UNIVERSITY OF CALIFORNIA SAN DIEGO

Regret Analysis for Discounted Reinforcement Learning

A dissertation submitted in partial satisfaction of the
requirements for the degree Doctor of Philosophy

in

Computer Science

by

Shuang Liu

Committee in charge:

Professor Hao Su, Chair
Professor Sanjoy Dasgupta
Professor Russell Impagliazzo
Professor Ramamohan Paturi
Professor Zhuowen Tu

2022

The Dissertation of Shuang Liu is approved, and it is acceptable in quality and form for publication on microfilm and electronically.

University of California San Diego

2022

DEDICATION

To my grandfather Cixuan Yu and grandmother Kailan Jiang.

TABLE OF CONTENTS

LIST OF ALGORITHMS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

I would like to thank my PhD advisor Hao Su, for his guidance and support during the last four years of my PhD study. He offered me the chance to explore the areas of machine learning that I am most interested in without any constraints, and is willing to have a discussion whenever I feel necessary.

I would also like to thank my former PhD advisor Kamalika Chaudhuri, who opened up many exciting directions to me and taught me a lot how to research and write papers.

I am fortunate to have my labmates accompanied me through my PhD journey, special thanks to Zhiwei, Tongzhou, Fangchen, and Quan.

Finally, I could not have come this far without my wife Xiaojing, who have been giving me endless support; and my parents, who have been educating me and encouraging me.

Chapter 2, 3, 4 of this dissertation are primarily based on material in the manuscript “Regret Bounds for Discounted MDPs" (Shuang Liu and Hao Su), of which the dissertation author was the primary researcher and author.

Chapter 5 of this dissertation is primarily based on material in the manuscript “Provably Efficient Kernelized Q-Learning" (Shuang Liu and Hao Su), of which the dissertation author was the primary researcher and author.

VITA

2016	Bachelor of Science, Shanghai Jiao Tong University
2022	Doctor of Philosophy, University of California San Diego

PUBLICATIONS

Zhiwei Jia, Xuanlin Li, Zhan Ling, Shuang Liu, Yiran Wu and Hao Su. “Improving Policy Optimization with Generalist-Specialist Learning”. ICML 2022

Shuang Liu, Renshen Wang, Michalis Raptis and Yasuhisa Fujii. “Unified Line and Paragraph Detection by Graph Convolutional Networks”. DAS 2022 (oral presentation)

Jiachen Li, Quan Vuong, Shuang Liu, Minghua Liu, Kamil Ciosek, Henrik Christensen and Hao Su “Multi-task Batch Reinforcement Learning with Metric Learning”. NeurIPS 2020

Shuang Liu, Olivier Bousquet and Kamalika Chaudhuri. “Approximation and Convergence Properties of Generative Adversarial Learning” NeurIPS 2017 (spotlight presentation)

MANUSCRIPTS

Shuang Liu and Hao Su. “Provably Efficient Kernelized Q-Learning”.

Shuang Liu and Hao Su. “Regret Bounds for Discounted MDPs”.

Shuang Liu and Kamalika Chaudhuri “The Inductive Bias of Restricted f-GANs”.

Shuang Liu, Cheng Chen and Zhihua Zhang “Regret vs. Communication: Distributed Stochastic Multi-Armed Bandits and Beyond”.

ABSTRACT OF THE DISSERTATION

Regret Analysis for Discounted Reinforcement Learning

by

Shuang Liu

Doctor of Philosophy in Computer Science

University of California San Diego, 2022

Professor Hao Su, Chair

Reward discounting has become an indispensable ingredient in designing practical reinforcement learning (RL) algorithms. However, standard notions of regret in theoretical RL are not capable of capturing the essence of reward discounting and, as such, fail to serve as optimality criteria for analyzing practical RL algorithms. Three questions naturally arise:

(Q1) Can we have a different notion of regret that encapsulates the idea of reward discounting?

(Q2) Can we design RL algorithms, potentially with reward discounting, that can be analyzed under this different notion?

(Q3) Can we make these algorithms tackle non-linear problems effectively and efficiently so

that they can compete with widely-used RL algorithms on standard benchmarks?

We address these three questions in this dissertation. To answer (Q1), we introduce a new notion of regret, named γ-regret, to capture the concept of regret in discounted RL. The parameter γ in γ-regret serves as an alternative to the horizon or the diameter in traditional RL analysis. The definition of γ-regret not only captures the idea of reward discounting in prevalent RL algorithms, but also enables rigorous theoretical analysis of these practices. It is also deeply connected with other regret-related notions in the existing RL literature.

Under the γ-regret framework, both algorithm design and theoretical analysis become more challenging and require innovation. Nonetheless, we are able to make significant progress toward answering (Q2). In particular, we first derive a lower bound of $\Omega\left(\sqrt{T/(1-\gamma)}\right)$, where T is the total number of interactions, on the γ-regret under the tabular setting. We then introduce two algorithmic instantiations of the Q-learning paradigm, tabular double Q-learning and kernelized Q-learning, both of which are amenable to theoretical analysis under the γ-regret framework. In particular, we obtain an upper bound of $\tilde{O}\left(\sqrt{T/(1-\gamma)^3}\right)$ and $\tilde{O}\left(\sqrt{T/(1-\gamma)^5}\right)$ respectively on the γ-regret for these two algorithms under various settings; notably, tabular double Q-learning is the first RL algorithm that has near-optimal γ-regret, and since it was introduced, many more RL algorithms has been designed and analyzed under the γ-regret framework.

We also take a big step toward answering (Q3). Specifically, we complement our analysis of kernelized Q-learning with experiments on classic control tasks; remarkably, kernelized Q-learning is the first RL algorithm that can nearly solve the MountainCar environment in as few as one thousand steps.

Chapter 1

Introduction

Reinforcement learning (RL) is concerned with how an algorithm should interact with a (partially) unknown environment to maximize the cumulative reward. An environment is typically modeled as a Markov decision process (MDP), which is specified by a state space, an action space, a reward function, and a transition function. During the interaction, the algorithm is associated with a state in the state space at any time. Whenever, the algorithm takes an action in the action space, it receives a reward according to the reward function, and is transitioned (i.e., re-associated) to a new state according to the transition function. Both the reward function and the transition function take the current state associated with the algorithm and the action taken as inputs. The interaction is terminated typically after a certain number of actions have been taken.

RL is a very general model that captures the decision-making process of intellectual entities and have been one of the most studied problems in machine learning. RL should not be confused with planning, where the reward function and the transition function are both given to the algorithm before the interaction. In RL, these two functions are only partially known to the algorithm; most of the time, the algorithm designer does not have any prior on these two functions except for their domain and range.

In this dissertation, we focus on an important aspect of reinforcement learning — reward discounting. Reward discounting has become an indispensable ingredient in designing practical RL algorithms. However, standard notions of regret in theoretical RL cannot capture the essence of reward discounting and, as such, fail to serve as optimality criteria for analyzing practical RL algorithms. In view of this, three questions naturally arise:

(Q1) Can we have a different notion of regret that encapsulates the idea of reward discounting?

(Q2) Can we design RL algorithms, potentially with reward discounting, that can be analyzed under this different notion?

(Q3) Can we make these algorithms tackle non-linear problems effectively and efficiently so that they can compete with widely-used RL algorithms on standard benchmarks?

We will address these three questions in this dissertation. In the rest of the introduction, we first highlight how reward discounting is treated in practice (Section 1.1). We then show a concrete example to give the reader a better idea why reward discounting is necessary (Section 1.2). We conclude with a summary of our contributions in terms of addressing the three questions (Section 1.3). The final section also serves as an outline of the remaining part of the dissertation.

1.1 Reward Discounting in Practice

1.1.1 Q-Learning

Q-learning, which dates back to as early as Watkins and Dayan (1992), has undergone tremendous development in the past decades and remains one of the most sample-efficient RL frameworks in practice.

Early studies of Q-learning focused on the tabular setting, where both the state space and the action space are finite sets. In a common implementation, a two-dimensional array Q is initialized arbitrarily (e.g., by zeros) and updated throughout the learning process such that whenever the algorithm takes an action a in a state s and then receives a reward r and gets transitioned to a new state s', we set

$$Q[s][a] = (1-\alpha)Q[s][a] + \alpha\left(r + \gamma \max_{a'} Q[s'][a']\right), \tag{1.1}$$

where $\gamma \in [0,1)$ is a discounting factor. Actions that maximizes $Q[s][a]$ are taken with priority — for example, the algorithm can use ε-*greedy* to take actions, i.e., in state s, it takes a to maximize $Q[s][a]$ with probability $1-\varepsilon$, and randomly otherwise. It can be shown (Watkins and Dayan, 1992) that as the algorithm takes more and more actions, each $Q[s][a]$ will converge to some constant $Q_*(s,a)$.

The modern implementation of Q-learning extends beyond the tabular setting and tackles continuous state spaces. It maintains a value function Q_t at each step t such that

$$Q_t \approx \arg\min_{Q} \mathop{\mathbb{E}}_{s,a,r,s'} \left[\left(r + \gamma \max_{a'} Q_{t-1}(s',a') - Q(s,a) \right)^2 \right], \tag{1.2}$$

where (s,a,r,s'), representing the quadruple of state, action, reward, next state, is sampled from data collected by previous interactions, and $\gamma \in [0,1)$ is a discounting factor. Again, actions that maximizes Q_t are taken with priority, e.g., using the ε-greedy strategy as in the tabular case. The most successful implementation of the above paradigm is perhaps deep Q-learning (DQN) (Mnih et al., 2013), in which each Q_t is represented by a (deep) neural network.

As we can see from the above two instances of Q-learning, reward discounting, or more specifically, the discounting factor γ, plays an important role when updating an internally maintained value function.

1.1.2 Policy Gradient

Policy gradient (Sutton et al., 1999) is another RL framework that also has a very long history. It treats the policy, instead of the value function, as the first-class citizen. In modern implementations, the possible policies are parameterized, e.g., by the weights of a neural network. Here we denote a policy by π_θ, where θ is its parameterization. During learning, π_θ is updated with the help of another parameterized function $Q_\omega(s,a)$, which is also frequently updated such that it approximates the γ-discounted return starting from the state-action pair (s,a) and following the policy π_θ, e.g., it is usually updated through gradient descent such that

$$\mathop{\mathbb{E}}_{s,a,r,s'}\left[\left(r+\gamma\cdot\mathbb{E}_{a'\sim\pi_\theta(s')}\left[Q_\omega(s',a')\right]-Q_\omega(s,a)\right)^2\right]\approx 0, \tag{1.3}$$

where (s,a,r,s'), representing the quadruple of state, action, reward, next state, is sampled from data collected under π_θ, and $\gamma\in[0,1)$ is a discounting factor. π_θ is then update through gradient descent where the gradient is

$$\nabla_\theta \mathop{\mathbb{E}}_{s,a}\left[Q_\omega(s,\pi_\theta(s))\right], \tag{1.4}$$

where (s,a), representing the state-action pair, is sampled from data collected under π_θ.

As we can see, reward discounting, or more specifically, the discounting factor γ, is necessary in updating Q_ω, and therefore also intrinsically indispensable in updating π_θ.

1.2 An Example

The previous two sections have already highlighted a discrepancy in current RL practice and theory: reward discounting has become an indispensable ingredient in designing practical RL algorithms; however, standard notions of regret in theoretical RL are not capable of capturing the essence of reward discounting and, as such, fail to serve as optimality criteria for analyzing practical RL algorithms.

In view of this discrepancy, the first question that comes to mind is, perhaps, regarding the role that reward discounting is playing in making RL algorithms more effective in practice. To motivate the discussion, let us consider a concrete example. Consider the MDP illustrated in Figure 1.1. The MDP consists of two sub-MDPs connected by N middle states. One sub-MDP is less rewarding — the learner receives a reward in $[0,1]$ if it takes an action within this sub-MDP; the other one is more rewarding, the learner receives a reward in $[3,4]$ if it takes an action within this sub-MDP. The exact reward function within each sub-MDP does not matter in this example.

The learner starts from a state (marked as initial state in the figure) within the less rewarding sub-MDP, and interacts with the (whole) MDP for T steps. The learner always has the action to stay in a sub-MDP if the learner is already in it, but it may also choose to traverse between the two sub-MDPs following the arrows shown in the figure; however, taking action in any middle state (circles that do not belong to either sub-MDP in the figure) incurs a reward of -1.

Because it only requires $N+1$ steps to arrive at the more-rewarding sub-MDP from the initial state, during which the learner would incur N times the reward -1 and another reward in $[0,1]$ at the initial state, apparently, if $T \geq 2N+1$, a reasonable learner should head right to the more-rewarding sub-MDP from the very beginning and stay there afterward. In fact, doing so would guarantee a total reward of at least $2N$, while otherwise the total reward would be strictly less than $2N$.

On the other hand, as long as $T \leq N$, it makes no sense for the learner to leave the less-rewarding MDP, because the learner cannot reach the more-rewarding sub-MDP anyway, and spending time on middle states only decreases the total reward, and the learner would have to pass the initial state if it wanted to interact with the less-rewarding MDP again after spending time in middle states.

An important observation that can be made from the above example is that, if the total time budget T of a learner is small, it probably should be more myopic in order to earn more rewards during the limited time budget. In other words, it should discount the future rewards according to the total time budget.

1.3 Our Contributions

We address the three questions (Q1)-(Q3) in this dissertation. To answer (Q1), we introduce a new notion of regret, named γ-regret, to capture the concept of regret in discounted RL (Chapter 2). The parameter γ in γ-regret serves as an alternative to the horizon or the diameter in traditional RL analysis. The definition of γ-regret not only captures idea of reward discounting in prevalent RL algorithms but also enables rigorous theoretical analysis of these practices. It is also deeply connected with other regret-related notions in the existing RL literature.

Under the γ-regret framework, both algorithm design and theoretical analysis become more challenging and require innovation. Nonetheless, we are able to make significant progress toward answering (Q2). In particular, we first derive a lower bound of $\Omega\left(\sqrt{T/(1-\gamma)}\right)$, where T is the total number of interactions, on the γ-regret under the tabular setting (Chapter 3). We then introduce two algorithmic instantiations of the Q-learning paradigm, tabular double Q-learning (Chapter 4) and kernelized Q-learning (Chapter 5), both of which are amenable to theoretical analysis under the γ-regret framework.

The tabular double Q-learning algorithm is a theoretification of the double Q-learning paradigm (Hasselt, 2010), which is a variant of the updating rule (1.1). We obtain an upper bound of $\tilde{O}\left(\sqrt{T/(1-\gamma)^3}\right)$ on the γ-regret for tabular double Q-learning (Theorem 4.2.1). Notably, tabular double Q-learning is the first RL algorithm that has near-optimal γ-regret, and since it was introduced, many more RL algorithms has been designed and analyzed under the γ-regret framework (Zhou et al., 2021a;b; He et al., 2020).

Kernelized Q-learning (KQL) is a theoretification of the paradigm (1.2). We obtain an upper bound of $\tilde{O}\left(\sqrt{T/(1-\gamma)^5}\right)$ on the γ-regret for KQL (Theorem 5.4.17). Specifically, we restrict each Q_t to a reproducing kernel Hilbert space (RKHS) and perform optimization using kernel ridge regression, and exploration is done through upper confidence bound.

RKHS are non-parametric function classes that are rich enough in that they are dense in the space of continuous functions if the kernel is *universal* (Sriperumbudur et al., 2011). Therefore, RKHS are, in some sense, as universal as neural networks (Hornik et al., 1989) while being easier to analyze thanks to decades of research on kernel machines. Combining standard machinery for analyzing kernel-based learning with our novel approach to analyze Q-learning, we arrive at general regret bounds for arbitrary kernels (Theorem 5.4.17); in particular, we provide concrete regret bounds for linear kernels (Corollary 5.4.25) and Gaussian RBF kernels

(Corollary 5.4.27). Notably, the latter looks almost identical to the former, only that the actual dimension is replaced by a different dimensionality that is at most polylogarithmic in the number of steps.

We also take a big step toward answering (Q3). Specifically, we complement our theoretical analyses with experiments on a subset of classic control tasks provided in OpenAI Gym (Brockman et al., 2016). We faithfully implement the exact KQL algorithm for which we derive regret bounds and choose hyperparameters based on the bounds. We demonstrate superior sample efficiency of KQL, even when compared with DQN (Section 5.5). To the best of our knowledge, KQL is the first provably low-regret algorithm that excels in commonly used benchmarks; in particular, KQL is the first RL algorithm that can nearly solve the MountainCar environment in as few as one thousand steps.

