When Is Generalizable Reinforcement Learning Tractable?

Dhruv Malik [1] **Yuanzhi Li** [1] **Pradeep Ravikumar** [1]

Abstract

Agents trained by reinforcement learning (RL) often fail to generalize beyond the environment they were trained in, even when presented with new scenarios that seem similar to the training environment. We study the query complexity required to train RL agents that generalize to multiple environments. Intuitively, tractable generalization is only possible when the environments are similar or close in some sense. To capture this, we introduce *Weak Proximity*, a natural structural condition that requires the environments to have highly similar transition and reward functions and share a policy providing optimal value. Despite such shared structure, we prove that tractable generalization is impossible in the worst case. This holds even when each individual environment can be efficiently solved to obtain an optimal linear policy, and when the agent possesses a generative model. Our lower bound applies to the more complex task of representation learning for efficient generalization to multiple environments. On the positive side, we introduce *Strong Proximity*, a strengthened condition which we prove is sufficient for efficient generalization.

1. Introduction

Reinforcement learning (RL) is the dominant paradigm for sequential decision making in machine learning. But many issues prevent RL from being regularly used in the real world. For example, one typically trains and tests RL agents in the same environment. In such cases, an agent can memorize behavior that achieves high reward, without acquiring the true behavior that the system designer desires. This has raised concerns about RL agents overfitting to a single environment, instead of learning meaningful skills (Farebrother et al., 2018). And although RL agents can solve difficult tasks, they struggle to transfer the skills they learned in one task to perform well in a different but similar task (Rakelly et al., 2019; Yu et al., 2019). Yet, in the real world, it is reasonable to expect that RL agents will see scenarios that are different from the specific scenarios they trained for.

Hence, a desirable property of RL agents is that of *generalization*, broadly defined as the ability to discern the correct notion of behavior and perform well in semantically similar environments. We focus on two popular generalization settings. The *Average Performance* setting assumes there is an underlying distribution over the environments that an agent might encounter. The agent's goal is to perform well on average across this distribution (Packer et al., 2018; Nichol et al., 2018; Cobbe et al., 2019). The *Meta Reinforcement Learning* setting is closely related (Finn et al., 2017; Clavera et al., 2019; Rakelly et al., 2019). Here an agent first learns from training environments sampled from a distribution. Then at test time the agent must leverage this experience to adapt to a new environment sampled from the same distribution, via only a few queries in the new environment.

Of course, in full generality, both notions of generalization are impossible to achieve efficiently. Hence, key to both lines of inquiry is the premise that the environments are structurally similar. For example, a robot may face the differing tasks of screwing a bottle cap and turning a doorknob, but both tasks involve turning the wrist (Rakelly et al., 2019). The hope is that if the environments are sufficiently similar, then RL can exploit this structure to efficiently discover policies that generalize. Yet, it remains unclear what it means for different environments to be close or similar. Motivated by this, we ask the following question:

What are the structural conditions on the environments that permit efficient generalization?

This question underlies the analysis of our paper. We focus on environments that share state-action spaces, since even this basic case is not well understood in the literature. Indeed, even in this simplified setting, efficient generalization is highly non-trivial. We make the following contributions.

Our Contributions. We introduce *Weak Proximity*, a natural structural condition that is motivated by classical RL results, and requires the environments to have highly similar transition and reward functions and share optimal trajectories. We prove a statistical lower bound demonstrating that

[1]Machine Learning Department, Carnegie Mellon University. Correspondence to: Dhruv Malik <dhruvm@andrew.cmu.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

tractable generalization is impossible, despite this shared structure. This lower bound holds even when each individual environment can be efficiently solved to obtain an optimal linear policy, and when the agent possesses a generative model. Consequentially, we show that a classical metric for measuring the relative closeness of MDPs is not the right metric for modern RL generalization settings. Our lower bound implies that learning a state representation for the purpose of efficiently generalizing to multiple environments, is worst case sample inefficient — even when such a representation exists, the environments are ostensibly similar, and any single environment can be efficiently solved.

