor i/m \rfloor$;
        $k^{th}$ teacher sets goal;
        if goal is valid then
            Student tries to achieve the goal;
    for $i = 1, 2, \cdots, N \cdot m$ do
        $k = \lfloor i/m \rfloor$;
        Update $\theta_{T_k}$ ;                                               //RL Loss
        Update $\theta_S$ ;                                            //RL and BC Loss
```



(a) Block Task

(b) Racetrack Task

Figure 3: Some of the tasks offered by the Driving Environment

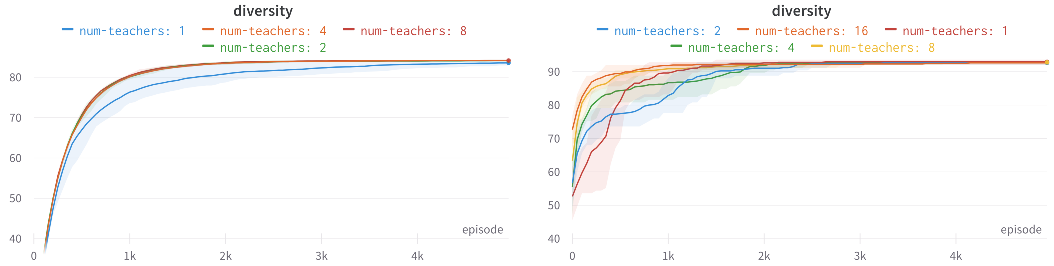
4 Experimental Results

The evaluation of our algorithm aims to answer the following research questions:

R1: Does the algorithm with multiple teachers perform better than single teacher ?
R2: Does the teachers provide better coverage of the state-space or offer diverse goals?
R3: Can an intern agent learn from scratch using the goals generated by the teachers for a different student?

We tested our hypothesis on two different environments: FetchReach and a novel driving simulator that we introduced. The driving simulator is a goal conditioned and resettable environment developed using PyBullet Coumans & Bai (2016–2021), and is a continuous action based maze. This maze-type structure ensures that the agent has to do difficult maneuvering in order to reach the goal, which would be improbable without proper training, thereby, making it a good test-bed for curriculum learning research. The observation space, for the driving simulator, consists of an observation vector, which has information about the position, orientation and velocity of the agent, and an occupancy map, which is a 75×75, three-channeled binary map: the first channel denotes the position of car, the second channel shows obstacles on the map, and the third shows the goal location. The action is a continuous 2D vector $[t, s]$, where $t \in [0, 1]$ is the throttle and $s \in [-0.6, 0.6]$ is the steering angle (in radians). If the agent is able to reach the goal, it is given a +5 reward; otherwise, no reward is given. Due to this sparse reward structure, it is usually hard for an agent to learn to reach any random goal. Thus, it is an ideal environment to test if introducing a teacher agent can help the student learn faster. One can also test other hypotheses including effect of adding multiple teachers, diversity of the goals generated by the teachers, training of intern agents etc. We compare our approach against the most relevant work by Sukhnaatar et al. Sukhbaatar et al. (2018). In our results, the algorithm corresponding to number of teachers as 1 refers to this baseline. Some of the experiments are run using Cogment (Redefined et al. (2021)).

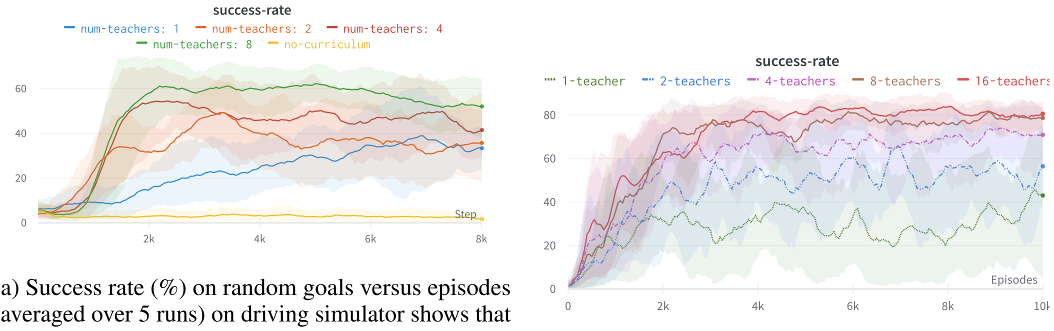
R1: Does the algorithm with multiple teachers perform better than single teacher ? To answer this, we show the performance of the student agent in terms of its ability to precisely reach a set of



(a) Driving Environment

(b) Fetch Reach

Figure 4: (a) Driving environment (b) Fetch reach, cumulative percentage of states visited by the teachers vs episode. to count the number of states visited, continuous (a) 2D (b) 3D state space is divided into a (a) 50 × 50 (b) 5 × 5 × 5 grid.