1.4 Bibliographic Notes

Early regret analysis for RL focused on the tabular setting. Jaksch et al. (2010); Osband and Van Roy (2016) gave regret bounds in the average reward setting, Osband et al. (2016); Azar et al. (2017); Jin et al. (2018); Simchowitz and Jamieson (2019); Zhang and Ji (2019); Russo (2019); Zanette and Brunskill (2019); Zhang et al. (2020; 2021) gave regret bounds in the episodic setting. Regret bounds for the discounted setting was not studied until recently (Liu and Su, 2021; He et al., 2021b), due to trickier definition of the regret.

Later work generalized tabular analyses to the linear model episodic setting (Wang et al., 2019; Jin et al., 2020; Zanette et al., 2020), the linear mixture model episodic setting (Cai et al., 2020; Ayoub et al., 2020; Modi et al., 2020; Yang and Wang, 2020; Zhou et al., 2021a; He et al., 2021a), and the linear mixture model discounted setting (Zhou et al., 2021b).

Bandit problems can be thought as a special case of RL problems where the state space contains a single element, by setting the horizon to 1 in the episodic setting, or setting the discounting factor to 0 in the discounted setting. Naturally, many ideas for RL analyses

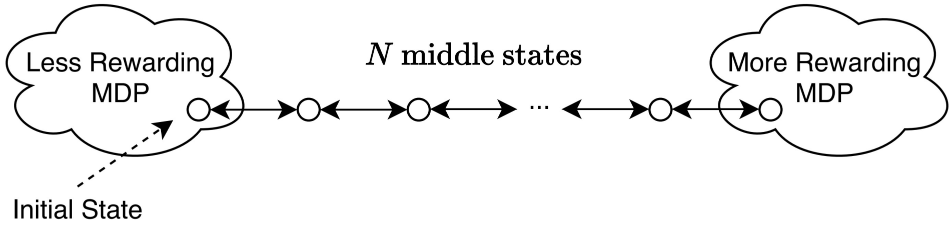

Figure 1.1. An MDP formed by two sub-MDPs connected by *N* middle states.

originated from the bandit literature.

The earliest model for bandit problems is multi-armed bandit (Lai and Robbins, 1985). Later work generalize the multi-armed model to the linear model (Dani et al., 2008; Rusmevichientong and Tsitsiklis, 2010; Abbasi-Yadkori et al., 2011; Li et al., 2021) and the linear contextual model (Auer, 2002; Li et al., 2010; Chu et al., 2011; Li et al., 2019). Kernelization of bandit algorithms are further developed for kernel bandit (Srinivas et al., 2009; 2012; Chowdhury and Gopalan, 2017) and kernel contextual bandit (Krause and Ong, 2011; Valko et al., 2013; Zhou et al., 2020).

Analyses for RL is generally much harder than for bandits, in that it needs to additionally avoid exponential dependencies on the state space and horizon. In particular, (near) optimal dependencies for these quantities require very sophisticated techniques.

Chapter 2

γ-Regret

In this chapter, we will introduce a new notion of regret, named γ-regret, to capture the concept of regret in discounted RL. The parameter γ in γ-regret serves as an alternative to the horizon or the diameter in traditional RL analysis. The definition of γ-regret not only captures idea of reward discounting in prevalent RL algorithms, but also enables rigorous theoretical analysis of these practices. It is also deeply connected with other regret-related notions in the existing RL literature.

2.1 Notations

For a measurable set X, let $\mathcal{P}(X)$ be the set of all probability measures over X.

Let $\mathcal{S}$ and $\mathcal{A}$ be measurable spaces, let

$$\mathbb{P} : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$$

and

$$R : \mathcal{S} \times \mathcal{A} \to \mathcal{P}([0,1])$$

be functions. Consider a Markov decision process (MDP) specified by the state space $\mathcal{S}$, the action space $\mathcal{A}$, the transition function $\mathcal{P}$, and the reward function R. Consider an algorithm that

Algorithm 1. T-Step Reinforcement Learning

```
Parameters: number of steps $T$, state space $\mathcal{S}$, actions space $\mathcal{A}$.
Receive the initial state $s_1$.
for step $t = 1, 2, \cdots, T$
    Take action $a_t$, then observe $r_t \sim R(s_t, a_t)$ and $s_{t+1} \sim \mathbb{P}(s_t, a_t)$
end for
```

interacts with this MDP for T steps. Let us label the steps with integers $1, 2, \cdots, T$. Initially, the algorithm is given an initial state s_0. Then for step $t = 1, 2, \cdots, T$, the algorithm chooses an action $a_t \in \mathcal{A}$; as a consequence, the reward $r_t \sim R(s_t, a_t)$ and the state $s_{t+1} \sim \mathbb{P}(s_t, a_t)$ are revealed to the algorithm. Note that a priori, the algorithm only knows T, $\mathcal{S}$, and $\mathcal{A}$, but not R and $\mathbb{P}$, although it is able to infer R and $\mathbb{P}$ better and better as the interactions go on. The whole process is presented in an algorithmic format in Algorithm 1.

A family of MDPs that are of particular interest is *tabular* MDP, where the state space and the action space are both finite sets. In this case we denote the size of the state space by S and the size of the action space by A.

2.2 The Definition

For any $\gamma \in [0, 1)$, state $s \in \mathcal{S}$, and policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$, denote by $V_\pi(s)$ the expected γ-discounted total rewards generated by starting from state s and following policy π to choose the next action repeatedly, i.e.

$$V_\pi(s) = \mathop{\mathbb{E}}_{\substack{s'_0 = s \\ a'_\tau \sim \pi(s'_\tau) \\ r'_\tau \sim R(s'_\tau, a'_\tau) \\ s'_{\tau+1} \sim \mathbb{P}(s'_\tau, a'_\tau)}} \left[\sum_{\tau=0}^{\infty} \gamma^\tau r'_\tau \right].$$

We can define the maximum γ-discounted total rewards from state s by

$$V_*(s) = \sup_{\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})} V_\pi(s).$$

The γ-regret is defined by

$$\mathrm{Regret}_\gamma(T) = \sum_{t=1}^{T} (1-\gamma) V_*(s_t) - \sum_{t=1}^{T} r_t.$$

Note that here $(1-\gamma)$ is for normalization so that the quantities on both sides of the minus sign have range $[0,T]$.

2.3 Properties

To get a better understanding of the definition of γ-regret, let us assume the T-step reinforcement learning process, as depicted in Algorithm 1, continues to run after step T, indefinitely. The exact behaviour of the algorithm after step T does not matter for our discussion. The algorithm will then continue to observe $r_{T+1}, r_{T+2}, \cdots$. Therefore, we can define

$$\Delta_t = V_*(s_t) - \sum_{\tau=0}^{\infty} \gamma^\tau r_{t+\tau},$$

which essentially captures how optimal the algorithm is from the viewpoint of step t. In order to characterize the overall optimality of the algorithm, we can then consider the quantity

$$\sum_{t=1}^{T} \Delta_t.$$

As it turns out, this quantity is closely related to $\mathrm{Regret}_\gamma(T)$, and it really does not matter too much how the algorithm behaves after step T. Specifically, we have the following lemma.

Lemma 2.3.1.

$$(1-\gamma)\sum_{t=1}^{T} \Delta_t - \frac{1}{1-\gamma} \le \mathrm{Regret}_\gamma(T) \le (1-\gamma)\sum_{t=1}^{T} \Delta_t + \frac{1}{1-\gamma}.$$

Proof. Note that

$$\begin{aligned}
\mathrm{Regret}_\gamma(T) - (1-\gamma)\sum_{t=1}^{T}\Delta_t &= \sum_{t=1}^{T} r_t - (1-\gamma)\sum_{t=1}^{T}\sum_{\tau=0}^{\infty}\gamma^\tau r_{t+\tau} \\
&= \sum_{t=1}^{T} r_t \left(1-(1-\gamma)\sum_{\tau=0}^{t-1}\gamma^\tau\right) \\
&= \sum_{t=1}^{T} r_t \left((1-\gamma)\sum_{\tau=t}^{\infty}\gamma^\tau\right) \\
&\in \left[-\sum_{t=1}^{T} r_t\gamma^t, \sum_{t=1}^{T} r_t\gamma^t\right] \\
&\in \left[-\frac{1}{1-\gamma}, \frac{1}{1-\gamma}\right].
\end{aligned}$$

This concludes the proof. □

This lemma essentially says that, in principle, we could have defined $\mathrm{Regret}_\gamma(T)$ to be $(1-\gamma)\sum_{t=1}^{T}\Delta_t$. However, since this alternative definition depends on how the algorithm behaves after step T, it is better to avoid the (not-so-important) ambiguity and stick to our original definition of γ-regret.

Now let us compare γ-regret with one of the standard definitions of regret in the theoretical RL literature. To this end, let us make the dependency on γ in $V_\pi(s)$ and $V_*(s)$ explicit by writing them as $V_{\pi,\gamma}(s)$ and $V_{*,\gamma}(s)$.

Theoretical analysis on T-step RL are traditional done on MDPs with finite state and action space, through the notion of *average-reward regret*, which is defined by

$$\mathrm{Regret}_*(T) = \sum_{t=1}^{T}\rho^*_{s_t} - \sum_{t=1}^{T} r_t,$$

where

$$\rho^*_s = \sup_{\pi:\mathcal{S}\to\mathcal{P}(A)} \rho^\pi_s$$

and

$$\rho_s^\pi = \lim_{N\to\infty} \frac{1}{N} \mathop{\mathbb{E}}_{\substack{s'_0=s\\ a'_\tau\sim\pi(s'_\tau)\\ r'_\tau\sim R(s'_\tau,a'_\tau)\\ s'_{\tau+1}\sim\mathbb{P}(s'_\tau,a'_\tau)}} \left[\sum_{\tau=0}^{N-1} r'_\tau\right].$$

It can then be shown that for any policy $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$ and state $s \in \mathcal{S}$,

$$\lim_{\gamma\to 1}(1-\gamma)V_{\pi,\gamma}(s) = \rho_s^\pi.$$

Since we assumed the state space and the action space are both finite, there are only finitely many different deterministic policies, and both ρ_s^* and $V_*(s)$ can be attained at a deterministic policy, we arrive at

$$\lim_{\gamma\to 1}(1-\gamma)V_{*,\gamma}(s) = \rho_s^*.$$

Consequently, the average-reward regret can be related to γ-regret by

$$\lim_{\gamma\to 1}\mathrm{Regret}_\gamma(T) = \mathrm{Regret}_*(T).$$

Therefore, γ-regret can be seen as a non-asymptotic version of the average-reward regret.

γ-Regret can also be related to the *sample complexity of exploration*, introduced in Kakade (2003). Specifically, this complexity, which we denote by $N_\gamma(\varepsilon,\delta)$, is the smallest integer such that with probability at least $1-\delta$, assuming the algorithm runs forever, there are at most $N_\gamma(\varepsilon,\delta)$ different t such that

$$\mathbb{E}[\Delta_t] > \varepsilon.$$

However, $N_\gamma(\varepsilon,\delta)$ itself does not measure directly the performance of an algorithm in the first T

steps. For example, an algorithm that (miraculously) incurs the highest possible total reward in the first T steps could have a very large $N_\gamma(\varepsilon, \delta)$ simply because it behaves randomly after T steps.

2.4 Bibliographic Notes

Current analysis of $\text{Regret}_*(T)$ all assume that the MDP is at least weakly communicating, and therefore conveniently ρ_s^* does not depend on s. The analysis was pioneered by Jaksch et al. (2010), who identified the diameter of the MDP, D to be necessary when bounding $\text{Regret}_*(T)$, and in the case $D \ll T$, provided a lower bound of $\Omega(\sqrt{DSAT})$, which is still the best lower bound in terms of D to date. They also derived an upper bound of $\tilde{O}(DS\sqrt{AT})$. Later many more similar bounds were derived under slightly different, but similar assumptions. For instance, Bartlett and Tewari (2012); Fruit et al. (2018) assumed the MDP weakly communicates and the optimal bias vector h^* has bias-span $\text{sp}(h^*) \ll T$; Ortner (2020) chooses to consider the maximal mixing time of the Markov chains induced by all the policies.

In terms of the sample complexity of exploration $N_\gamma(\varepsilon, \delta)$, the best upper bound to date is $\tilde{O}\left(\frac{SA}{\varepsilon^2(1-\gamma)^6}\right)$ (Szita and Szepesvári, 2010), while the best lower bound to date is $\tilde{\Omega}\left(\frac{SA}{\varepsilon^2(1-\gamma)^3}\right)$ (Lattimore and Hutter, 2012).

This chapter is primarily based on material in section 2 and 3 in Liu and Su (2021), of which the dissertation author was the primary researcher and author.

Chapter 3

Lower Bounds

The first step to understand a new notion of regret is, perhaps, by deriving a lower bound of it in a simple scenario, for it not only assures us that the new notion is non-trivial, but also gives us a sense later when we prove upper bounds how tight those bounds are.

In this chapter, we will derive a lower bound on the γ-regret in the tabular setting. Recall that in the tabular setting, the state space has a finite size S and the action space has a finite size A. Such a lower bound, if turns out to be non-trivial, will help us get a better understanding of γ-regret, for if γ regret can be minimized by trivial means, it would not be of any interest, let alone serving as an optimality criterion for RL algorithms.

The following theorem gives a lower bound on the expectation of γ-regret. Notably, it suffices to consider an MDP with only two states.

Theorem 3.0.1. *For any $\gamma \in \left(\frac{2}{3}, 1\right)$, positive integers $A \geq 30, T \geq \frac{A}{1-\gamma}$, and any (possibly randomized) T-step RL algorithm, there exists a two-state MDP such that*

$$\mathbb{E}[\mathrm{Regret}_{\gamma}(T)] \geq \frac{\sqrt{AT}}{2304(1-\gamma)^{\frac{1}{2}}} - \frac{1}{1-\gamma}.$$

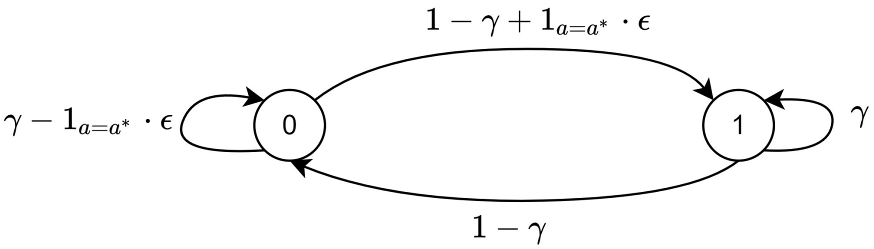
$1-\gamma+1_{a=a^*}\cdot\epsilon$
$\gamma-1_{a=a^*}\cdot\epsilon$
0
1
γ
$1-\gamma$

Figure 3.1. A two-state MDP to prove the lower bound.