To provide a sufficient condition for efficient generalization, we introduce *Strong Proximity*. This structural condition strengthens Weak Proximity by additionally constraining the environments to share an optimal policy. We provide an algorithm which exploits Strong Proximity to provably and efficiently generalize, when the environments share deterministic transitions.

In this extended abstract, we will only provide results for the Average Performance Setting, due to space constraints. However, we stress that all our results hold for the Meta RL setting, and we defer these to the main paper.

2. Problem Formulation

Notation & Preliminaries. We always use M to denote a Markov decision process (MDP). Recall that an undiscounted finite horizon MDP is specified by a set of states $\mathcal{S}$, a set of actions $\mathcal{A}$, a transition function $\mathcal{T}$ which maps from state-action pairs to distributions over states, a reward function R which maps state-action pairs to nonnegative real numbers, and a finite planning horizon H. We assume that the state-action pairs are featurized, so that $\mathcal{S}\times\mathcal{A}\subset\mathbb{R}^d$, and that $\|(s,a)\|_2 = 1$ for all $(s,a)\in\mathcal{S}\times\mathcal{A}$. Any MDP we consider is undiscounted and has a finite action space, but could have an uncountable state space. If we need to refer to the transition or reward function of a specific MDP M, then we shall denote this via $\mathcal{T}_M$ or R_M. We will denote a distribution over MDPs as $\mathcal{D}$. We also assume that $\mathcal{S}$ can be partitioned into H different levels. This means that for each $s\in\mathcal{S}$ there exists a unique $h\in\{0,1\dots H-1\}$ such that it takes h timesteps to arrive at s from s_0. We say that such a state s lies on level h, and denote $\mathcal{S}_h$ to be the set of states on level h. For any MDP, we assume a single initial state s_0, which strengthens our lower bounds.

A policy maps each state to a corresponding distribution over actions, and shall typically be denoted by π. The total expected reward accumulated by policy π when initialized at state s in MDP M is given by $\mathbb{E}\left[\sum_{h=\text{level}(s)}^{H-1} R_M(s_h,a_h)\mid\pi\right]$ and will be denoted by $V_M^s(\pi)$. Here the expectation is over the trajectory $\{(s_h,a_h)\}_{h=\text{level}(s)}^{H-1}$ given that the first state in the trajectory is s. So $V_M^s(\pi)$ is the value of the policy π in MDP M with respect to (w.r.t) initial state s. Analogously, if a policy is parameterized by $\overline{\theta}=\{\theta_h\}_{h=0}^{H-1}$, then we denote it as $\pi(\overline{\theta})$, and the notation $V_M^s(\pi)$ is then replaced by $V_M^s(\overline{\theta})$. We assume that the cumulative reward collected by any trajectory from any initial state s in any MDP M is always bounded by 1. $\mathbb{TV}(P,Q)$ denotes the total variation distance between probability distributions P and Q.

2.1. Problem Setting

Average Performance Setting. There is a fixed distribution $\mathcal{D}$ over a family of MDPs. One can sample MDPs from $\mathcal{D}$. The algorithm can query states in the sampled MDPs, to learn some common structure. The goal is to solve

$$\max_{\pi}\mathbb{E}_{M\sim\mathcal{D}}\left[V_M^{s_0}(\pi)\right]. \tag{1}$$

The Meta RL setting is deferred to the main paper. "Sampling an MDP" means drawing an MDP i.i.d from $\mathcal{D}$, so that the agent can then interact with it by performing trajectories in it. Note that in Eq. (1), we assume a single initial state s_0. This strong assumption only strengthen our lower bounds. Furthermore, it is necessary to understand this simpler setting, before looking at more complex scenarios.

To solve Eqs. (1), we need to define an appropriate query model for the algorithm. We consider two query models, the first of which is strictly stronger than the second.