(a) Success rate (%) on random goals versus episodes (averaged over 5 runs) on driving simulator shows that generalizability increases with the increase in number of teacher agents. Moreover, we can note that the no-curriculum baseline (where random goals are shown to the student during training) fails to learn anything useful due to the sparse reward setting.

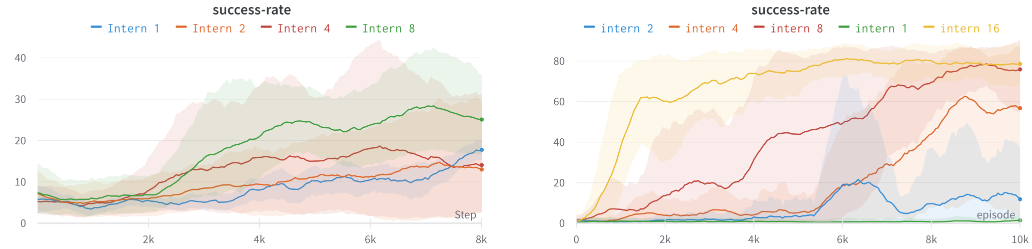
(b) Success rate on random goals versus episodes (averaged over 5 runs) on Fetch-reach environment. As earlier, increase in number of teacher agents leads to increase in generalization ability. However, this trend saturates after a certain number of teachers.

Figure 5: Success-rate

random goals in fetch reach and driving simulator respectively in Figure 5. All the results are averaged over five random seeds. In both the environments, we observe that the performance improves as we increase the number of teacher agents. To make sure that the students with different number of teachers see the same number of goals in one episode, we adjust the multiplier m accordingly (for example, for 16 teachers, we keep the multiplier 1, and for 1 teacher, we keep the multiplier 16).

R2: Does the teachers provide better coverage of the state-space or offer diverse goals? We hypothesize that the performance improvement of the student agent is because of the diversity of the goals generated by the multiple teacher agents (Figure 4). To test that, we first need to define the notion of diversity. While several prior works Cideron et al. (2020a); Masood & Doshi-Velez (2019); Eysenbach et al. (2019); Cideron et al. (2020b) have defined various diversity metrics and proposed algorithms for optimizing the diversity, a more appropriate metric in our case is in terms of the state space covered. We divide the state space into $n \times n$ equal parts and measure the number of parts covered by the goals generated so far. Our results show that the state-space coverage increases with increasing number of teachers (Figure 4).

R3: Can an intern agent learn from scratch using the goals generated by the teachers for a different student? We then further investigate whether the intern agent can learn based on the goals generated by the teacher agent while it was training a different student. We found that while the performance is worse compared to the student (i.e, it's online counterpart), the interns still managed to perform decently. When the goals were generated using more teachers, the goals were more diverse and hence the intern agent was able to learn much faster and better as compared to the intern trained on less diverse goals (generated by fewer teachers)(Figure 6). Furthermore, the diminishing distance between the success rates of the student and intern learners with increasing number of teachers



(a) Driving Environment

(b) Fetch Reach

Figure 6: Success rate of intern trained with different number of teachers on random goals. Intern n refers to an intern agent trained with goals from n teachers

(Figures 7 and 8) suggests that behaviour cloning is rendered less useful as the diversity of the goals increases.

As seen in Figure 6, for the fetch-reach and driving simulator environments, we notice that the performance of the intern increases with the number of teachers that generated goals. However, the performance improvement is only minimal when we increase the number of teachers beyond 8. This can be attributed to the complexity of the environment and the diversity of goals generated by the teachers.

5 Limitations and Future Work

Our algorithm has been validated only on simulations of robotics environments and might encounter unforeseen challenges while training on large-scale industrial robots. However, we hope to take advantage of recent progress in the field of sim2real transfer Kaspar et al. (2020) to mitigate some of these issues.

Future work also includes investigating the effect of using an explicit diversity component in teacher objectives and trying different diversity metrics. We would also like to decipher what these different teachers learn, for example in a robotics environment, is it possible to understand if any of the teachers propose goals specific to a sub-task?

References

Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. *Advances in neural information processing systems*, 29:1471–1479, 2016a.