3.1 Proof of Theorem 3.0.1

We will construct an MDP similar to the one in the proof of Jaksch et al. (2010, Theorem 5) for our proof. Specifically, the MDP has two states 0 and 1; the learner receives reward 0 in state 0 and reward 1 in state 1, regardless the action taken; the learner goes from state 1 to state 0 with probability $1-\gamma$ regardless the action taken; the learner goes from state 0 to state 1 with probability $1-\gamma+\mathbb{1}_{a=a^*}\cdot\varepsilon$ when action a is taken, where $\varepsilon=\frac{1}{24}\sqrt{\frac{A(1-\gamma)}{T}}$ and a^* will be chosen later. Such an MDP is illustrated in Figure 3.1. It is easy to see that $\varepsilon\leq 1-\gamma$ since we assumed that $T\geq\frac{A}{1-\gamma}$. By definition, we have that

$$V_*(0)=\gamma(1-\gamma+\varepsilon)V_*(1)+\gamma(\gamma-\varepsilon)V_*(0),$$
$$V_*(1)=1+\gamma(1-\gamma)V_*(0)+\gamma^2V_*(1).$$

We can solve the above equations to get

$$V_*(0)=\frac{\gamma-\gamma^2+\gamma\varepsilon}{(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon)}, \tag{3.1}$$
$$V_*(1)=\frac{1-\gamma^2+\gamma\varepsilon}{(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon)}. \tag{3.2}$$

Note that because $\varepsilon \le 1-\gamma$, we have in the denominators of (3.1) and (3.2) that

$$(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon) \in \left[(1-\gamma)^2, 4(1-\gamma)^2\right] \tag{3.3}$$

Let N_0 and N_1 be the number of steps (in the first T steps) that the leaner is in state 0 and 1 respectively, and let N_0^* be the number of steps (in the first T steps) the learner is in state 0 and takes action a^*, using the same argument as in the proof of Jaksch et al. (2010, Theorem 5), we have that

$$E[N_1] \le \frac{T}{2} + \mathbb{E}[N_0^*] \cdot \frac{\varepsilon}{1-\gamma} + \frac{1}{2(1-\gamma)}, \tag{3.4}$$

$$\mathbb{E}[N_0^*] \le \frac{T}{2A} + \frac{1}{2A(1-\gamma)} + \frac{\varepsilon T}{2}\sqrt{\frac{T}{A(1-\gamma)}} + \frac{\varepsilon T}{2(1-\gamma)\sqrt{A}}. \tag{3.5}$$

Therefore,

$$
\begin{aligned}
&(1-\gamma)^{-1} \cdot \mathbb{E}[\mathrm{Regret}_\gamma(T)] \\
&\geq \mathbb{E}[N_0] \cdot V_*(0) + \mathbb{E}[N_1] \cdot \left(V_*(1) - \frac{1}{1-\gamma} \right) \\
&= \frac{\mathbb{E}[N_0] \cdot \gamma(1-\gamma+\varepsilon) - \mathbb{E}[N_1] \cdot \gamma(1-\gamma)}{(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon)} \\
&\overset{(a)}{\geq} \frac{\frac{T\gamma\varepsilon}{2} - \gamma - \frac{\varepsilon\gamma}{2(1-\gamma)} - \mathbb{E}[N_0^*] \cdot \frac{\varepsilon\gamma(2-2\gamma+\varepsilon)}{1-\gamma}}{(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon)} \\
&\overset{(b)}{\geq} \gamma \cdot \frac{\frac{T\varepsilon}{2} - 1 - \frac{\varepsilon}{2(1-\gamma)} - \left(\frac{T}{2A} + \frac{1}{2A(1-\gamma)} + \frac{\varepsilon T}{2}\sqrt{\frac{T}{A(1-\gamma)}} + \frac{\varepsilon T}{2(1-\gamma)\sqrt{A}} \right) \cdot \frac{\varepsilon(2-2\gamma+\varepsilon)}{1-\gamma}}{(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon)} \\
&\overset{(c)}{\geq} \gamma \cdot \frac{\frac{T\varepsilon}{4} - 1 - 3\varepsilon \cdot \left(\frac{5T}{8A} + \frac{\varepsilon T}{2}\sqrt{\frac{T}{A(1-\gamma)}} + \frac{\varepsilon T}{2(1-\gamma)\sqrt{A}} \right)}{(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon)} \\
&\overset{(d)}{=} \gamma \cdot \frac{\frac{\sqrt{AT(1-\gamma)}}{96} - 1 - \left(\frac{\sqrt{AT(1-\gamma)}}{384} + \frac{\sqrt{AT(1-\gamma)}}{192} \right)}{(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon)} \\
&\overset{(e)}{=} \frac{\sqrt{AT(1-\gamma)}}{576(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon)} - \frac{1}{(1-\gamma)(1-2\gamma^2+\gamma+\gamma\varepsilon)} \\
&\overset{(f)}{\geq} \frac{\sqrt{AT}}{2304(1-\gamma)^{1.5}} - \frac{1}{(1-\gamma)^2},
\end{aligned}
$$

where (a) is due to (3.4) and the fact that $N_0 + N_1 = T$, (b) is due to (3.5), (c) is because by assumption $T \geq \frac{A}{1-\gamma} \geq \frac{4}{1-\gamma}$ and $\varepsilon \leq 1-\gamma$, (d) is by substituting ε with the chosen value and recall that by assumption $A \geq 30$ and $T(1-\gamma) \geq 1$, (e) is because by our assumption $\gamma \in \left(\frac{2}{3}, 1\right)$, (f) is due to (3.3). Rearranging the terms concludes the proof.

3.2 Bibliographic Notes

Our proof of the lower bound (Theorem 3.0.1) on γ-regret is an adaptation of the proof for the average-reward setting in Jaksch et al. (2010, Theorem 5). The major challenge the γ-regret formulation brings is that the value function, now being the discounted total return instead of the long-term average, can vary from state to state.

Shortly after our lower bound appeared online, it was improved it to

$$\Omega\left(\sqrt{SAT/(1-\gamma)}\right)$$

by He et al. (2020) using a simple extension of our construction and proof — note the additional $\sqrt{S}$ factor, who also gives a matching upper bound up to logarithmic factors, thus showing the tightness of the lower bounds.

This chapter is primarily based on material in section 4 and 6 in Liu and Su (2021), of which the dissertation author was the primary researcher and author.

Chapter 4

Tabular Double Q-Learning

4.1 The Algorithm

The double Q-learning paradigm, introduced in Hasselt (2010), has become an important variant of the vanilla Q-learning paradigm. In double Q-learning, instead of maintaining a single Q-value function, two Q-value functions are maintained, denoted by $Q^A(s,a)$ and $Q^B(s,a)$. Every time the algorithm takes an action a in a state s, receives a reward r and gets transitioned to a new state s', it will choose one of Q^A and Q^B, e.g. randomly, and if Q^A is chosen, it will perform the update

$$Q^A(s,a) \leftarrow (1-\alpha)Q^A(s,a) + \alpha\left(r + \gamma \max_{a'} Q^B(s',a')\right),$$

and symmetrically if Q^B is chosen, it will perform the update

$$Q^B(s,a) \leftarrow (1-\alpha)Q^B(s,a) + \alpha\left(r + \gamma \max_{a'} Q^A(s',a')\right),$$

where in both cases α is a parameter.

In this section, we will introduce a provably efficient version of the double Q-learning algorithm. We simply name this algorithm tabular double Q-learning, although it has much fewer arbitrariness compared to the one used in practice for it to achieve provable efficiency. We present tabular double Q-learning in Algorithm 2.

Compared to the original neural-network-oriented version, the tabular version has certain specifications that are crucial for theoretical analysis:

- Instead of initializing the Q-value functions arbitrarily as done in the original version, in our version the two Q-value functions have to be initialized to $\frac{1}{1-\gamma}$, or for that matter, the maximal possible discounted cumulative return if it is known. This is rather critical in terms of exploration. In fact, without any prior knowledge on the dynamics of the underlying MDP, setting the initial value of the Q-value functions to the maximal possible value is a necessary condition for the Q-value functions to stay to be an upper confidence bound during the learning process.

- The behavioral policy (i.e., the strategy used to choose which action to take) cannot be arbitrary as in the original version. The action has to be taken greedily based on the Q-value function that immediately gets updated afterward. A greedy behavioral policy is rather standard in designing algorithms that has low regret, for, the Q-value functions in fact serve as upper confidence bounds, and taking actions that has the largest upper confidence bound will make the algorithm to make progress either on the exploitation frontier or the exploration frontier.

- In the original version, in each round, one of the two Q-value functions gets chosen, for example, randomly, and gets updated from the other one. In our version, the updates have to happen in a strictly alternating fashion — the Q-value function gets updated in a round is used for updating the other Q-value function in the next round. The alternating updating scheme may seem to be a natural choice in practice, among other options such as choosing one of the Q-value functions randomly every time; however, it becomes critical in our analysis. In particular, the alternating updating scheme in some sense avoids self-dependency and enables certain terms to cancel out during the process of bounding the γ-regret.

Algorithm 2. Tabular Double Q-Learning

```
**Parameters:** $T$, $\gamma$, $\mathcal{S}$, $\mathcal{A}$, $p$
**Initialize:** $b_t = \frac{2}{1-\gamma}\sqrt{\frac{\ln\left(\frac{\pi^2 SAt^2}{p}\right)}{(1-\gamma)t}}$ for $t \geq 1$, $\alpha_t = \frac{2-\gamma}{1+t-t\gamma}$ for $t \geq 1$

for $(s,a) \in \mathcal{S} \times \mathcal{A}$                                  ▷ start initialization
    $Q^0(s,a), Q^1(s,a) \leftarrow \frac{1}{1-\gamma}$
    $N^0(s,a), N^1(s,a) \leftarrow 0$
end for
receive initial state $s_0$
for step $h = 1, 2, \cdots, T$                                           ▷ main loop
    $\iota \leftarrow h \bmod 2$
    $a_h \leftarrow \arg\max_{a' \in \mathcal{A}} Q^{\iota}(s_h, a')$
    take action $a_h$, then observe $r_h$ and $s_{h+1}$
    $N^{\iota}(s_h, a_h) \leftarrow N^{\iota}(s_h, a_h) + 1$
    $t \leftarrow N^{\iota}(s_h, a_h)$
    $Q^{\iota}(s_h, a_h) \leftarrow (1-\alpha_t)Q^{\iota}(s_h, a_h) + \alpha_t \left(r + b_t + \gamma \max_{a' \in \mathcal{A}} Q^{1-\iota}(s_{h+1}, a')\right)$
end for
```

- Instead of directly updating the Q-value functions using Bellman equations like in the original version, an adjusting term are added to the received reward when updating the Q-value functions so that the maintained Q-value functions stay to be upper confidence bounds throughout the learning process.

4.2 Upper Bounds

In this section, we will establish an upper bound on the regret of the tabular double Q-learning algorithm presented in the previous section. More specifically, we have the following theorem.

Theorem 4.2.1. *For any $\gamma \in [0,1)$ and $p \in (0,1]$, with probability at least $1-p$, for any positive integer T, Algorithm 2 has*

$$\mathrm{Regret}_\gamma(T) \leq \frac{14\sqrt{SAT \ln\left(\frac{\pi^2 SAT^2}{p}\right)}}{(1-\gamma)^{\frac{3}{2}}} + \frac{2SA+4}{1-\gamma},$$

and consequently,

$$\mathbb{E}\left[\mathrm{Regret}_\gamma(T)\right] \leq \frac{14\sqrt{SAT \ln\left(\pi^2 SAT^3\right)}}{(1-\gamma)^{\frac{3}{2}}} + \frac{2SA+5}{1-\gamma}.$$

We can see from the above theorem that the upper bound is of the order

$$\tilde{O}\left(\sqrt{\frac{SAT}{(1-\gamma)^3}}\right).$$

Recall that in the bibliographic notes of the previous chapter we mentioned that the most recent lower bound is of the order

$$\Omega\left(\sqrt{\frac{SAT}{(1-\gamma)}}\right)$$

and there exists an algorithm that has γ-regret of the order

$$\tilde{O}\left(\sqrt{\frac{SAT}{(1-\gamma)}}\right).$$

Therefore, in this sense, the tabular double Q-learning may not be minimax optimal, at least according to our analysis. Nonetheless, it is the first proposed algorithm that has a γ-regret that has square root dependency on T and polynomial dependency on $(1-\gamma)$; it also gives a theoretical justification of the widely used double Q-learning algorithm.

4.2.1 Proof of Theorem 4.2.1

The proof will be in the same style as in Jin et al. (2018), with technical modifications to handle the cyclic dependencies in the non-episodic setting. Recall that in Algorithm 2, for any $t \geq 1$, $\alpha_t = \frac{2-\gamma}{1+t-t\gamma}$. We furthermore define $\alpha_0 = 1$. Let $\alpha_t^i = \alpha_i \prod_{j=i+1}^{t}(1-\alpha_j)$; it is easy to verify that $\sum_{i=0}^{t} \alpha_t^i = 1$. Define by Q_h and N_h the Q^0 and N^0 function at the beginning of iteration h if h is even, or the Q^1 and the N^1 function at the beginning of iteration h if h is odd. Let $n_h = N_h(s_h, a_h)$. For $i = 1, 2, \cdots, n_h$, let $\text{prev}_i(h)$ be the i_{th} smallest $h' < h$ such that h' and h have the same parity, $s_{h'} = s_h$, and $a_{h'} = a_h$. Define

$$
\begin{aligned}
V_h(s) &= \max_a Q_h(s,a), \\
\bar{R}(s,a) &= \mathop{\mathbb{E}}_{r \sim R_*(s,a)}[r], \\
\bar{r}_h &= \bar{R}(s_h, a_h), \\
V_*(\mathbb{P}(s,a)) &= \mathop{\mathbb{E}}_{s' \sim \mathbb{P}(s,a)}\left[V_*(s')\right], \\
Q_*(s,a) &= \bar{R}(s,a) + V_*(\mathbb{P}(s,a)), \\
\phi_h &= V_h(s_h) - V_*(s_h), \\
\delta_h &= \phi_h + \Delta_h.
\end{aligned}
$$

Lemma 4.2.2. *The following statements are true:*

(i). $\frac{\ln(C \cdot t)}{\sqrt{t}} \leq \sum_{i=1}^{t} \alpha_t^i \sqrt{\frac{\ln(C \cdot i)}{i}} \leq 2 \cdot \frac{\ln(C \cdot t)}{\sqrt{t}}$ *for any* $t \geq 1$ *and* $C \geq e$.

(ii). $\sum_{i=1}^{t} \left(\alpha_t^i\right)^2 \leq \frac{2}{(1-\gamma)t}$ *for any* $t \geq 1$.

(iii). $\sum_{t=i}^{\infty} \alpha_t^i = 2 - \gamma$ *for any* $i \geq 1$.

Proof. For (ii) and (iii), the same proof as in Jin et al. (2018), Lemma 4.1.(b)-(c) can be applied, with H replaced by $\frac{1}{1-\gamma}$, and note that in proving (iii) the requirement for n and k to be positive integers in their proof can be relaxed to n and k being real numbers that are at least 1. We will prove (i) by induction on t. The base case $t = 1$ holds because $\alpha_t^1 = 1$. Assuming the statement is true for t, then on one hand,

$$\begin{aligned}\sum_{i=1}^{t+1} \alpha_{t+1}^i \sqrt{\frac{\ln(C\cdot i)}{i}} &\overset{(a)}{=} \alpha_{t+1}\sqrt{\frac{\ln(C\cdot(t+1))}{t+1}} + (1-\alpha_{t+1})\sum_{i=1}^{t}\alpha_t^i\sqrt{\frac{\ln(C\cdot i)}{i}} \\ &\overset{(b)}{\geq} \alpha_{t+1}\sqrt{\frac{\ln(C\cdot(t+1))}{t+1}} + (1-\alpha_{t+1})\sqrt{\frac{\ln(C\cdot t)}{t}} \\ &\overset{(c)}{\geq} \sqrt{\frac{\ln(C\cdot(t+1))}{t+1}},\end{aligned}$$

where in (a) we used the definition of α_t^i, in (b) we used the induction assumption, and (c) is because $x \mapsto \frac{\ln(C\cdot x)}{x}$ is a non-increasing function when $C \geq e$ and $x \geq 1$. On the other hand, we have

$$\begin{aligned}&\sum_{i=1}^{t+1} \alpha_{t+1}^i \sqrt{\frac{\ln(C\cdot i)}{i}} \\ &\overset{(a)}{=} \alpha_{t+1}\sqrt{\frac{\ln(C\cdot(t+1))}{t+1}} + (1-\alpha_{t+1})\sum_{i=1}^{t}\alpha_t^i\sqrt{\frac{\ln(C\cdot i)}{i}} \\ &\overset{(b)}{\leq} \alpha_{t+1}\sqrt{\frac{\ln(C\cdot(t+1))}{t+1}} + 2(1-\alpha_{t+1})\sqrt{\frac{\ln(C\cdot t)}{t}} \\ &\overset{(c)}{=} \frac{2-\gamma}{2+t-(t+1)\gamma}\sqrt{\frac{\ln(C\cdot(t+1))}{t+1}} + \frac{2t(1-\gamma)}{2+t-(t+1)\gamma}\sqrt{\frac{\ln(C\cdot t)}{t}} \\ &\leq \frac{2-\gamma}{2+t-(t+1)\gamma}\sqrt{\frac{\ln(C\cdot(t+1))}{t+1}} + \frac{2\sqrt{t}(1-\gamma)\sqrt{t+1}}{2+t-(t+1)\gamma}\sqrt{\frac{\ln(C\cdot(t+1))}{t+1}} \\ &\leq \frac{2+2(t+1)(1-\gamma)}{1+(t+1)(1-\gamma)}\sqrt{\frac{\ln(C\cdot(t+1))}{t+1}} \\ &= 2\sqrt{\frac{\ln(C\cdot(t+1))}{t+1}},\end{aligned}$$

where in (a) we used the definition of α_t^i, in (b) we used the induction assumption, and in (c) we used the definition of α_t. Therefore, the statement in (i) is true for any $t \geq 1$. □

Lemma 4.2.3. *For any h,*

$$\begin{aligned}
&Q_h(s_h,a_h) - Q_*(s_h,a_h) \\
&= \alpha_{n_h}^0 \left(\frac{1}{1-\gamma} - Q_*(s,a)\right) + \sum_{i=1}^{n_h} \alpha_{n_h}^i b_i + \gamma \sum_{i=1}^{n_h} \alpha_{n_h}^i \phi_{\text{prev}_h^i+1} \\
&\quad + \sum_{i=1}^{n_h} \alpha_{n_h}^i \left(r_{\text{prev}_h^i} - \bar{r}_{\text{prev}_h^i} + \gamma\left(V_*\left(s_{\text{prev}_h^i+1}\right) - V_*(\mathbb{P}(s_h,a_h))\right)\right).
\end{aligned}$$