Strong Query Model (SQM). Sampling an MDP from $\mathcal{D}$ incurs no cost. The agent has a generative model of any sampled MDP M. To interact with M, the agent inputs a state-action pair (s,a) of M into the model, and receives $R_M(s,a)$ and a state sampled from $\mathcal{T}_M(s,a)$. This incurs a query cost of one.

Weak Query Model (WQM). Sampling an MDP from $\mathcal{D}$ incurs a query cost of $q_{\mathcal{D}}\geq 1$. Within a sampled MDP M, the agent operates in the standard episodic RL setup, by starting from s_0, taking an action and observing the next state and reward, and repeating. Each action taken during an episode incurs a query cost of one.

We shall present our lower bounds under SQM, which makes these results stronger, but shall present our upper bound under the natural and standard WQM.

Without any conditions on $\mathcal{D}$, generalization can be intractable, even under SQM. This will occur if the MDPs supporting $\mathcal{D}$ do not share structure. This will also occur if any individual MDP cannot be solved efficiently. Nevertheless, in practice one often deals with MDPs which share meaningful structure (Cobbe et al., 2019; Rakelly et al., 2019). For instance, the transition distributions of the MDPs may be close in a suitable metric. Similarly, the reward func-

tions of the MDPs might be close in an appropriate norm, or each MDP may share a set of optimal trajectories. And in practice, individual MDPs can usually be optimized efficiently (Packer et al., 2018; Yu et al., 2019). In such cases, it is reasonable to expect tractable generalization. We are interested in formalizing conditions that permit efficient generalization. We will particularly focus on conditions which capture shared structure of the MDPs and the tractability of individual MDPs. We now formally state the problem we consider throughout our paper.

Which conditions on $\mathcal{D}$ allow us to solve the Average Performance setting efficiently?

As mentioned above, there are two types of requirements. The first requirement should ensure that the MDPs are meaningfully similar. We formalize such conditions in Section 2.2. The second requirement should ensure that any individual MDP is efficiently solvable, else there is no hope to efficiently find policies that generalize for many MDPs. We formalize such properties in Section 2.3.

2.2. Strong & Weak Proximity

We now identify conditions that capture when the MDPs supporting $\mathcal{D}$ share meaningful structure. Since MDPs are defined in terms of rewards and transitions, it is very natural to impose conditions directly on the rewards and transitions. To this end, we state the following condition.

Condition 1 (Similar Rewards & Transitions) *The distribution $\mathcal{D}$ satisfies this condition with parameters $\xi_{\mathrm{r}}, \xi_{\mathrm{tr}} \geq 0$ when:*

(a) Each MDP supporting $\mathcal{D}$ shares the same state-action space $\mathcal{S} \times \mathcal{A}$.

(b) For all M_i, M_j supporting $\mathcal{D}$ and all $(s,a) \in \mathcal{S} \times \mathcal{A}$ we have $|R_{M_i}(s,a) - R_{M_j}(s,a)| \leq \xi_{\mathrm{r}}$.

(c) For all M_i, M_j supporting $\mathcal{D}$ and all $(s,a) \in \mathcal{S} \times \mathcal{A}$ we have $\mathbb{TV}(\mathcal{T}_{M_i}(s,a), \mathcal{T}_{M_j}(s,a)) \leq \xi_{\mathrm{tr}}$.

The parameters $\xi_{\mathrm{r}}, \xi_{\mathrm{tr}}$ quantify the similarity of different MDPs. Conditions of this form are canonical and have yielded fruitful research in classical literature, in the guise of the Simulation Lemma (Kearns & Koller, 1999; Kearns & Singh, 2002; Brafman & Tennenholtz, 2003; Kakade et al., 2003; Abbeel & Ng, 2005). To concretize this condition with an example, consider a suite of simulated robotic goal reaching tasks (Yu et al., 2019), where the physics simulator is the same in each task, so the transitions are fixed and $\xi_{\mathrm{tr}} = 0$, but the goal location changes from task to task, implying that $\xi_{\mathrm{r}} > 0$. We now establish our Weak Proximity condition, which strictly strengthens Condition 1.