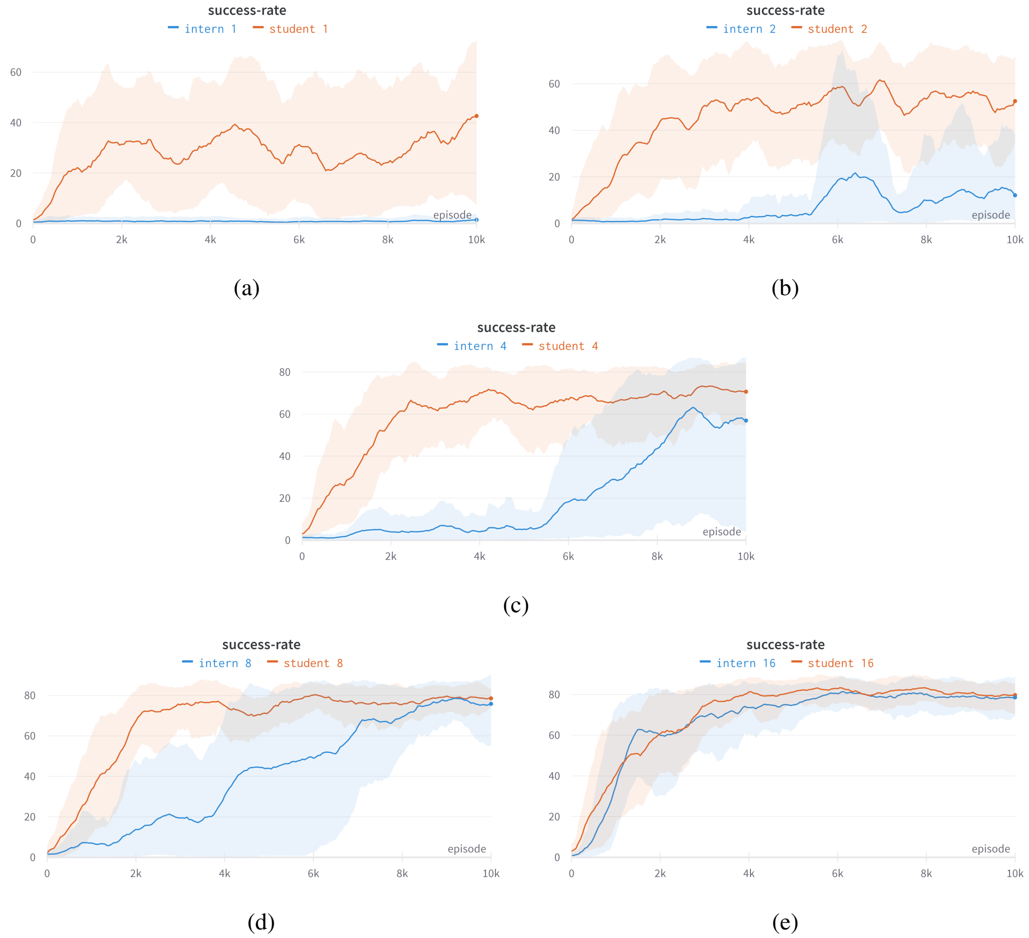
Marc G. Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Rémi Munos. Unifying count-based exploration and intrinsic motivation. *CoRR*, abs/1606.01868, 2016b. URL `http://arxiv.org/abs/1606.01868`.

Mario Bollini, Stefanie Tellex, Tyler Thompson, Nicholas Roy, and Daniela Rus. Interpreting and executing recipes with a cooking robot. In *Experimental Robotics*, pp. 481–495. Springer, 2013.

Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *arXiv preprint arXiv:1810.12894*, 2018.

Andres Campero, Roberta Raileanu, Heinrich Küttler, Joshua B. Tenenbaum, Tim Rocktäschel, and Edward Grefenstette. Learning with amigo: Adversarially motivated intrinsic goals. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net, 2021a. URL `https://openreview.net/forum?id=ETBc_MIMgoX`.

Andres Campero, Roberta Raileanu, Heinrich Küttler, Joshua B Tenenbaum, Tim Rocktäschel, and Edward Grefenstette. Learning with amigo: Adversarially motivated intrinsic goals. *ICLR*, 2021b.