Proof. We have that

$$Q_h(s_h,a_h) = \alpha_{n_h}^0 \frac{1}{1-\gamma} + \sum_{i=1}^{n_h} \alpha_{n_h}^i \left(r_{\text{prev}_h^i} + b_i + \gamma V_{\text{prev}_h^i+1}\left(s_{\text{prev}_h^i+1}\right)\right)$$

and

$$\begin{aligned}
Q_*(s_h,a_h) &\overset{(a)}{=} \sum_{i=0}^{n_h} \alpha_{n_h}^i \left(\bar{r}_{\text{prev}_h^i} + \gamma V_*(\mathbb{P}(s_h,a_h))\right) \\
&= \alpha_{n_h}^0 Q_*(s,a) + \sum_{i=1}^{n_h} \alpha_{n_h}^i \left(\bar{r}_{\text{prev}_h^i} + \gamma V_*(\mathbb{P}(s_h,a_h))\right),
\end{aligned}$$

where in (a) we used the fact that $\sum_{i=0}^{t} \alpha_t^i = 1$ for any t and the definition of $Q_*(s,a)$. Therefore we have

$$\begin{aligned}
&Q_h(s_h,a_h) - Q_*(s_h,a_h) \\
&= \alpha_{n_h}^0 \left(\frac{1}{1-\gamma} - Q_*(s,a)\right) + \\
&\quad \sum_{i=1}^{n_h} \alpha_{n_h}^i \left(r_{\text{prev}_h^i} - \bar{r}_{\text{prev}_h^i} + b_i + \gamma\left(V_{\text{prev}_h^i+1}(s_{\text{prev}_h^i+1}) - V_*(\mathbb{P}(s_h,a_h))\right)\right) \\
&= \alpha_{n_h}^0 \left(\frac{1}{1-\gamma} - Q_*(s,a)\right) + \\
&\quad \sum_{i=1}^{n_h} \alpha_{n_h}^i b_i + \gamma \sum_{i=1}^{n_h} \alpha_{n_h}^i \phi_{\text{prev}_h^i+1} + \\
&\quad \sum_{i=1}^{n_h} \alpha_{n_h}^i \left(r_{\text{prev}_h^i} - \bar{r}_{\text{prev}_h^i} + \gamma\left(V_*(s_{\text{prev}_h^i+1}) - V_*(\mathbb{P}(s_h,a_h))\right)\right).
\end{aligned}$$

This concludes the proof. □

Lemma 4.2.4. *Define random variables $r_{s,a,i}$ to be the reward received after taking action a on state s the i_{th} time, and $s'_{s,a,i}$ to be the next state when receiving reward $r_{s,a,i}$, then for any T, with probability at least $1-p$, the following hold simultaneously*

(i). For any h, $0 \leq Q_h(s_h,a_h) - Q_*(s_h,a_h) \leq \frac{\alpha^0_{n_h}}{1-\gamma} + \gamma \sum_{i=1}^{n_h} \alpha^i_{n_h} \phi_{\mathrm{prev}^i_h+1} + 3\beta_{n_h}$,

where $\beta_t = \frac{2}{1-\gamma}\sqrt{\frac{\ln\left(\frac{\pi^2 SAt^2}{p}\right)}{(1-\gamma)t}}$ if $t \geq 1$ and $\beta_0 = 0$.

(ii). $\sum_{h=1}^{T} \bar{r}_h - r_h + \gamma(V_*(\mathbb{P}(s_h,a_h)) - V_*(s_{h+1})) \leq \xi_T$, where $\xi_T = \frac{\sqrt{2}}{1-\gamma}\sqrt{T\ln\left(\frac{3}{2p}\right)}$.

Proof. It suffices to show that (i) happens with probability at least $1-\frac{p}{3}$ and (ii) happens with probability at least $1-\frac{2p}{3}$.

We focus on (i) first. The case where $n_h = 0$ is trivial, so we assume $n_h \geq 1$. Fix any s,a,t, let $x_i = \alpha^i_t\left(r_{s,a,i} - \bar{R}(s,a) + \gamma\left(V_*(s'_{s,a,i}) - V_*(\mathbb{P}(s,a))\right)\right)$. We can see that $\{x_i\}_{i=1}^t$ is a Martingale difference sequence and $|x_i| \leq \frac{\alpha^i_t}{1-\gamma}$, therefore by Azuma-Hoeffding inequality we have that with probability at least $1-\frac{2p}{\pi^2 SAt^2}$,

$$\left|\sum_{i=1}^{t} x_i\right| \leq \frac{1}{1-\gamma}\sqrt{2\ln\left(\frac{\pi^2 SAt^2}{p}\right)\sum_{i=1}^{t}\left(\alpha^i_t\right)^2}$$
$$\overset{(a)}{\leq} \frac{1}{1-\gamma}\sqrt{\frac{4\ln\left(\frac{\pi^2 SAt^2}{p}\right)}{(1-\gamma)t}} = \beta_t,$$

where in (a) we used Lemma 4.2.2.(ii). Using a union bound, the above inequalities hold for all s, a, $t \geq 1$ with probability at least

$$1 - SA\sum_{t=1}^{\infty}\frac{2p}{\pi^2 SAt^2} = 1 - \frac{p}{3}.$$

According to Lemma 4.2.3, it suffices to show that with probability at least $1-\frac{p}{3}$, we have that for any s,a,t,

$$0 \leq \sum_{i=1}^{t} \alpha_t^i b_i + \sum_{i=1}^{t} \alpha_t^i \left(r_{s,a,i} - \bar{R}(s,a) + \gamma \left(V_*(s'_{s,a,i}) - V_*(\mathbb{P}(s,a)) \right) \right) \leq 3\beta_t, \tag{4.1}$$

and then the first inequality in (i) follows by induction and the second inequality in (i) follows naturally. In fact, to see (4.1), first note that by Lemma 4.2.2.(i) we have that

$$\beta_t \leq \sum_{i=1}^{t} a_t^i b_i \leq 2\beta_t$$

and the previous arguments showed that with probability at least $1 - \frac{p}{3}$ we have that for any s,a,t,

$$\left| \sum_{i=1}^{t} \alpha_t^i \left(r_{s,a,i} - \bar{R}(s,a) + \gamma \left(V_*(s'_{s,a,i}) - V_*(\mathbb{P}(s,a)) \right) \right) \right| \leq \beta_t.$$

This concludes the proof that (i) is true with probability at least $1 - \frac{p}{3}$.

Next we focus on (ii). Let $y_h = \bar{r}_h - r_h + \gamma(V_*(\mathbb{P}(s_h,a_h)) - V_*(s_{h+1}))$. We can see that $\{y_h\}_{i=1}^{T}$ is a Martingale difference sequence and $|y_h| \leq \frac{1}{1-\gamma}$, therefore by Azuma-Hoeffding inequality we have that with probability at least $1 - \frac{2p}{3}$,

$$\sum_{h=1}^{T} y_h \leq \frac{1}{1-\gamma} \sqrt{2T \ln\left(\frac{3}{2p}\right)} = \xi_T.$$

This concludes the proof that (ii) is true with probability at least $1 - \frac{2p}{3}$. □

We are now ready to begin our proof. From now on all the calculation will condition on the events where the statements in Lemma 4.2.4 are true. It is important to notice that in this case we have that for any h, $\phi_h \geq 0$ and $\Delta_h \leq \delta_h$. First note that

$$\begin{aligned}
\delta_h &= Q_h(s_h, a_h) - \sum_{t=0}^{\infty} \gamma^t r_{h+t} \\
&= (Q_h(s_h, a_h) - Q_*(s_h, a_h)) + \left(Q_*(s_h, a_h) - \sum_{t=0}^{\infty} \gamma^t r_{h+t} \right) \\
&\overset{(a)}{\leq} \alpha_{n_h}^0 \cdot \frac{1}{1-\gamma} + \gamma \sum_{i=1}^{n_h} \alpha_{n_h}^i \phi_{\mathrm{prev}_h^i + 1} + 3\beta_{n_h} + \\
&\qquad \bar{r}_h - r_h + \gamma \left(V_*(\mathbb{P}(s_h, a_h)) - \sum_{t=0}^{\infty} \gamma^t r_{h+1+t} \right) \\
&\leq \alpha_{n_h}^0 \cdot \frac{1}{1-\gamma} + \gamma \sum_{i=1}^{n_h} \alpha_{n_h}^i \phi_{\mathrm{prev}_h^i + 1} + 3\beta_{n_h} + \\
&\qquad \gamma(\delta_{h+1} - \phi_{h+1}) + (\bar{r}_h - r_h) + \gamma(V_*(\mathbb{P}(s_h, a_h)) - V_*(s_{h+1})),
\end{aligned}$$

where (a) is due to Lemma 4.2.4.(i). Therefore, according to Lemma 4.2.4.(ii) we have

$$\begin{aligned}
&\sum_{h=1}^{T} \delta_h \\
&\leq \frac{1}{1-\gamma} \sum_{h=1}^{T} \alpha_{n_h}^0 + \gamma \sum_{h=1}^{T} \sum_{i=1}^{n_h} \alpha_{n_h}^i \phi_{\mathrm{prev}_h^i + 1} + \gamma \sum_{h=1}^{T} \delta_{h+1} - \gamma \sum_{h=1}^{T} \phi_{h+1} + \xi_T + 3 \sum_{h=1}^{T} \beta_{n_h}.
\end{aligned}$$

Using the fact that $|\delta_h| \leq \frac{1}{1-\gamma}$ for any h and rearranging the terms, we get

$$\begin{aligned}
&(1-\gamma) \sum_{h=1}^{T} \delta_h \\
&\leq \frac{1}{1-\gamma} \sum_{h=1}^{T} \alpha_{n_h}^0 + \gamma \left(\sum_{h=1}^{T} \sum_{i=1}^{n_h} \alpha_{n_h}^i \phi_{\mathrm{prev}_h^i + 1} - \sum_{h=1}^{T} \phi_{h+1} \right) + \xi_T + 3 \sum_{h=1}^{T} \beta_{n_h} + \frac{2\gamma}{1-\gamma}.
\end{aligned} \tag{4.2}$$

To continue the calculation, first note that $\alpha_{n_h}^0$ is 1 if $n_h = 0$ and is 0 otherwise, therefore

$$\begin{aligned}
\sum_{h=1}^{T} \alpha_{n_h}^0 &= \sum_{h=1}^{T} \mathbb{1}_{n_h = 0} \\
&\leq 2SA.
\end{aligned}$$

Next note that

$$
\begin{aligned}
&\sum_{h=1}^{T}\sum_{i=1}^{n_h}\alpha_{n_h}^{i}\phi_{\mathrm{prev}_h^i+1}-\sum_{h=1}^{T}\phi_{h+1}\\
&\leq \sum_{h=1}^{T}\phi_h\sum_{t=n_h+1}^{\infty}\alpha_t^{n_h}-\sum_{h=1}^{T}\phi_{h+1}\\
&\overset{(a)}{\leq} \frac{2-\gamma}{2}\sum_{h=1}^{T}\phi_h-\sum_{h=1}^{T}\phi_h+\phi_0\\
&=(1-\gamma)\sum_{h=1}^{T}\phi_h+\frac{1}{1-\gamma}\\
&=(1-\gamma)\sum_{h=1}^{T}(\delta_h-\Delta_h)+\frac{1}{1-\gamma},
\end{aligned}
$$

where (a) is because of Lemma 4.2.2.(iii). Now going back to (4.2) we get

$$
\begin{aligned}
&(1-\gamma)\sum_{h=1}^{T}\delta_h\leq\frac{2SA+3}{1-\gamma}+(1-\gamma)\sum_{h=1}^{T}(\delta_h-\Delta_h)+\xi_T+3\sum_{h=1}^{T}\beta_{n_h}\\
\iff\ &(1-\gamma)\sum_{h=1}^{T}\Delta_h\leq\frac{2SA+3}{1-\gamma}+\xi_T+3\sum_{h=1}^{T}\beta_{n_h}
\end{aligned}
$$

Finally, note that

$$\begin{aligned}
\sum_{h=1}^{T} \beta_{n_h} &\leq \frac{2\sqrt{\ln\left(\frac{\pi^2 SAT^2}{p}\right)}}{(1-\gamma)^{1.5}} \sum_{h=1}^{T} \mathbb{1}_{n_h \geq 1} \cdot \sqrt{\frac{1}{n_h}} \\
&= \frac{2\sqrt{\ln\left(\frac{\pi^2 SAT^2}{p}\right)}}{(1-\gamma)^{1.5}} \sum_{s,a} \sum_{t=1}^{N_T(s,a)} \sqrt{\frac{1}{t}} \\
&\overset{(a)}{\leq} \frac{2\sqrt{\ln\left(\frac{\pi^2 SAT^2}{p}\right)}}{(1-\gamma)^{1.5}} \sum_{s,a} 2\sqrt{N_T(s,a)} \\
&\overset{(b)}{\leq} \frac{4\sqrt{SAT\ln\left(\frac{\pi^2 SAT^2}{p}\right)}}{(1-\gamma)^{1.5}},
\end{aligned}$$

where (a) is because $\sum_{i=1}^{t} \sqrt{\frac{1}{i}} \leq 2\sqrt{t}$ and (b) is by Cauchy-Schwarz inequality and the fact that $\sum_{s,a} N_T(s,a) \leq T$. Therefore,

$$\begin{aligned}
&(1-\gamma)\sum_{h=1}^{T} \Delta_h \\
&\leq \frac{2SA+3}{1-\gamma} + \frac{\sqrt{2}}{1-\gamma}\sqrt{T\ln\left(\frac{3}{2p}\right)} + \frac{12\sqrt{SAT\ln\left(\frac{\pi^2 SAT^2}{p}\right)}}{(1-\gamma)^{1.5}} \qquad (4.3) \\
&\leq \frac{2SA+3}{1-\gamma} + \frac{14\sqrt{SAT\ln\left(\frac{\pi^2 SAT^2}{p}\right)}}{(1-\gamma)^{1.5}}
\end{aligned}$$

On the other hand, we have

$$
\begin{aligned}
&(1-\gamma)\sum_{h=1}^{T}\Delta_h \\
&= \sum_{h=1}^{T}(1-\gamma)V_*(s_h) - (1-\gamma)\sum_{h=1}^{T}\sum_{t=0}^{\infty}\gamma^t r_{h+t} \\
&\geq \sum_{h=1}^{T}(1-\gamma)V_*(s_h) - \sum_{u=1}^{T} r_u - (1-\gamma)\sum_{u=T}^{\infty} r_u \sum_{v=u-T+1}^{u}\gamma^v \\
&= \mathrm{Regret}_\gamma(T) - (1-\gamma)\sum_{u=T}^{\infty} r_u \sum_{v=u-T+1}^{u}\gamma^v \\
&\geq \mathrm{Regret}_\gamma(T) - \frac{1}{1-\gamma}.
\end{aligned}
\tag{4.4}
$$

Combining together (4.3) and (4.4), we arrive at

$$
\mathrm{Regret}_\gamma(T) \leq \frac{14\sqrt{SAT\ln\left(\frac{\pi^2 SAT^2}{p}\right)}}{(1-\gamma)^{1.5}} + \frac{2SA+4}{1-\gamma}.
$$

This concludes the proof.

4.2.2 Relation to Sample Complexity of Exploration

Recall that the sample complexity of exploration $N_\gamma(\varepsilon,\delta)$ is the smallest integer such that with probability at least $1-\delta$, assuming the algorithm runs forever, there are at most $N_\gamma(\varepsilon,\delta)$ different t such that

$$
\mathbb{E}\left[\Delta_t\right] > \varepsilon.
$$

It is easy to see that

$$
\mathbb{E}\left[\mathrm{Regret}_\gamma(T)\right] \in O\left(\inf_{\varepsilon} N_\gamma\left(\varepsilon, \frac{1}{T}\right) + \varepsilon T(1-\gamma) + \frac{1}{1-\gamma}\right).
$$

Plugging in the best existing bound for $N_\gamma(\varepsilon,\delta)$, which is $\tilde{O}\left(\frac{SA}{\varepsilon^2(1-\gamma)^6}\right)$ from Szita and Szepesvári

(2010), we arrive at an upper bound of $\tilde{O}\left(\frac{T^{\frac{2}{3}}(SA)^{\frac{1}{3}}}{(1-\gamma)^{\frac{4}{3}}}\right)$ on $\mathbb{E}\left[\text{Regret}_\gamma(T)\right]$. It may seem that this bound has better dependencies on S, A, and γ than those from our tabular double Q-learning, but this is not the case. In fact, we have the following inequalities:

$$\tilde{O}\left(\frac{T^{\frac{2}{3}}(SA)^{\frac{1}{3}}}{(1-\gamma)^{\frac{4}{3}}}\right) \geq \begin{cases} \tilde{O}(T), & \text{if } T < \dfrac{SA}{(1-\gamma)^4}, \\ \tilde{O}\left(\dfrac{\sqrt{SAT}}{(1-\gamma)^2}\right), & \text{otherwise.} \end{cases} \tag{4.5}$$

Note that in the above inequalities $\tilde{O}(T)$ is a trivial upper bound on $\text{Regret}_\gamma(T)$ for any T, while $\tilde{O}\left(\frac{\sqrt{SAT}}{(1-\gamma)^2}\right)$ has a worse dependency on γ than our upper bound.