Condition 2 (Weak Proximity) *The distribution $\mathcal{D}$ satisfies Weak Proximity with parameters $\xi_{\mathrm{r}}, \xi_{\mathrm{tr}}, \alpha \geq 0$ when:*

(a) $\mathcal{D}$ satisfies Condition 1 with parameters $\xi_{\mathrm{r}}, \xi_{\mathrm{tr}} \geq 0$.

(b) There exists a deterministic policy $\pi^\star$ which for any MDP M satisfies $V_M^{s_0}(\pi^\star) \geq \max_{\pi'} V_M^{s_0}(\pi') - \alpha$.

Weak Proximity strengthens Condition 1 by additionally requiring (via part (b)) that there exists some policy $\pi^\star$ which provides α-suboptimal value for each MDP supporting $\mathcal{D}$. Intuitively, this condition implicitly constrains the MDPs to be similar, since there is a single policy which provides (nearly) optimal value, irrespective of the MDP it is deployed in. Furthermore, recall from Eq. (1) that the objective is defined in terms of value w.r.t the initial state s_0. So it is natural to assume, as we do in part (b), that there is one policy which provides good value w.r.t s_0 for all MDPs.

We now present Strong Proximity, a condition which strictly strengthens Weak Proximity. We will later show that unlike its Weak counterpart, Strong Proximity indeed permits efficient generalization.

Condition 3 (Strong Proximity) *The distribution $\mathcal{D}$ satisfies Strong Proximity with parameters $\xi_{\mathrm{r}}, \xi_{\mathrm{tr}}, \alpha \geq 0$ when:*

(a) $\mathcal{D}$ satisfies Condition 1 with parameters $\xi_{\mathrm{r}}, \xi_{\mathrm{tr}} \geq 0$.

(b) There exists a deterministic policy $\pi^\star$ which is a near optimal policy for each MDP. Concretely, the policy $\pi^\star$ satisfies $V_M^s(\pi^\star) \geq \max_{\pi'} V_M^s(\pi') - \alpha$ for each state s and each MDP M.

2.3. Tractability of Individual Optimization

As discussed previously, in order to efficiently solve Eq. (1), we require that each individual MDP supporting $\mathcal{D}$ can be efficiently solved. We now state two such properties, the first of which is strictly stronger than the second. Since these properties require a notion of query cost, we state both of them with reference to a generic query model QM, and when we later present our results we will instantiate QM to be either SQM or WQM. We use $\pi_M^\star$ to denote an arbitrary deterministic optimal policy of MDP M. We also use the standard notion of a linear policy (Du et al., 2020).

Property 1 (Strong Individual Optimization (SIO)) *Let the query model be QM. The distribution $\mathcal{D}$ satisfies SIO with parameters $k > 0$ and $0 \leq \beta < 1/4$ when:*

(a) Any MDP M supporting $\mathcal{D}$ admits an optimal linear policy.

(b) There exists a fixed and known algorithm, such that given any MDP M and any state s, this algorithm uses

at most $\mathcal{O}(|\mathcal{A}|H^k)$ query cost (under QM) on M to identify (almost surely) a linear policy $\pi(\overline{\theta})$ parameterized by $\overline{\theta} = \{\theta_h\}_{h=0}^{H-1}$ which satisfies $\max_{\pi'} V_M^s(\pi') \geq V_M^s(\overline{\theta}) \geq \max_{\pi'} V_M^s(\pi') - \beta$. This algorithm then outputs $\pi(\overline{\theta})$ as well as $V_M^s(\overline{\theta})$.