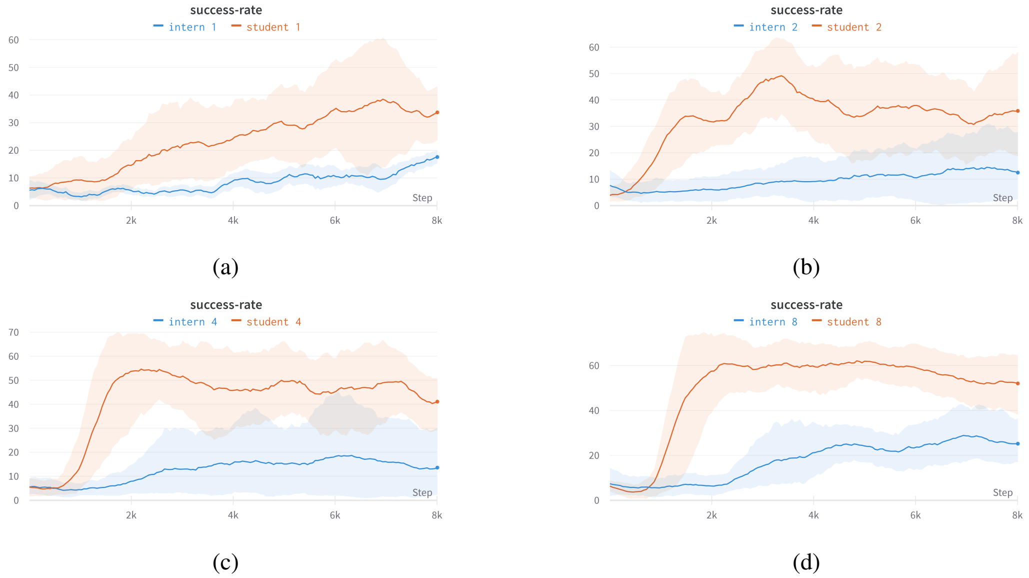


(a) (b)

(c)

(d) (e)

Figure 7: Direct comparison of success rates of students vs interns for the fetch reach environment for different number of teachers.



(a) (b)

(c) (d)

Figure 8: Direct comparison of success rates of students vs interns for the driving environment for different number of teachers.

Ching-An Cheng, Andrey Kolobov, and Alekh Agarwal. Policy improvement from multiple experts. *NeuRIPS*, 2020.

Geoffrey Cideron, Thomas Pierrot, Nicolas Perrin, Karim Beguir, and Olivier Sigaud. QD-RL: efficient mixing of quality and diversity in reinforcement learning. *CoRR*, abs/2006.08505, 2020a.

Geoffrey Cideron, Thomas Pierrot, Nicolas Perrin, Karim Beguir, and Olivier Sigaud. QD-RL: Efficient Mixing of Quality and Diversity in Reinforcement Learning. working paper or preprint, December 2020b.

Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. `http://pybullet.org`, 2016–2021.

Yuqing Du, Pieter Abbeel, and Aditya Grover. It takes four to tango: Multiagent selfplay for automatic curriculum generation. In *10th International Conference on Learning Representations, ICLR 2022*, pp. 1515–1528, 2022.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations*, 2019.

Carlos Florensa, David Held, Xinyang Geng, and Pieter Abbeel. Automatic goal generation for reinforcement learning agents. In *International conference on machine learning*, pp. 1515–1528. PMLR, 2018.

Scott Fujimoto, Herke van Hoof, and David Meger. Addressing function approximation error in actor-critic methods, 2018.

Michael Gimelfarb, Scott Sanner, and Chi-Guhn Lee. Reinforcement learning with multiple experts: A bayesian model combination approach. *NeurIPS*, 2018.

Sai Krishna Gottipati, Keehong Seo, Dhaivat Bhatt, Vincent Mai, Krishna Murthy, and Liam Paull. Deep active localization. *IEEE Robotics and Automation Letters*, 4(4):4394–4401, 2019. doi: 10.1109/LRA.2019.2932575.

Geoffrey E Hinton. Training products of experts by minimizing contrastive divergence. *Neural computation*, 2002.

Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. *Advances in neural information processing systems*, 29, 2016.

Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton. Adaptive mixtures of local experts. *Neural Computation*, 1991.

David Janz, Jiri Hron, Przemysław Mazur, Katja Hofmann, José Miguel Hernández-Lobato, and Sebastian Tschiatschek. Successor uncertainties: exploration and uncertainty in temporal difference learning. *Advances in Neural Information Processing Systems*, 2019.

Manuel Kaspar, Juan David Muñoz Osorio, and Jürgen Bock. Sim2real transfer for reinforcement learning without dynamics randomization. *CoRR*, abs/2002.11635, 2020.

Andrey Kurenkov, Ajay Mandlekar, Roberto Martin-Martin, Silvio Savarese, and Animesh Garg. Ac-teach: A bayesian actor-critic method for policy learning with an ensemble of suboptimal teachers. *CoRL*, 2019.