If the upper bounds on $N_\gamma(\varepsilon,\delta)$ were to hold uniformly over all possible ε, then we could translate the (uniform) upper bound on $N_\gamma(\varepsilon,\delta)$ into upper bounds on γ-regret in a better way. In fact, if the best existing upper bound on $N_\gamma(\varepsilon,\delta)$, $\tilde{O}\left(\frac{SA}{\varepsilon^2(1-\gamma)^6}\right)$, were to hold uniformly over all possible ε, then we would have

$$\mathbb{E}\left[\text{Regret}_\gamma(T)\right] \in O\left((1-\gamma)\left(\int_{\varepsilon_0}^{\frac{1}{1-\gamma}} \frac{SA}{\varepsilon^2(1-\gamma)^6} + T\varepsilon_0\right) + \frac{1}{1-\gamma}\right);$$

in other words, we could get an upper bound on γ-regret as good as $\tilde{O}\left(\frac{\sqrt{SAT}}{(1-\gamma)^2}\right)$. We can see that even in this imagined ideal scenario the translated upper bound still has a worse dependency on γ than ours.

4.3 Bibliographic Notes

Our tabular double Q-learning algorithm and its analysis are inspired by Jin et al. (2018), who showed that a specific tabular version of Q-Learning (Watkins and Dayan, 1992) has near-optimal regret in the episodic setting. Their algorithm and analysis, however, are not directly applicable to the non-episodic setting, for the following two reasons:

First, in the episodic setting, there are H value functions $Q_0, Q_1, \cdots, Q_{H-1}$ to be learned, each Q_i depends only on $Q_{j>i}$ — there is no cyclic dependency; on the other hand, in the non-episodic setting, there is only one single value function, so a hierarchical induction in the analysis is not possible. To deal with self-dependency, we find it very useful to replace the regular Q-learning with double Q-learning (Hasselt, 2010), which has been widely used in deep reinforcement learning since it was introduced (Hasselt et al., 2016; Hessel et al., 2018).

Second, a key ingredient in the proof of Jin et al. (2018) is the choice of learning rate $\alpha_t = \frac{H+1}{H+t}$ — a nice consequence of this choice is that the total per-episode-step regret blow-up is $(1+1/H)$; since there are at most H steps in each episode, the total blow-up is $(1+1/H)^H$, which is upper bounded by the constant e regardless how large H is. The same quantity $(1+1/H)^H$ also appeared in Azar et al. (2017) for the same reason. However, in the non-episodic setting, the blow-up could become arbitrarily large because the learner is not reset every H steps; therefore, different techniques are required to control the blow-up of the regret.

Our tabular double Q-learning algorithm also looks visually similar to Dong et al. (2019, Algorithm 1); however, the goal of Dong et al. (2019) is to propose a model-free algorithm that has low sample complexity of exploration, and the proof technique therein is very different from ours.

Episodic regret bounds have also been studied, and a minimax episodic regret of $\sqrt{HSAT}$ has been proved (Azar et al., 2017; Zanette and Brunskill, 2019), where H is the episode length.

This chapter is primarily based on material in section 5 and 6 in Liu and Su (2021), of which the dissertation author was the primary researcher and author.

Chapter 5

Kernelized Q-Learning

In the previous chapter, we proposed an algorithm that works well in the tabular setting in terms of the γ-regret. Naturally, we want to go beyond the tabular setting and tackle linear or even non-linear reward functions and transition functions, and this will be the topic of this chapter.

5.1 Preliminaries

For a measurable space X, let $\mathcal{B}(X)$ be the space of real-valued bounded measurable functions over X equipped with the supremum norm $\|\cdot\|_\infty$. For any positive integer n, denote by $[n]$ the set $\{1, 2, \cdots, n\}$.

Let $\mathcal{H}$ be a Hilbert space. Denote by $\langle\cdot,\cdot\rangle_{\mathcal{H}}$ the inner product on $\mathcal{H}$. Denote by $\|\cdot\|_{\mathcal{H}}$ the norm induced by the inner product. If $T : \mathcal{H} \to \mathcal{H}$ is a bounded self-adjoint positive-definite linear operator, define the Mahalanobis norm $\|f\|_T = \sqrt{\langle Tf, f\rangle_{\mathcal{H}}}$. Denote by $\mathcal{B}_{\mathrm{HS}}(\mathcal{H})$ the space of Hilbert-Schmidt operators from $\mathcal{H}$ to $\mathcal{H}$. Denote by $\langle\cdot,\cdot\rangle_{\mathrm{HS}}$ the inner product on $\mathcal{B}_{\mathrm{HS}}(\mathcal{H})$. Denote by $\|\cdot\|_{\mathrm{HS}}$ the norm induced by the inner product. For any $u, v \in \mathcal{H}$, denote by $u \otimes v$ the linear operator such that for any $f \in \mathcal{H}$, $(u \otimes v)f = \langle u, f\rangle_{\mathcal{H}} \cdot v$.

Let $\mathcal{H}$ be a real-valued RKHS over a set X such that the corresponding kernel $\mathcal{K}$ is bounded. For any $x \in X$, denote by $\mathcal{K}_x$ the reproducing function at x, i.e., $\mathcal{K}_x$ is the unique function that satisfies for any $f \in \mathcal{H}$, $f(x) = \langle f, \mathcal{K}_x\rangle_{\mathcal{H}}$. In fact, $\mathcal{K}_x = (x \mapsto \mathcal{K}(x,\cdot))$. For any

$f \in \mathcal{H}$, define the norm

$$\|f\|_* = \sup_{x \in X} \langle f, \mathcal{K}_x \rangle_{\mathcal{H}}.$$

For any bounded linear operator $T : \mathcal{H} \to \mathcal{H}$, define the norm

$$\|T\|_* = \sup_{x \in X} \langle T\mathcal{K}_x, \mathcal{K}_x \rangle_{\mathcal{H}}.$$

5.2 Assumptions

To simplify the exposition, let us assume the reward function is deterministic, and denote it by $r_*(s,a)$. In other words, $R_*(s,a)$ is a Dirac measure at $r_*(s,a)$. Our analysis can be easily generalized to incorporate stochastic reward functions.

Let $\mathcal{H}$ be a real-valued RKHS defined on $\mathcal{S} \times \mathcal{A}$ such that the corresponding kernel $\mathcal{K} : (\mathcal{S} \times \mathcal{A})^2 \to \mathbb{R}$ is bounded. Without loss of generality, we will assume $\|\mathcal{K}\|_\infty > 0$; and we further assume $\mathcal{K}$ is properly normalized so that for any $(s,a) \in \mathcal{S} \times \mathcal{A}$, $\left\|\mathcal{K}_{(s,a)}\right\|_{\mathcal{H}} \in [1/2, 1]$.

Let us convert $\mathbb{P}$ into a linear operator by defining

$$\begin{aligned} M_* : \mathcal{B}(\mathcal{S}) &\to \mathcal{B}(\mathcal{S} \times \mathcal{A}) \\ f &\mapsto \left((s,a) \mapsto \mathbb{E}_{\mathbb{P}(s,a)}[f]\right). \end{aligned}$$

We will assume there exists an (unknown) $M : \mathcal{B}(\mathcal{S}) \to \mathcal{H}$ such that

$$\sup_{\substack{f \in \mathcal{B}(\mathcal{S}) \\ \|f\|_\infty \leq 1}} \|(M - M_*)f\|_\infty \leq \varepsilon.$$

The operator norm of M is defined by

$$\|M\| = \sup_{\substack{f\in\mathcal{B}(\mathcal{S})\\ \|f\|_\infty\le 1}} \|M(f)\|_{\mathcal{H}}.$$

We assume $\|M\| \le \rho$. Similarly, for the reward function, we will also assume there exists an (unknown) $r \in \mathcal{H}$ such that

$$\|r_* - r\|_\infty \le \varepsilon, \quad \|r\|_{\mathcal{H}} \le \rho.$$

Let $\sigma \in [0,1]$ be a bound on the stochasticity of the MDP. Specifically, for any $(s,a) \in \mathcal{S}\times\mathcal{A}$, $f \in \mathcal{B}(\mathcal{S})$ such that $\|f\|_\infty \le 1$, $f(s')$ is σ-sub-Gaussian when $s' \sim \mathbb{P}(s,a)$. Clearly, σ can always be set to 1, and for a deterministic MDP, σ can be set to 0.

5.3 The Algorithm

We propose kernelized Q-learning (KQL) (Algorithm 3). It is a theoretification of the deep Q-learning paradigm in the context of kernel spaces. It can run in time $\mathcal{O}((T+g)T^2|\mathcal{A}|)$, where g is the time to compute $x,y \mapsto \mathcal{K}(x,y)$. When the kernel is linear, the running time can be improved (by special implementation) to $\mathcal{O}(Td^2|\mathcal{A}|)$, where d is the dimension of the feature space that $\mathcal{S}\times\mathcal{A}$ is embedded into.

We list closed-form representations of the variables maintained by Algorithm 3 below. In fact, our results in this chapter hold for any algorithm for which these variables are maintained.

Fact 5.3.1. *Denote by $\mathcal{K}_\tau = \mathcal{K}_{(s_\tau,a_\tau)}$ and $r_\tau = r_*(s_\tau,a_\tau)$. Algorithm 3 satisfies*

(1) For any $t\in[T]$, $\widehat{W}_t = \left(\sum_{\tau=1}^{t-1}\mathcal{K}_\tau\otimes\mathcal{K}_\tau + \lambda I\right)^{-1}$.

(2) For any $t\in[T]$,

$$\widehat{Q}_t = \left(\sum_{\tau=1}^{t-1}\left(r_\tau + \gamma\max_a \widetilde{Q}_{t-1}[\tau+1][a]\right)\mathcal{K}_\tau\right)\widehat{W}_t.$$

Algorithm 3. Kernelized Q-Learning

```
Parameters: $T$, $\gamma$, $\mathcal{K}$, $\lambda > 0$, $0 \leq \beta \leq \frac{2\sqrt{T+\lambda}}{1-\gamma}$.
Initialize: $\widehat{W}_1 = \frac{1}{\lambda} I$. $\widehat{Q}_1 = 0$.
Receive the initial state $s_1$.
for step $t = 1, 2, \cdots, T$
    for $a \in \mathcal{A}$
        $\widehat{\Gamma}_t[t][a] = \left\|\mathcal{K}_{(s_t,a)}\right\|_{\widehat{W}_t}$.                                  ▷ $\mathcal{O}(T^2)$.
        $\widetilde{Q}_t[t][a] = \mathrm{clip}_{[0,1/(1-\gamma)]}\left(\widehat{Q}_t(s_t,a) + \beta \cdot \widehat{\Gamma}_t[t][a]\right)$          ▷ $\mathcal{O}(T)$.
    end for
    Take action $a_t = \arg\max_a \widetilde{Q}_t[t][a]$, observe $s_{t+1} \sim \mathbb{P}(s_t, a_t)$.      ▷ $\mathcal{O}(T^2|\mathcal{A}|)$.
    for $\tau = 1, 2, \cdots, t-1$
        for $a \in \mathcal{A}$
            $\widetilde{Q}_t[\tau][a] = \mathrm{clip}_{[0,1/(1-\gamma)]}\left(\widehat{Q}_t(s_\tau,a) + \beta \cdot \widehat{\Gamma}_t[\tau][a]\right)$     ▷ $\mathcal{O}(T)$.
        end for
    end for
    $u = \widehat{W}_t \mathcal{K}_{(s_t,a_t)}$.                                                   ▷ $\mathcal{O}(T^2)$.
    $s = \left\langle u, \mathcal{K}_{(s_t,a_t)}\right\rangle_{\mathcal{H}}$.                                               ▷ $\mathcal{O}(T^2)$.
    $\widehat{W}_{t+1} = \widehat{W}_t - \frac{u \otimes u}{1+s}$.                                            ▷ $\mathcal{O}(T^2)$.
    for $\tau = 1, 2, \cdots, t$
        for $a \in \mathcal{A}$
            $\widehat{\Gamma}_{t+1}[\tau][a] = \sqrt{\left(\widehat{\Gamma}_t[\tau][a]\right)^2 - \frac{u^2(s_\tau,a)}{1+s}}$.                       ▷ $\mathcal{O}(T)$.
        end for
    end for
    $\widehat{Q}_{t+1} = \left(\sum_{\tau=1}^t \left(r_*(s_\tau,a_\tau) + \gamma \max_a \widetilde{Q}_t[\tau+1][a]\right) \cdot \mathcal{K}_{(s_\tau,a_\tau)}\right) \widehat{W}_{t+1}$.   ▷ $\mathcal{O}(T^2|\mathcal{A}|)$.
end for
```

(3) For any $\tau, t \in [T]$ such that $\tau \leq t$, $a \in \mathcal{A}$

$$\widetilde{Q}_t[\tau][a] = \underset{\left[0,\frac{1}{1-\gamma}\right]}{\mathrm{clip}} \left(\widehat{Q}_t(s_\tau,a) + \beta \left\|\mathcal{K}_{(s_\tau,a)}\right\|_{\widehat{W}_t}\right).$$

5.4 Upper Bounds

The goal of this section is to derive γ-regret upper bounds for KQL. We will introduce the necessary technicalities in the first two sections and dive into the bounds afterwards.

5.4.1 Effective Dimension and Pseudo Dimension

We are going to introduce two concepts used in the learning literature to capture the properties of a RKHS, *effective dimension* and *pseudo dimension*. Note that many similar concepts have been introduced (see Srinivas et al. (2009; 2012); Valko et al. (2013); Chowdhury and Gopalan (2017); Yang and Wang (2020); Yang et al. (2020) among many others) under various names such as *information gain*. All these variants are in some sense equivalent to each other, in that they all capture the effective dimensionality of a RKHS, perhaps up to a logarithmic factor.

In this section, let X be a set, $\mathcal{K} : X \times X \to \mathbb{R}$ be a kernel, n be a positive integer, $x_{1:n} = (x_1, x_2, \cdots, x_n) \in X^n$, and $\lambda > 0$, $\mathfrak{K}$ be a $n \times n$ matrix where $\mathfrak{K}[i][j] = \mathcal{K}(x_i, x_j)$, $\Sigma = \sum_{i=1}^{n} \mathcal{K}_{x_i} \otimes \mathcal{K}_{x_i}$.

Effective Dimension

Definition 5.4.1 (Effective dimension (Zhang, 2005; Hastie et al., 2009; Calandriello et al., 2017)[1]**).** The effective dimension of $x_{1:n}$ w.r.t. $\mathcal{K}$ at scale λ is defined to be

$$d_{\text{eff}}(\lambda, x_{1:n}) = \operatorname{tr}\left((\mathfrak{K} + \lambda I)^{-1} \mathfrak{K}\right).$$

We also define $d_{\text{eff}}(\lambda, \emptyset) = 0$.

Remark 5.4.2. It is easy to see that $d_{\text{eff}}(\lambda, x_{1:n})$ is a non-negative non-increasing function of λ, it tends to 0 as $\lambda \to \infty$. It is also a bounded function of λ in that $d_{\text{eff}}(\lambda, x_{1:n}) \leq n$.

The following lemma gives a basis-independent representation of d_{eff}; in particular, it shows that the effective dimension is invariant to the permutation of data, therefore $x_{1:n}$ in the definition of d_{eff} can be simply treated as a set.

[1] It is also called effective degrees of freedom in Hastie et al. (2009).

Lemma 5.4.3 ((Zhang, 2005))**.**

$$d_{\text{eff}}(\lambda, x_{1:n}) = \operatorname{tr}\left((\Sigma + \lambda I)^{-1}\Sigma\right).$$

The following lemma gives yet another representation of d_{eff}, which will be particularly useful when analyzing our proposed algorithm later.