Part (a) requires that for any MDP supporting $\mathcal{D}$, there exists an optimal linear policy. Part (b) requires that the user has knowledge of an algorithm, which can efficiently find a linear policy providing β-suboptimal value from any input state s in any MDP M. SIO is a fairly strong property, since it says that a linear policy is sufficient to optimize any individual MDP, whereas in practice one typically requires nonlinear neural network policies. SIO also heavily constrains each individual MDP supporting $\mathcal{D}$ to be efficiently solvable from any initial state. We stress that we will prove our *lower bounds* under SIO, which makes our result stronger. Meanwhile, we prove our *upper bounds* under the following property, which is significantly weaker than SIO, and makes no unrealistic linearity assumptions.

Property 2 (Weak Individual Optimization (WIO)) *Let the query model be QM. The distribution $\mathcal{D}$ satisfies WIO with parameter $0 \leq \beta < 1/4$ when the following holds. There exists an oracle $\widehat{V}$, which takes as input a state s and MDP M, and outputs $\widehat{V}_M^s$ satisfying $\max_{\pi'} V_M^s(\pi') \geq \widehat{V}_M^s \geq \max_{\pi'} V_M^s(\pi') - \beta$, via query cost (under QM) on M that is polynomial in $|\mathcal{A}|, H$.*

3. Main Results

In this extended abstract, we only present results for when the MDPs supporting $\mathcal{D}$ share a deterministic transition function, due to space constraints. Our lower bounds extend to when the MDPs share a reward function but have varying transitions, and we defer this to the main paper.

The classical Simulation Lemma (Kearns & Koller, 1999; Kearns & Singh, 2002; Brafman & Tennenholtz, 2003; Kakade et al., 2003; Abbeel & Ng, 2005) shows that if the MDPs supporting $\mathcal{D}$ share deterministic transitions (so that $\mathcal{D}$ at the minimum satisfies Condition 1 with $\xi_{\text{tr}} = 0$), then generalization is only non-trivial in the regime where ξ_{r} is $\Omega(\frac{1}{H})$. The following result is a *lower bound* which shows that when $\xi_{\text{r}} = \Theta(\frac{1}{H})$, then Weak Proximity is not sufficient to efficiently generalize in the Average Performance Setting.

Theorem 1 *Let the query model be SQM. For any $k \geq 3$, there exists $\mathcal{D}$ satisfying Weak Proximity with $\xi_{\text{r}} = \Theta(\frac{1}{H})$, $\xi_{\text{tr}} = 0$ & $\alpha = 0$ and SIO with $\beta = 0$ & k, such that the MDPs supporting $\mathcal{D}$ are deterministic and the following holds. Any (possibly randomized) algorithm requires $\Omega(|\mathcal{A}|^H)$ total query cost to find (with probability at least $1/2$) a policy π satisfying*

$$\mathbb{E}_{M\sim\mathcal{D}}\left[V_M^{s_0}(\pi)\right] \geq \max_{\text{linear policy } \pi'} \mathbb{E}_{M\sim\mathcal{D}}\left[V_M^{s_0}(\pi')\right] - 1/4.$$

The theorem demonstrates that one can require an exponential query cost to find a policy that is nearly as good as the best *linear* policy (which is of course easier than finding the best generic policy). This holds even though individual MDPs are easily optimized to obtain a linear policy providing optimal value, as defined in SIO, and even though the MDPs are similar as defined in Weak Proximity. This suggests that the classical (and natural) way of measuring variation in MDPs using Condition 1 is not the right metric for the modern Average Performance setting. Indeed, classical results show that generalization is trivial when ξ_{r} is $o(\frac{1}{H})$. But when ξ_{r} is $\Theta(\frac{1}{H})$ then these settings become exponentially hard, even under the additional Weak Proximity condition as well as SIO & SQM.

Note that Theorem 1 holds even though each MDP supporting $\mathcal{D}$ shares a state-action space. So these lower bounds immediately apply to more complex settings where the MDPs are defined on disjoint state-action spaces, and where learning an appropriate representation is necessary. Indeed, it is popular in practice to learn a feature mapping which maps similar states to the same vector. Our results show that if such a mapping enables efficient solution of Eq. (1), then learning the mapping itself is worst case inefficient.