Guohao Li, Matthias Müller, Vincent Casser, Neil Smith, Dominik L Michels, and Bernard Ghanem. Oil: Observational imitation learning. In *Robotics: Science and Systems*, 2019a.

Mao Li, Yi Wei, and Daniel Kudenko. Two-level q-learning: learning from conflict demonstrations. *The Knowledge Engineering Review*, 2019b.

Manuel Lopes, Tobias Lang, Marc Toussaint, and Pierre-Yves Oudeyer. Exploration in model-based reinforcement learning by empirically estimating learning progress. *NIPS*, 2012.

Muhammad A. Masood and Finale Doshi-Velez. Diversity-inducing policy gradient: Using maximum mean discrepancy to find a set of diverse policies. *CoRR*, abs/1906.00088, 2019.

Alberto Maria Metelli, Amarildo Likmeta, and Marcello Restelli. Propagating uncertainty in reinforcement learning via wasserstein barycenters. *Advances in Neural Information Processing Systems*, 2019.

Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. *arXiv preprint arXiv:1509.08731*, 2015.

Douglas Morrison, Adam W Tow, Matt Mctaggart, R Smith, Norton Kelly-Boxall, Sean Wade-Mccue, Jordan Erskine, Riccardo Grinover, Alec Gurman, T Hunn, et al. Cartman: The low-cost cartesian manipulator that won the amazon robotics challenge. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018.

Junhyuk Oh, Yijie Guo, Satinder Singh, and Honglak Lee. Self-imitation learning. In *International Conference on Machine Learning*, pp. 3878–3887. PMLR, 2018.

OpenAI, Matthias Plappert, Raul Sampedro, Tao Xu, Ilge Akkaya, Vineet Kosaraju, Peter Welinder, Ruben D'Sa, Arthur Petron, Henrique Ponde de Oliveira Pinto, Alex Paino, Hyeonwoo Noh, Lilian Weng, Qiming Yuan, Casey Chu, and Wojciech Zaremba. Asymmetric self-play for automatic goal discovery in robotic manipulation. *CoRR*, 2021. URL `https://arxiv.org/abs/2101.04882`.

Ian Osband, Benjamin Van Roy, and Zheng Wen. Generalization and exploration via randomized value functions. In *International Conference on Machine Learning*, 2016.

Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pp. 2778–2787. PMLR, 2017.

Silviu Pitis, Harris Chan, Stephen Zhao, Bradly Stadie, and Jimmy Ba. Maximum entropy gain exploration for long horizon multi-goal reinforcement learning. In *International Conference on Machine Learning*, pp. 7750–7761. PMLR, 2020.

Sebastien Racaniere, Andrew Lampinen, Adam Santoro, David Reichert, Vlad Firoiu, and Timothy Lillicrap. Automated curriculum generation through setter-solver interactions. In *International conference on learning representations*, 2019.

Roberta Raileanu and Tim Rocktäschel. Ride: Rewarding impact-driven exploration for procedurally-generated environments. *arXiv preprint arXiv:2002.12292*, 2020.

A. I. Redefined, Sai Krishna Gottipati, Sagar Kurandwad, Clod'eric Mars, Gregory Szriftgiser, and François Chabot. Cogment: Open source framework for distributed multi-actor training, deployment & operations. *CoRR*, abs/2106.11345, 2021. URL `https://arxiv.org/abs/2106.11345`.

Tom Schaul, Daniel Horgan, Karol Gregor, and David Silver. Universal value function approximators. In *International conference on machine learning*, pp. 1312–1320. PMLR, 2015.

Jürgen Schmidhuber. A possibility for implementing curiosity and boredom in model-building neural controllers. In *Proc. of the international conference on simulation of adaptive behavior: From animals to animats*, pp. 222–227, 1991.

Yun-Hsuan Su, Kevin Huang, and Blake Hannaford. Multicamera 3d viewpoint adjustment for robotic surgery via deep reinforcement learning. *Journal of Medical Robotics Research*, 6(01n02): 2140003, 2021.

Sainbayar Sukhbaatar, Zeming Lin, Ilya Kostrikov, Gabriel Synnaeve, Arthur Szlam, and Rob Fergus. Intrinsic motivation and automatic curricula via asymmetric self-play. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*, 2018.

Daochen Zha, Wenye Ma, Lei Yuan, Xia Hu, and Ji Liu. Rank the episodes: A simple approach for exploration in procedurally-generated environments. *ICLR*, 2021.

Songyuan Zhang, Zhangjie Cao, Dorsa Sadigh, and Yanan Sui. Confidence-aware imitation learning from demonstrations with varying optimality. In *NeurIPS*, 2021.