Lemma 5.4.4. $d_{\text{eff}}(\lambda, x_{1:n}) = \sum_{i=1}^{n} \|\mathcal{K}_{x_i}\|^2_{(\Sigma+\lambda I)^{-1}}$.

Proof. Let $W = (\Sigma + \lambda I)^{-1}$, note that

$$\begin{aligned}
\sum_{i=1}^{n} \|\mathcal{K}_{x_i}\|_W^2 &= \sum_{i=1}^{n} \langle W\mathcal{K}_{x_i}, \mathcal{K}_{x_i}\rangle_{\mathcal{H}} \\
&= \sum_{i=1}^{n} \operatorname{tr}\left(W\left(\mathcal{K}_{x_i} \otimes \mathcal{K}_{x_i}\right)\right) \\
&= \operatorname{tr}\left(W \sum_{i=1}^{n} \left(\mathcal{K}_{x_i} \otimes \mathcal{K}_{x_i}\right)\right) \\
&\overset{(a)}{=} d_{\text{eff}}(\lambda, x_{1:n}),
\end{aligned}$$

where (a) is due to Lemma 5.4.3. □

The following lemma shows that the effective dimension is non-decreasing as data accumulate.

Lemma 5.4.5. $d_{\text{eff}}(\lambda, x_{1:(n-1)}) \le d_{\text{eff}}(\lambda, x_{1:n})$.

Proof. Let $W_n = \left(\sum_{i=1}^{n} \mathcal{K}_{x_i} \otimes \mathcal{K}_{x_i} + \lambda I\right)^{-1}$ and $f = W_{n-1}\mathcal{K}_{x_n}$. It is easy to verify that

$$W_n = W_{n-1} - \frac{f \otimes f}{1 + \|\mathcal{K}_{x_n}\|^2_{W_{n-1}}}. \tag{5.1}$$

Therefore,

$$
\begin{aligned}
&d_{\text{eff}}(\lambda, x_{1:n}) - d_{\text{eff}}(\lambda, x_{1:(n-1)}) \\
&\overset{(a)}{=} \sum_{i=1}^{n} \|\mathcal{K}_{x_i}\|_{W_n}^2 - \sum_{i=1}^{n-1} \|\mathcal{K}_{x_i}\|_{W_{n-1}}^2 \\
&\overset{(b)}{=} \|\mathcal{K}_{x_n}\|_{W_{n-1}}^2 - \sum_{i=1}^{n} \frac{\langle \mathcal{K}_{x_i}, f\rangle_{\mathcal{H}}^2}{1 + \|\mathcal{K}_{x_n}\|_{W_{n-1}}^2} \\
&= \frac{\|\mathcal{K}_{x_n}\|_{W_{n-1}}^2}{1 + \|\mathcal{K}_{x_n}\|_{W_{n-1}}^2} - \sum_{i=1}^{n-1} \frac{\langle \mathcal{K}_{x_i}, f\rangle_{\mathcal{H}}^2}{1 + \|\mathcal{K}_{x_n}\|_{W_{n-1}}^2} \\
&= \frac{\|\mathcal{K}_{x_n}\|_{W_{n-1}}^2}{1 + \|\mathcal{K}_{x_n}\|_{W_{n-1}}^2} - \frac{\left\langle \left(\sum_{i=1}^{n-1} \mathcal{K}_{x_i} \otimes \mathcal{K}_{x_i}\right) f, f\right\rangle_{\mathcal{H}}}{1 + \|\mathcal{K}_{x_n}\|_{W_{n-1}}^2} \\
&\geq \frac{\|\mathcal{K}_{x_n}\|_{W_{n-1}}^2}{1 + \|\mathcal{K}_{x_n}\|_{W_{n-1}}^2} - \frac{\|\mathcal{K}_{x_n}\|_{W_{n-1}}^2}{1 + \|\mathcal{K}_{x_n}\|_{W_{n-1}}^2} \\
&= 0,
\end{aligned}
$$

where (a) is due to Lemma 5.4.4 and (b) is due to (5.1). □

It is instrumental to see how d_{eff} behaves under common kernels. For linear kernels, the following lemma is well-known.

Lemma 5.4.6 (Effective dimension under linear kernels)**.** *If X is a d-dimensional Euclidean space and $\mathcal{K}(x,y) = x^{\mathsf{T}} y$, then $d_{\text{eff}}(\lambda, x_{1:n}) \leq d$.*

Proof. Let $\Sigma = \sum_{i=1}^{n} x_i x_i^{\mathsf{T}}$. By Lemma 5.4.3 we have that $d_{\text{eff}}(\lambda, x_{1:n}) = \operatorname{tr}\left((\Sigma + \lambda I)^{-1}\Sigma\right)$. Let $\lambda_1, \lambda_2, \cdots, \lambda_d$ be the eigenvalues of Σ, we have $\operatorname{tr}\left((\Sigma + \lambda I)^{-1}\Sigma\right) = \sum_{i=1}^{n} \frac{\lambda_i}{\lambda_i + \lambda} \leq d$. □

For Gaussian RBF kernels, we have the following recent result. Notably, the effective dimension grows only polylogarithmically with $\frac{n}{\lambda}$.

Lemma 5.4.7 (Effective dimension under Gaussian RBF kernels (Altschuler et al., 2018))**.** *Let d be a positive integer, $X = \left\{x \in \mathbb{R}^d : \|x\|_2 \leq 1\right\}$, and $\mathcal{K}(x,y) = e^{-\eta \|x-y\|_2^2}$ for some $\eta \geq 0$, then* *for any $\lambda \leq n$,*

$$d_{\text{eff}}(\lambda, x_{1:n}) \leq 3\left(6 + \frac{41}{d}\eta + \frac{3}{d}\ln\frac{n}{\lambda}\right)^{d}.$$

Pseudo Dimension

Definition 5.4.8 (Pseudo dimension[2] (Jézéquel et al., 2019))**.** The pseudo dimension of $x_{1:n}$ w.r.t. $\mathcal{K}$ at scale λ is defined to be

$$d_{\text{pse}}(\lambda, x_{1:n}) = \ln\left(\det\left(I + \frac{\mathfrak{K}}{\lambda}\right)\right).$$

We also define $d_{\text{pse}}(\lambda, \emptyset) = 0$.

Remark 5.4.9. Similar to d_{eff}, $d_{\text{pse}}(\lambda, x_{1:n})$ is also a non-negative non-increasing function of λ, it tends to 0 as $\lambda \to \infty$. However, in contrast to d_{eff}, $d_{\text{pse}}(\lambda, x_{1:n})$ is not a bounded function of λ; if $\mathfrak{K}$ is not a zero matrix, as $\lambda \to 0$, it tends to infinity.

Remark 5.4.10. Because the determinant only changes sign when swapping rows or columns, the pseudo dimension is invariant to the permutation of data, therefore just like in the case of d_{eff}, $x_{1:n}$ in the above definition can be simply treated as a set.

The following lemma shows that d_{pse} is at most a logarithmic factor (in terms of $\frac{n}{\lambda}$) larger than d_{eff}.

Lemma 5.4.11 ((Jézéquel et al., 2019))**.**

$$d_{\text{pse}}(\lambda, x_{1:n}) \leq \ln\frac{e(n+\lambda)}{\lambda} \cdot d_{\text{eff}}(\lambda, x_{1:n}).$$

The following lemma will play an important role in our analysis.

Lemma 5.4.12. *The following are true:*

[2] This quantity was not given a name in the original paper, we name it in light of Definition 5.4.1 and Lemma 5.4.11 to facilitate the discussion.

(I). $d_{\text{pse}}(\lambda, x_{1:(n-1)}) \leq d_{\text{pse}}(\lambda, x_{1:n})$.

(II). *Let* $\Sigma_i = \sum_{j=1}^{i} \mathcal{K}_{x_j} \otimes \mathcal{K}_{x_j}$*, then for any* $\gamma \in [0,1)$*,*

$$\begin{aligned}&\sum_{i=1}^{n} \sum_{\tau=1}^{i} \gamma^{i-\tau} \|\mathcal{K}_{x_i}\|^2_{(\Sigma_{\tau-1}+\lambda I)^{-1}} \\ &\leq \frac{1/\lambda}{\ln(1+1/\lambda)(1-\gamma)^2} \cdot d_{\text{pse}}(\lambda, x_{1:n}).\end{aligned}$$

Proof. First note that since Σ_{i-1}/λ is a linear operator of rank at most $i-1$,

$$\det(\Sigma_{i-1}/\lambda + I)$$

is well-defined. Therefore,

$$\begin{aligned}&\|\mathcal{K}_{x_i}\|^2_{(\Sigma_{i-1}+\lambda I)^{-1}} \\ &= \|\mathcal{K}_{x_i}/\lambda\|^2_{(\Sigma_{i-1}/\lambda+I)^{-1}} \\ &\overset{(a)}{=} \frac{e^{d_{\text{pse}}(\lambda, x_{1:i})}}{e^{d_{\text{pse}}(\lambda, x_{1:(i-1)})}} - 1,\end{aligned} \tag{5.2}$$

where (a) can be proved for example using the same argument used for Cesa-Bianchi and Lugosi (2006, Lemma 11.11). Note that (5.2) already implies (I), since a norm is non-negative. Next, it is easy to verify that $\|\mathcal{K}_{x_i}\|^2_{(\Sigma_{i-1}+\lambda I)^{-1}} \leq 1/\lambda$ and that $x \leq \frac{b}{\ln(1+b)} \ln(1+x)$ for any $x \in [0,b]$. Therefore,

$$\begin{aligned}\|\mathcal{K}_{x_i}\|^2_{(\Sigma_{i-1}+\lambda I)^{-1}} &\leq \frac{1/\lambda}{\ln(1+1/\lambda)} \cdot \ln\left(1 + \|\mathcal{K}_{x_i}\|^2_{(\Sigma_{i-1}+\lambda I)^{-1}}\right) \\ &\overset{(a)}{=} \frac{1/\lambda}{\ln(1+1/\lambda)} \cdot \left(d_{\text{pse}}(\lambda, x_{1:i}) - d_{\text{pse}}(\lambda, x_{1:(i-1)})\right),\end{aligned}$$

where (a) is due to (5.2). Therefore (I) is proved, since a norm cannot be negative. Note that the above argument is still true if we replace x_i by an arbitrary $x \in X$. Consequently, denote by $[a,b]$

the concatenation of sequence a and sequence b, we have

$$\begin{aligned}
&\sum_{i=1}^{n}\sum_{\tau=1}^{i}\gamma^{i-\tau}\|\mathcal{K}_{x_i}\|^2_{(\Sigma_{\tau-1}+\lambda I)^{-1}}\\
&=\frac{2/\lambda}{\ln(1+1/\lambda)}\left(\sum_{i=1}^{n}\sum_{\tau=1}^{i}\gamma^{i-\tau}d_{\mathrm{pse}}(\lambda,[x_{1:(\tau-1)},x_i])-\sum_{i=1}^{n}\sum_{\tau=1}^{i}\gamma^{i-\tau}d_{\mathrm{pse}}(\lambda,x_{1:(\tau-1)})\right)\\
&\overset{(a)}{\leq}\frac{2/\lambda}{\ln(1+1/\lambda)}\left(\underbrace{\sum_{i=1}^{n}\sum_{\tau=1}^{i}\gamma^{i-\tau}d_{\mathrm{pse}}(\lambda,x_{1:i})}_{(A)}-\underbrace{\sum_{i=1}^{n}\sum_{\tau=1}^{i}\gamma^{i-\tau}d_{\mathrm{pse}}(\lambda,x_{1:(\tau-1)})}_{(B)}\right)\\
&\overset{(b)}{\leq}\frac{2/\lambda}{\ln(1+1/\lambda)}\left(\sum_{i=1}^{n}\sum_{\tau=1}^{2i-n}\gamma^{i-\tau}d_{\mathrm{pse}}(\lambda,x_{1:i})\right)\\
&\overset{(c)}{\leq}\frac{2/\lambda}{\ln(1+1/\lambda)}\left(\sum_{i=1}^{n}\gamma^{n-i}\cdot\frac{d_{\mathrm{pse}}(\lambda,x_{1:n})}{1-\gamma}\right)\\
&\leq\frac{2/\lambda}{\ln(1+1/\lambda)(1-\gamma)^2}\cdot d_{\mathrm{pse}}(\lambda,x_{1:n}),
\end{aligned}$$

where we used (I) in (a), (b), (c) and the nonnegativity of d_{pse} in (b); in particular, in (b) we canceled the (i,τ)th term in (A) with the $(2i+1-\tau,i+1)$th term in (B), where the first index of a term is its position in the first summation and the second index is its position in the second summation. This concludes the proof of (II). □

In particular, (I) in Lemma 5.4.12 says that, similar to the effective dimension (Lemma 5.4.5), the pseudo dimension is also non-decreasing as data accumulate.

To conclude the introduction of pseudo dimension, let us introduce a generalization of Abbasi-Yadkori et al. (2011, Theorem 1) stated in terms of d_{pse} in the following Lemma. Its proof is almost identical to the proof in the special case when $\mathcal{H}$ has a linear kernel. We also note that the inequality in the lemma visually resembles Theorem 1 in Chowdhury and Gopalan (2017), however, the quantities on the left hand side of the inequalities are actually quite different.

Lemma 5.4.13 (Self-normalized bound for $\mathcal{H}$-valued martingales)**.** *Let $\{F_t\}_{t=0}^{\infty}$ be a filtration, $\{\eta_t\}_{t=1}^{\infty}$ be a real-valued stochastic process such that η_t is F_t measurable, and is zero mean and*

R-sub-Gaussian conditioned on F_{t-1}. Let $\{x_t\}_{t=1}^{\infty}$ be a X-valued stochastic process such that x_t is F_{t-1} measurable. Then for any $p > 0$, with probability at least $1-p$, for all $T \geq 0$,

$$\left\| \sum_{t=1}^{T} \eta_t \mathcal{K}_{x_t} \right\|^2_{\left(\lambda I + \sum_{\tau=1}^{T} \mathcal{K}_{x_\tau} \otimes \mathcal{K}_{x_\tau}\right)^{-1}} \\ \leq 2R^2 \left(d_{\mathrm{pse}}(\lambda, x_{1:T}) + \ln\left(\frac{1}{p}\right)\right).$$

5.4.2 Covering Number of Operators

Let $T : X \to Y$ be a bounded linear operator where X and Y are normed vector spaces. $\mathcal{N}(\varepsilon, T)$, the ε-covering number of T, is defined to be the cardinality of the smallest set $V \subseteq Y$ such that for any $x \in X$, $\|x\| \leq 1$, there exists a $v \in V$ such that $\|v - Tx\| \leq \varepsilon$. If there is no such set V of finite cardinality, then $N(\varepsilon, T)$ is defined to be ∞.

Given a real-valued RKHS $\mathcal{H}$, we are interested in the covering numbers of two identity mappings,

$$I_{\mathcal{H},*} : (\mathcal{H}, \|\cdot\|_{\mathcal{H}}) \to (\mathcal{H}, \|\cdot\|_{*}),$$
$$I_{\mathrm{HS},*} : (\mathcal{B}_{\mathrm{HS}}(\mathcal{H}), \|\cdot\|_{\mathrm{HS}}) \to (\mathcal{B}_{\mathrm{HS}}(\mathcal{H}), \|\cdot\|_{*}).$$

Both $\mathcal{N}(\varepsilon, I_{\mathcal{H},*})$ and $\mathcal{N}(\varepsilon, I_{\mathrm{HS},*})$ captures properties $\mathcal{H}$. However, unlike the effective dimension introduced in Section 5.4.1 and the pseudo dimension introduced in Section 5.4.1 that depend on a scale λ, these covering numbers are data-independent and depend on a granularity parameter ε instead.

Lemma 5.4.14 ($\mathcal{N}(\varepsilon, I_{\mathcal{H},*})$ and $\mathcal{N}(\varepsilon, I_{\mathrm{HS},*})$ under linear kernel)**.** *If*

$$X = \left\{ x \in \mathbb{R}^d : \|x\|_2 \leq 1 \right\}$$

and

$$\mathcal{K}(x,y) = x^{\mathsf{T}} y,$$

then

$$\ln \mathcal{N}(\varepsilon, I_{\mathcal{H},*}) \leq d \ln(1 + 2/\varepsilon)$$
$$\ln \mathcal{N}(\varepsilon, I_{\mathrm{HS},*}) \leq d^2 \ln(1 + 2/\varepsilon).$$

The proof of the above lemma is included in Appendix A.1.