We now show that Strong Proximity permits efficient generalization when the MDPs supporting $\mathcal{D}$ share deterministic transitions. Notably, to prove our upper bound we only require the weaker WQM and weaker WIO.

Theorem 2 *Let the query model be WQM. Consider any $\mathcal{D}$ satisfying WIO with $\beta \geq 0$ and Strong Proximity with $\xi_{\text{tr}} = 0$ and any $\alpha, \xi_{\text{r}} \geq 0$, such that the MDPs supporting $\mathcal{D}$ are deterministic. Fix $\epsilon, \delta > 0$. There exists an algorithm whose total query complexity is polynomial in $q_{\mathcal{D}}, |\mathcal{A}|, H, \frac{1}{\epsilon}, \log(\frac{1}{\delta})$, such that the following holds. With probability at least $1-\delta$, the algorithm outputs policy π satisfying*

$$\mathbb{E}_{M\sim\mathcal{D}}\left[V_M^{s_0}(\pi)\right] \geq \max_{\pi'} \mathbb{E}_{M\sim\mathcal{D}}\left[V_M^{s_0}(\pi')\right] - \epsilon - 3\,\alpha\,H - 3\,\beta\,H.$$

Note Theorem 2 holds under WIO. By contrast, Weak Proximity was insufficient for efficient generalization even when paired with SIO. This suggests that a condition that is both necessary and sufficient for efficient generalization lies somewhere between Weak and Strong Proximity — assuming, of course, that we do not assume an individual optimization property that is even stronger than SIO. Indeed, SIO is already strong, since SIO says that a linear policy is sufficient to optimize any individual MDP, but in practice one typically employs nonlinear neural network policies.

4. Discussion

In this paper, we studied the design of RL agents that generalize. We proved that efficient generalization is worst case impossible, even under structural conditions like Weak Proximity and strong assumptions on the query model and tractability of individual MDPs. This result extends to the task of learning representations for the purpose of efficient generalization. On the positive side, we provided Strong Proximity, which permits efficient generalization, even under mild assumptions on the query model and individual tractability. Our analysis highlights that classical metrics for measuring similarity of MDPs are inappropriate for modern RL. It also suggests that a condition which is both necessary and sufficient for efficient generalization lies between Weak & Strong Proximity – unless we make (arguably unreasonable) assumptions on the tractability of individual MDPs.

We emphasize again that the Weak Proximity condition is extremely natural. Since MDPs are defined in terms of their rewards and transitions, it is natural to constrain the rewards and transitions when measuring similarity of MDPs. Indeed, these hard constraints on the transitions and rewards at every state are unlikely to always be satisfied in practice, which only makes our lower bound stronger. And the existence of an optimal policy with respect to s_0, is also extremely natural and only makes the problem easier. Strong Proximity is also reasonable, but in our setting it requires that all MDPs share a state-action space. We believe that in practice, after an appropriate representation is learned which maps similar states of MDPs to the same state space (see below), then Strong Proximity would apply.

The primary limitation of our work is that our upper bound has limited applicability. It holds only when the MDPs share a state-action space, which is very restrictive in practice. Our rationale for working in this restricted setting was due to our lower bounds, which show that even this toy setting can be worst case inefficient, and because it is necessary to understand the toy setting before looking at more complex scenarios. Nevertheless, our upper bound is several steps removed from the practice of RL. It is best interpreted as a preliminary sufficient condition for when efficient generalization is possible, albeit in a toy setting, and is far from conclusive on this matter.

Note that our upper bound might apply if we are a priori given a feature mapping which maps similar states of different MDPs to the same state space. For example, in self driving, learning to drive in different countries might be difficult because the images of traffic signs are different. But if a known feature map extracts the underlying meaning of these signs, then our upper bound could conceivably apply. Of course, such a known feature map is rarely available a priori, and is usually learned from data. The key direction for future work, is how to learn such a feature mapping efficiently, while ensuring that it is still useful for generalization.