Lemma 5.4.15 ($\mathcal{N}(\varepsilon, I_{\mathcal{H},*})$ under Gaussian RBF kernel (Kühn, 2011))**.** *If*

$$X = \left\{ x \in \mathbb{R}^d : \|x\|_2 \leq 1 \right\}$$

and

$$\mathcal{K}(x,y) = e^{-\eta \|x-y\|_2^2}$$

for some $\eta \geq 0$*, then*

$$\ln \mathcal{N}(\varepsilon, I_{\mathcal{H},*}) \leq \left\lceil 2 \left(\ln \frac{2}{\varepsilon} + e^2 \eta \right) \right\rceil^d \ln \left(1 + \frac{4}{\varepsilon} \right).$$

Lemma 5.4.16 ($\mathcal{N}(\varepsilon, I_{\mathrm{HS},*})$ under Gaussian RBF kernel)**.** *If*

$$X = \left\{ x \in \mathbb{R}^d : \|x\|_2 \leq 1 \right\}$$

and

$$\mathcal{K}(x,y) = e^{-\eta \|x-y\|_2^2}$$

for some $\eta \geq 0$*, then*

$$\ln \mathcal{N}(\varepsilon, I_{\mathrm{HS},*}) \leq \left\lceil 2\left(\ln \frac{2\sqrt{2}}{\varepsilon} + e^2\eta\right)\right\rceil^{2d} \ln\left(1+\frac{4}{\varepsilon}\right).$$

The proof of the above lemma is included in Appendix A.2.

5.4.3 The General Setting

Now we are ready to state a rather general upper bound on the γ-regret for KQL. In fact, we have the following theorem.

Theorem 5.4.17. *Given $\lambda > 0$, $\gamma \in [0,1)$, $\rho \geq 1-\gamma$, $\varepsilon \geq 0$, $\sigma \in [0,1]$, $p > 0$, $d_\lambda \geq 1$, $c_\lambda \geq 0$, let*

$$\beta = \frac{1}{1-\gamma} \cdot \min\left(2\sqrt{T+\lambda},\ 3\rho\sqrt{\lambda} + \varepsilon\sqrt{Td_\lambda} + 2\sigma\sqrt{d_\lambda \ln\frac{e(T+\lambda)}{\lambda} + \ln\frac{2}{p} + c_\lambda}\right),$$

in Algorithm 3, or any algorithm that satisfies (1)-(3) in Fact 5.3.1 and acts to maximize $\widetilde{Q}_t$ therein, if

$$\begin{aligned} & d_\lambda \geq d_{\mathrm{eff}}\left(\lambda, \{(s_t, a_t)\}_{i=1}^T\right), \\ & c_\lambda \geq \ln\mathcal{N}\left(\frac{\lambda^2(1-\gamma)}{4T^2}, I_{\mathcal{H},*}\right) + \ln\mathcal{N}\left(\frac{\lambda^3(1-\gamma)}{32(T+\lambda)^3}, I_{\mathrm{HS},*}\right), \end{aligned} \tag{5.3}$$

then with probability at least $1-p$,

$$\begin{aligned} & \mathrm{Regret}_\gamma(T) = \\ & \mathcal{O}\left(\sqrt{\frac{Td_\lambda \log\frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)(1-\gamma)^5}} \cdot \left(\rho + \varepsilon\sqrt{\frac{d_\lambda T}{\lambda}} + \sigma\sqrt{\frac{d_\lambda \log\frac{e(T+\lambda)}{\lambda p} + c_\lambda}{\lambda}}\right)\right). \end{aligned}$$

Let us take a closer look at how the bound depends on ρ, ε, σ, and λ. Note that the bound can be decomposed into three parts, involving ρ, ε, σ respectively:

The first part, involving the complexity bound ρ, is

$$\mathcal{O}\left(\frac{\rho\sqrt{T}}{(1-\gamma)^{2.5}} \cdot \sqrt{\frac{d_\lambda \log\frac{e(K+\lambda)}{\lambda}}{\log(1+1/\lambda)}}\right).$$

Here d_λ upper bounds a non-increasing function of λ that is bounded by T and $\frac{\log\frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)}$ is a non-decreasing function of λ that tends to 1 as $\lambda \to 0$ and tends to ∞ as $\lambda \to \infty$. This suggests that λ needs to strive a balance between the two conflicting dependencies.

The second part, involving the approximation bound ε, is

$$\mathcal{O}\left(\frac{\varepsilon T}{(1-\gamma)^{2.5}} \cdot d_\lambda \sqrt{\frac{\log \frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)\lambda}}\right).$$

Here d_λ upper bounds a non-increasing function of λ, $\frac{\log \frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)\lambda}$ is a strictly decreasing function of λ, and tends to 1 as $\lambda \to \infty$. This suggests that as far as the ε-related part is concerned, the larger the λ the better.

The third part, involving the stochasticity σ, is

$$\mathcal{O}\left(\frac{\sigma\sqrt{T}}{(1-\gamma)^{2.5}} \sqrt{\frac{d_\lambda \log \frac{e(T+\lambda)}{\lambda}\left(d_\lambda \log \frac{e(T+\lambda)}{\lambda p} + c_\lambda\right)}{\log(1+1/\lambda)\lambda}}\right).$$

Here both d_λ and c_λ upper bound a non-increasing function of λ. $\frac{\log \frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)\lambda}$, as discussed before, is a strictly decreasing function of λ that tends to 1 as $\lambda \to \infty$. $\log \frac{e(T+\lambda)}{\lambda p}$ is a strictly decreasing function of λ that tends to $\log \frac{e}{p}$ as $\lambda \to \infty$. This suggests that as far as the σ-related part is concerned, the larger the λ the better.

To conclude, both the ε- and σ- related parts of the regret bound prefers larger λ; however, the ρ-related part in general calls for a λ that is neither too large nor too small.

Having made these observations, let us proceed to prove the theorem.

Proof of Theorem 5.4.17

We first introduce some notations. For any $t \in [T]$, let $\mathrm{prev}(t) = [t-1]$. Let

$$\begin{aligned}
\widetilde{Q}_t(s,a) &= \operatorname*{clip}_{[0,1/(1-\gamma)]}\left(\widehat{Q}_t(s,a) + \beta \left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_t}\right), \\
\widetilde{V}_t(s) &= \max_a \widetilde{Q}_t(s,a).
\end{aligned}$$

Define

$$\Phi = \frac{1}{1-\gamma} \cdot \left(3\rho\sqrt{\lambda} + \varepsilon\sqrt{Td_\lambda}\right),$$
$$\Psi_p = \frac{2\sigma}{1-\gamma} \cdot \sqrt{d_\lambda \ln \frac{e(T+\lambda)}{\lambda} + \ln \frac{1}{p} + c_\lambda}\ .$$

For any $p > 0$, let $\mathfrak{E}_p$ be the event that for any $s \in \mathcal{S}$, $a \in \mathcal{A}$, $t \in [T]$,

$$\left|\widehat{Q}_t(s,a) - Q_*(s,a) - \gamma\left(\left(M_*\left(\widetilde{V}_{t-1} - V_*\right)\right)(s,a)\right)\right|$$
$$\leq (\Phi + \Psi_p) \cdot \left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_t} + \frac{2\varepsilon}{1-\gamma},$$

where $\widetilde{V}_0$ can be any function that is bounded in $[0, 1/(1-\gamma)]$. We start from some auxiliary lemmas.

Lemma 5.4.18. *For any $t \in [T]$ and $f \in \mathcal{H}$, $\frac{\|f\|_{\mathcal{H}}}{\sqrt{\lambda+T}} \leq \|f\|_{\widehat{W}_t} \leq \frac{\|f\|_{\mathcal{H}}}{\sqrt{\lambda}}$.*

Proof. Let $\Sigma = \sum_{\tau \in \mathrm{prev}(t)} \mathcal{K}_{(s_\tau, a_\tau)} \otimes \mathcal{K}_{(s_\tau, a_\tau)}$ and $\lambda_1, \lambda_2, \cdots, \lambda_{|\mathrm{prev}(t)|} \geq 0$ be the eigenvalues of Σ. Note that the eigenvalues of $\widehat{W}_t = (\Sigma + \lambda I)^{-1}$ are $\frac{1}{\lambda_1+\lambda}, \frac{1}{\lambda_2+\lambda}, \cdots, \frac{1}{\lambda_{|\mathrm{prev}(t)|}+\lambda}$. The upper bound follows immediately. To get the upper bound, note that for each λ_i, there exists $f \in \mathcal{H}$, $\|f\|_{\mathcal{H}} = 1$, such that

$$\begin{aligned} \lambda_i &= \|\Sigma f\|_{\mathcal{H}} \\ &\overset{(a)}{\leq} \sum_{\tau \in \mathrm{prev}(t)} \|f\|_{\mathcal{H}} \|\mathcal{K}_{(s_\tau,a_\tau)}\|_{\mathcal{H}}^2 \\ &\overset{(b)}{\leq} T, \end{aligned}$$

where (a) is due to Cauchy-Schwarz and (b) is because by our assumption $\|\mathcal{K}_{(s_\tau,a_\tau)}\|_{\mathcal{H}} \leq 1$. The lower bound then follows. □

Because by our assumptions $\mathcal{K}$ is normalized such that for any $(s,a) \in \mathcal{S} \times \mathcal{A}$,

$$\left\|\mathcal{K}_{(s,a)}\right\|_{\mathcal{H}} \geq \frac{1}{2},$$

we have the following corollary of Lemma 5.4.18.

Corollary 5.4.19. *For any $t \in [T]$, $(s,a) \in \mathcal{S} \times \mathcal{A}$, $\left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_t} \geq \frac{1}{2\sqrt{T+\lambda}}$.*

Lemma 5.4.20. *For any $t \in [T]$, $\left\|\widehat{W}_t\right\|_{\mathrm{HS}} \leq \frac{\sqrt{T}}{\lambda}$.*

Proof. Let $\Sigma = \sum_{\tau \in \mathrm{prev}(t)} \mathcal{K}_{(s_\tau,a_\tau)} \otimes \mathcal{K}_{(s_\tau,a_\tau)}$ and $\lambda_1, \lambda_2, \cdots, \lambda_{|\mathrm{prev}(t)|} \geq 0$ be the eigenvalues of Σ. Note that the eigenvalues of $\widehat{W}_t = (\Sigma + \lambda I)^{-1}$ are $\frac{1}{\lambda_1+\lambda}, \frac{1}{\lambda_2+\lambda}, \cdots, \frac{1}{\lambda_{|\mathrm{prev}(t)|}+\lambda}$, and

$$\begin{aligned}
\left\|\widehat{W}_t\right\|_{\mathrm{HS}} &= \sqrt{\sum_{\tau \in \mathrm{prev}(t)} \frac{1}{(\lambda_i+\lambda)^2}} \\
&\leq \frac{\sqrt{T}}{\lambda}.
\end{aligned}$$

□

Lemma 5.4.21. *For any $t \in [T]$, $\left\|\widehat{Q}_t\right\|_{\mathcal{H}} \leq \frac{T}{\lambda(1-\gamma)}$.*

Proof. For any $f \in \mathcal{H}$, we have that

$$\begin{aligned}
\left|\widehat{Q}_t f\right| &= \left|\left(\sum_{\tau \in \mathrm{prev}(t)} \left(r_h(s_\tau, r_\tau) + \widetilde{V}_{t-1}(s_{\tau+1})\right) \cdot \mathcal{K}_{(s_\tau,a_\tau)}\right) \widehat{W}_t f\right| \\
&\leq \frac{1}{1-\gamma} \cdot \sum_{\tau \in \mathrm{prev}(t)} \left|\mathcal{K}_{(s_\tau,a_\tau)} \widehat{W}_t f\right| \\
&\overset{(a)}{\leq} \frac{1}{1-\gamma} \cdot \sum_{\tau \in \mathrm{prev}(t)} \|f\|_{\widehat{W}_t} \cdot \left\|\mathcal{K}_{(s_\tau,a_\tau)}\right\|_{\widehat{W}_t} \\
&\overset{(b)}{\leq} \frac{1}{1-\gamma} \cdot \sqrt{\sum_{\tau \in \mathrm{prev}(t)} \|f\|_{\widehat{W}_t}^2} \cdot \sqrt{\sum_{\tau \in \mathrm{prev}(t)} \left\|\mathcal{K}_{(s_\tau,a_\tau)}\right\|_{\widehat{W}_t}^2} \\
&\overset{(c)}{\leq} \frac{T}{\lambda(1-\gamma)} \cdot \|f\|_{\mathcal{H}},
\end{aligned}$$

where (a) and (b) are due to Cauchy-Schwarz, and (c) is due to the upper bound in Lemma 5.4.18. □

Lemma 5.4.22. *For any $p > 0$, if*

$$\beta = \min\left(\Phi + \Psi_p, \frac{2\sqrt{T+\lambda}}{1-\gamma}\right)$$

and $\mathfrak{E}_p$ happens, we have for any $t \in [T]$, $s \in \mathcal{S}$, $a \in \mathcal{A}$,

$$\begin{aligned}
&Q_*(s,a) - \frac{2\varepsilon}{1-\gamma} \cdot \sum_{\tau=1}^{t} \gamma^{t-\tau} \\
&\leq \widetilde{Q}_t(s,a) \\
&\leq Q_*(s,a) + 2\beta \sum_{\tau=1}^{t} \gamma^{t-\tau} \|\mathcal{K}_{(s,a)}\|_{\widehat{W}_\tau} + \frac{2\varepsilon}{1-\gamma} \cdot \sum_{\tau=1}^{t} \gamma^{t-\tau}.
\end{aligned}$$

Proof. We prove the lemma by induction. When $t = 1$, since $\widetilde{V}_0$ can be chosen as either 0 or $1/(1-\gamma)$, by the definition of $\mathfrak{E}_p$ and $\widetilde{Q}_t$, the choice of β, and Corollary 5.4.19, the inequalities hold. Now the inequality holds for $t = t' - 1$ where $1 < t' \leq T$, we have again by the definition of $\mathfrak{E}_p$, the choice of β, and Corollary 5.4.19,

$$\begin{aligned}
&\widehat{Q}_{t'}(s,a) + (\Phi + \Psi_p)\left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_{t'}} \\
&\in Q_*(s,a) + \gamma\left(M_*\left(\widetilde{V}_{t'-1} - V_*\right)\right)(s,a) + \left[-\frac{2\varepsilon}{1-\gamma}, 2\beta\left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_{t'}} + \frac{2\varepsilon}{1-\gamma}\right].
\end{aligned} \tag{5.4}$$

Note that on one hand

$$\begin{aligned}
&\left(M_*\left(\widetilde{V}_{t'-1}-V_*\right)\right)(s,a)\\
&=\mathbb{E}_{s'\sim\mathbb{P}_{(s,a)}}\left[\widetilde{V}_{t'-1}(s')-V_*(s')\right]\\
&\geq\mathbb{E}_{s'\sim\mathbb{P}(s,a)}\left[\widetilde{Q}_{t'-1}(s',a^*(s'))-Q_*(s',a^*(s'))\right]\\
&\overset{(a)}{\geq}-\frac{2\varepsilon}{1-\gamma}\cdot\sum_{\tau=1}^{t'-1}\gamma^{t'-1-\tau},
\end{aligned}$$

where (a) is by induction hypothesis. On the other hand, for any $s'\in\mathcal{S}$, let $\tilde{a}(s')$ be an arbitray element in $\arg\max\widetilde{Q}_{t'-1}(s',a)$, we have

$$\begin{aligned}
&\left(M_*\left(\widetilde{V}_{t'-1}-V_*\right)\right)(s,a)\\
&=\mathbb{E}_{s'\sim\mathbb{P}_{(s,a)}}\left[\widetilde{V}_{t'-1}(s')-V_*(s')\right]\\
&\leq\mathbb{E}_{s'\sim\mathbb{P}(s,a)}\left[\widetilde{Q}_{t'-1}(s',\tilde{a}(s'))-Q_*(s',\tilde{a}(s'))\right]\\
&\overset{(a)}{\leq}2\beta\sum_{\tau=1}^{t'-1}\gamma^{t'-1-\tau}\left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_\tau}+\frac{2\varepsilon}{1-\gamma}\cdot\sum_{\tau=1}^{t'-1}\gamma^{t'-1-\tau}.
\end{aligned}$$

Going back to (5.4) we arrive at

$$\begin{aligned}
&\widehat{Q}_{t'}(s,a)+(\Phi+\Psi_p)\left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_{t'}}\\
&\in Q_*(s,a)+\left[-\frac{2\varepsilon}{1-\gamma}\cdot\sum_{\tau=1}^{t'-\tau}\gamma^{\tau},\ 2\beta\sum_{\tau=1}^{t'}\gamma^{t'-\tau}\left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_\tau}+\frac{2\varepsilon}{1-\gamma}\cdot\sum_{\tau=1}^{t'}\gamma^{t'-\tau}\right].
\end{aligned}$$

Using the choice of β, Corollary 5.4.19, and the definition of $\widetilde{Q}_t$, we see that the inequality holds for $t=t'$. This concludes the proof. □

Lemma 5.4.23. *Let $\mathcal{Q}$ be the function class containing all functions*

$$\begin{aligned}
&\mathcal{S}\times\mathcal{A}\to\mathbb{R}\\
&(s,a)\mapsto f(s,a)+\|\mathcal{K}_{(s,a)}\|_W,
\end{aligned}$$

where $f\in\mathcal{H}$ is such that $\|f\|_{\mathcal{H}}\leq\frac{T}{\lambda(1-\gamma)}$ and $W\in\mathcal{B}_{\mathrm{HS}}(\mathcal{H})$ is such that $\|W\|_{\mathrm{HS}}\leq\frac{2(T+\lambda)}{\lambda(1-\gamma)}$. Then for any $p>0$, with probability at least $1-p$, for any $t\in[T]$, and $q\in\mathcal{Q}$,

$$\left\|\sum_{\tau\in\mathrm{prev}(t)}\left(v_q(s_{\tau+1})-\left(M_*v_q\right)(s_\tau,a_\tau)\right)\cdot\mathcal{K}_{(s_\tau,a_\tau)}\right\|_{\widehat{W}_t}\leq\sqrt{\lambda}+\Psi_p,$$

where $v_q(s)=\max_a\mathrm{clip}_{[0,1/(1-\gamma)]}\,q(s,a)$.