Acknowledgements

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE1745016. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

References

Abbeel, P. and Ng, A. Y. Exploration and apprenticeship learning in reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2005.

Brafman, R. I. and Tennenholtz, M. R-max - a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research*, 3: 213–231, 2003.

Clavera, I., Nagabandi, A., Liu, S., Fearing, R. S., Abbeel, P., Levine, S., and Finn, C. Learning to adapt in dynamic, real-world environments through meta-reinforcement learning. In *International Conference on Learning Representations*, 2019.

Cobbe, K., Klimov, O., Hesse, C., Kim, T., and Schulman, J. Quantifying generalization in reinforcement learning. In *Proceedings of the International Conference on Machine Learning*, 2019.

Du, S. S., Luo, Y., Wang, R., and Zhang, H. Provably efficient q-learning with function approximation via distribution shift error checking oracle. In *Advances in Neural Information Processing Systems*, 2019.

Du, S. S., Kakade, S. M., Wang, R., and Yang, L. F. Is a good representation sufficient for sample efficient reinforcement learning? In *International Conference on Learning Representations*, 2020.

Farebrother, J., Machado, M. C., and Bowling, M. Generalization and regularization in DQN. *arXiv preprint arxiv:1810.00123*, 2018.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *Proceedings of the International Conference on Machine Learning*, 2017.

Gu, S., Holly, E., Lillicrap, T., and Levine, S. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *Proceedings of the IEEE*

International Conference on Robotics and Automation, 2017.

Haarnoja, T., Pong, V., Zhou, A., Dalal, M., Abbeel, P., and Levine, S. Composable deep reinforcement learning for robotic manipulation. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2018.

Kakade, S., Kearns, M., and Langford, J. Exploration in metric state spaces. In *Proceedings of the International Conference on Machine Learning*, 2003.

Kearns, M. and Koller, D. Efficient reinforcement learning in factored mdps. In *Proceedings of the International Joint Conference on Artificial Intelligence*, 1999.

Kearns, M. and Singh, S. Near-optimal reinforcement learning in polynomial time. *Machine Learning*, 49:209–232, 2002.

Lattimore, T., Szepesvari, C., and Weisz, G. Learning with good feature representations in bandits and in RL with a generative model. In *Proceedings of the International Conference on Machine Learning*, 2020.

Mnih, V. et al. Human-level control through deep reinforcement learning. *Nature*, 518:529–533, 2015.

Nichol, A., Pfau, V., Hesse, C., Klimov, O., and Schulman, J. Gotta learn fast: A new benchmark for generalization in RL. *arXiv preprint arxiv:1804.03720*, 2018.

Packer, C., Gao, K., Kos, J., Krähenbühl, P., Koltun, V., and Song, D. Assessing generalization in deep reinforcement learning. *arXiv preprint arxiv:1810.12282*, 2018.

Rakelly, K., Zhou, A., Finn, C., Levine, S., and Quillen, D. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In *Proceedings of the International Conference on Machine Learning*, 2019.

Roy, B. V. and Dong, S. Comments on the du-kakade-wang-yang lower bounds. *arXiv preprint arxiv:1911.07910*, 2019.

Silver, D. et al. Mastering the game of go without human knowledge. *Nature*, 550:354–359, 2017.

Weisz, G., Amortila, P., Janzer, B., Abbasi-Yadkori, Y., Jiang, N., and Szepesvàri, C. On query-efficient planning in mdps under linear realizability of the optimal state-value function. *arXiv preprint arxiv:2102.02049*, 2021.

Yu, T., Quillen, D., He, Z., Julian, R., Hausman, K., Finn, C., and Levine, S. Meta-world: A benchmark and evaluation for multi-task and meta reinforcement learning. In *Proceedings of the Conference on Robot Learning*, 2019.