Proof. First let us assume we have a $\xi>0$ and a function class $\tilde{\mathcal{Q}}:\mathcal{S}\times\mathcal{A}\to\mathbb{R}$ of finite cardinality such that for any $q\in\mathcal{Q}$ there exists a $\gamma(q)\in\tilde{\mathcal{Q}}$ such that $\|q-\gamma(q)\|_\infty\leq\xi$. Then with probability at least $1-p$, for any $t\in[T]$, and $q\in\mathcal{Q}$,

$$\begin{aligned}
&\left\|\sum_{\tau\in\mathrm{prev}(t)}\left(v_q(s_{\tau+1})-\left(M_*v_q\right)(s_{\tau,h},a_{\tau,h})\right)\cdot\mathcal{K}_{(s_\tau,a_\tau)}\right\|_{\widehat{W}_t}\\
&\leq\left\|\sum_{\tau\in\mathrm{prev}(t)}\left(v_{\gamma(q)}(s_{\tau+1})-\left(M_*v_{\gamma(q)}\right)(s_{\tau,h},a_{\tau,h})\right)\cdot\mathcal{K}_{(s_\tau,a_\tau)}\right\|_{\widehat{W}_t}+\left\|\sum_{\tau\in\mathrm{prev}(t)}2\xi\cdot\mathcal{K}_{(s_\tau,a_\tau)}\right\|_{\widehat{W}_t}\\
&\overset{(a)}{\leq}\left\|\sum_{\tau\in\mathrm{prev}(t)}\left(v_{\gamma(q)}(s_{\tau+1})-\left(M_*v_{\gamma(q)}\right)(s_\tau,a_\tau)\right)\cdot\mathcal{K}_{(s_\tau,a_\tau)}\right\|_{\widehat{W}_t}+\frac{1}{\sqrt{\lambda}}\left\|\sum_{\tau\in\mathrm{prev}(t)}2\xi\cdot\mathcal{K}_{(s_\tau,a_\tau)}\right\|_{\mathcal{H}}\\
&\overset{(b)}{\leq}\frac{\sqrt{2}\sigma}{1-\gamma}\sqrt{d_\lambda\ln\frac{e(T+\lambda)}{\lambda}+\ln\left|\widetilde{V}\right|+\ln\frac{1}{p}}+\frac{2\xi T}{\sqrt{\lambda}},
\end{aligned}$$

where in (a) we used the upper bound in Lemma 5.4.18, in (b) we used Lemma 5.4.13, Lemma 5.4.11, and a union bound.

It remains to choose $\widetilde{V}$ and ξ. Note that for any q_1 (induced by f_1, W_1) and q_2 (induced by f_2, W_2) in $\mathcal{Q}$,

$$\|q_1 - q_2\|_\infty \leq \|f_1 - f_2\|_* + \sqrt{\|W_1 - W_2\|_*},$$

so it suffices to bound $\|f_1 - f_2\|_*$ by $\xi/2$ and bound $\|W_1 - W_2\|_*$ by $\xi^2/4$, therefore we can choose $\widetilde{V}$ such that

$$\ln\left|\widetilde{V}\right| \leq \ln\mathcal{N}\left(\frac{\xi\lambda(1-\gamma)}{2T}, I_{\mathcal{H},*}\right) + \ln\mathcal{N}\left(\frac{\xi^2\lambda(1-\gamma)}{8(T+\lambda)}, I_{\mathrm{HS},*}\right),$$

choosing $\xi = \frac{\lambda}{2T}$ concludes the proof. □

Lemma 5.4.24. *For any $p > 0$, $\mathfrak{E}_p$ happens with probability at least $1 - p$.*

Proof. First note that for any $t \in [T]$, $Q_* = r_* + \gamma M_* V_*$, and consequently

$$\|Q_* - (r + \gamma M V_*)\|_\infty \leq \frac{\varepsilon}{1-\gamma}.$$

To proceed, note that for any $t \in [T]$,

$$\begin{aligned}
&\widehat{Q}_t - (r + \gamma M V_*) \\
&= \left(\sum_{\tau \in \mathrm{prev}(t)} \left(r(s_\tau, a_\tau) + \gamma \widetilde{V}_{t-1}(s_{\tau+1}) \right) \cdot \mathcal{K}_{(s_\tau, a_\tau)} - (r + \gamma M V_*) \left(\widehat{W}_t \right)^{-1} \right) \widehat{W}_t \\
&= \underbrace{-\lambda \left(r + \gamma M V_* \right) \widehat{W}_t}_{E_1} \\
&+ \underbrace{\gamma \left(\sum_{\tau \in \mathrm{prev}(t)} \left(\widetilde{V}_{t-1}(s_{\tau+1}) - \left(M_* \widetilde{V}_{t-1} \right) (s_\tau, a_\tau) \right) \cdot \mathcal{K}_{(s_\tau, a_\tau)} \right) \widehat{W}_t}_{E_2} \\
&+ \underbrace{\gamma \left(\sum_{\tau \in \mathrm{prev}(t)} \left(M \left(\widetilde{V}_{t-1} - V_* \right) \right) (s_\tau, a_\tau) \cdot \mathcal{K}_{(s_\tau, a_\tau)} \right) \widehat{W}_t}_{E_3} \\
&+ \underbrace{\gamma \left(\sum_{\tau \in \mathrm{prev}(t)} \left((M_* - M) \left(\widetilde{V}_{t-1} - V_* \right) \right) (s_\tau, a_\tau) \cdot \mathcal{K}_{(s_\tau, a_\tau)} \right) \widehat{W}_t}_{E_4}.
\end{aligned}$$

Let us bound $E_1(s,a)$, $E_2(s,a)$, $E_3(s,a)$, $E_4(s,a)$ separately. First note that

$$\begin{aligned}
|E_1(s,a)| &\leq \lambda \left| r \widehat{W}_t \mathcal{K}_{(s,a)} \right| + \lambda \left| (\gamma M V_*) \widehat{W}_t \mathcal{K}_{(s,a)} \right| \\
&\overset{(a)}{\leq} \lambda \cdot \left(\|r\|_{\widehat{W}_t} + \|\gamma M V_*\|_{\widehat{W}_t} \right) \cdot \left\| \mathcal{K}_{(s,a)} \right\|_{\widehat{W}_t} \\
&\overset{(b)}{\leq} \sqrt{\lambda} \cdot \left(\|r\|_{\mathcal{H}} + \|\gamma M V_*\|_{\mathcal{H}} \right) \cdot \left\| \mathcal{K}_{(s,a)} \right\|_{\widehat{W}_t} \\
&\leq \sqrt{\lambda} \left(\|r_h\|_{\mathcal{H}} + \frac{\gamma}{1-\gamma} \|M\| \right) \cdot \left\| \mathcal{K}_{(s,a)} \right\|_{\widehat{W}_t} \\
&\leq \frac{\sqrt{\lambda} \rho}{1-\gamma} \cdot \left\| \mathcal{K}_{(s,a)} \right\|_{\widehat{W}_t},
\end{aligned}$$

where (a) is due to Cauchy-Schwarz and (b) is due to the upper bound in Lemma 5.4.18.

To bound E_2, recall that for any $t \geq 2$

$$\widetilde{V}_{t-1}(s,a) = \max_{a} \operatorname*{clip}_{[0,1/(1-\gamma)]} \left(\widehat{Q}_{t+1}(s,a) + \left\| \mathcal{K}_{(s,a)} \right\|_{\beta \widehat{W}_{t+1}} \right),$$

and by Lemma 5.4.21 $\widehat{Q}_{t+1}(s,a) \leq \frac{T}{\lambda(1-\gamma)}$, and by Lemma 5.4.20 as well as the constraint on β, $\|\beta \widehat{W}_{t+1}\|_{\mathrm{HS}} \leq \frac{2(T+\lambda)}{\lambda(1-\gamma)}$. Therefore by Lemma 5.4.23 we have that with probability at least $1-p$, uniformly for all $t \in [T]$,

$$\begin{aligned}
|E_2(s,a)| &= \left| \left(\sum_{\tau \in \mathrm{prev}(t)} \left(\widetilde{V}_{t-1}(s_{\tau+1}) - \left(M_* \widetilde{V}_{t-1} \right)(s_\tau, a_\tau) \right) \cdot \mathcal{K}_{(s_\tau,a_\tau)} \right) \widehat{W}_t \mathcal{K}_{(s,a)} \right| \\
&\leq \left\| \sum_{\tau \in \mathrm{prev}(t)} \left(\widetilde{V}_{t-1}(s_{\tau+1}) - \left(M_* \widetilde{V}_{t-1} \right)(s_\tau, a_\tau) \right) \cdot \mathcal{K}_{(s_\tau,a_\tau)} \right\|_{\widehat{W}_t} \cdot \left\| \mathcal{K}_{(s,a)} \right\|_{\widehat{W}_t} \\
&\leq \left(\sqrt{\lambda} + \Psi_p \right) \cdot \left\| \mathcal{K}_{(s,a)} \right\|_{\widehat{W}_t}.
\end{aligned}$$

Also note that

$$\begin{aligned}
E_3(s,a) &= \left(\sum_{\tau \in \mathrm{prev}(t)} \left\langle M\left(\widetilde{V}_{t-1} - V_* \right), \mathcal{K}_{(s_\tau,a_\tau)} \right\rangle_{\mathcal{H}} \cdot \mathcal{K}_{(s_\tau,a_\tau)} \right) \widehat{W}_t \mathcal{K}_{(s,a)} \\
&= \left(\left(M\left(\widetilde{V}_{t-1} - V_* \right) \right) \left(\sum_{\tau \in \mathrm{prev}(t)} \mathcal{K}_{(s_\tau,a_\tau)} \otimes \mathcal{K}_{(s_\tau,a_\tau)} \right) \right) \widehat{W}_t \mathcal{K}_{(s,a)} \\
&= \left(M\left(\widetilde{V}_{t-1} - V_* \right) \right) \mathcal{K}_{(s,a)} - \lambda \left(M\left(\widetilde{V}_{t-1} - V_* \right) \right) \widehat{W}_t \mathcal{K}_{(s,a)}
\end{aligned}$$

Therefore, we have that

$$\begin{aligned}
&\left|E_3(s,a) - \left(M\left(\widetilde{V}_{t-1} - V_*\right)\right)(s,a)\right| \\
&= \lambda \left|\left(M\left(\widetilde{V}_{t-1} - V_*\right)\right)\widehat{W}_t \mathcal{K}_{(s,a)}\right| \\
&\overset{(a)}{\leq} \lambda \cdot \left\|M\left(\widetilde{V}_{t-1} - V_*\right)\right\|_{\widehat{W}_t} \cdot \left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_t} \\
&\overset{(b)}{\leq} \sqrt{\lambda} \cdot \left\|M\left(\widetilde{V}_{t-1} - V_*\right)\right\|_{\mathcal{H}} \cdot \left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_t} \\
&\leq \frac{\sqrt{\lambda}}{1-\gamma} \cdot \|M\| \cdot \left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_t} \\
&\leq \frac{\sqrt{\lambda}\rho}{1-\gamma} \cdot \left\|\mathcal{K}_{(s,a)}\right\|_{\widehat{W}_t}
\end{aligned}$$

where (a) is due to Cauchy-Scharz and (b) is due to the upper bound in Lemma 5.4.18. We also have

$$\left|\left((M - M_*)\left(\widetilde{V}_{t-1} - V_*\right)\right)(s,a)\right| \leq \frac{\varepsilon}{1-\gamma}.$$

Finally, note that

$$\begin{aligned}
|E_4(s,a)| &\leq \frac{\varepsilon}{1-\gamma} \sum_{\tau \in \mathrm{prev}(t)} \left|\mathcal{K}_{(s_\tau,a_\tau)} \widehat{W}_t \mathcal{K}_{s,a}\right| \\
&\overset{(a)}{\leq} \frac{\varepsilon}{1-\gamma} \sqrt{\sum_{\tau \in \mathrm{prev}(t)} \left\|\mathcal{K}_{(s_\tau,a_\tau)}\right\|_{\widehat{W}_t}^2 \cdot \sum_{\tau \in \mathrm{prev}(t)} \left\|\mathcal{K}_{s,a}\right\|_{\widehat{W}_t}^2} \\
&\overset{(b)}{\leq} \frac{\varepsilon\sqrt{Td_\lambda}}{1-\gamma} \cdot \left\|\mathcal{K}_{s,a}\right\|_{\widehat{W}_t}
\end{aligned}$$

where (a) is due to Cauchy-Schwarz and (b) is due to Lemma 5.4.4 and Lemma 5.4.5. Putting everything together concludes the proof. □

Now we are ready to prove our main theorem. The proof will be conditioned on $\mathfrak{E}_{p/2}$, which happens with probability at least $p/2$ by Lemma 5.4.24. Define

$$\begin{aligned}
\delta_t &= \widetilde{V}_t(s_t) - V_t, \\
\phi_t &= \widetilde{V}_t(s_t) - V_*(s_t), \\
\zeta_t &= \left(M_*(V_*)\right)(s_t, a_t) - V_*(s_{t+1}).
\end{aligned}$$

Note that

$$\phi_t \geq \widetilde{Q}_t(s_t, a^*(s_t)) - Q_*(s_t, a^*(s_t)) \overset{(a)}{\geq} -\frac{2\varepsilon}{(1-\gamma)^2}, \tag{5.5}$$

where (a) is due to Lemma 5.4.22. We have for any $t \in [T]$,

$$\begin{aligned}
\delta_t &= \widetilde{Q}_t(s_t, a_t) - V_t \\
&= \Big(\widetilde{Q}_t(s_t, a_t) - Q_*(s_t, a_t)\Big) + \left(Q_*(s_t, a_t) - V_t\right) \\
&= \Big(\widetilde{Q}_t(s_t, a_t) - Q_*(s_t, a_t)\Big) + \gamma\zeta_t + \gamma\delta_{t+1} - \gamma\phi_{t+1} \\
&\overset{(a)}{\leq} 2\beta \sum_{\tau=1}^{t} \gamma^{t-\tau} \left\|\mathcal{K}_{(s_t,a_t)}\right\|_{\widehat{W}_\tau} + \frac{2\varepsilon}{(1-\gamma)^2} + \gamma\zeta_t + \gamma\delta_{t+1} - \gamma\phi_{t+1}
\end{aligned}$$

where (a) is due to the upper bound in Lemma 5.4.22. Taking a summation on both sides, we have,

$$\begin{aligned}
\sum_{t=1}^{T} \delta_t &\leq \frac{2\varepsilon T}{(1-\gamma)^2} + 2\beta \sum_{t=1}^{T} \sum_{\tau=1}^{t} \gamma^{t-\tau} \left\|\mathcal{K}_{(s_t,a_t)}\right\|_{\widehat{W}_\tau} + \gamma \sum_{t=1}^{T} \left(\zeta_t + \delta_{t+1} - \phi_{t+1}\right) \\
&\overset{(a)}{\leq} \frac{2\varepsilon T}{(1-\gamma)^2} + 2\beta \sqrt{\sum_{t=1}^{T} \sum_{\tau=1}^{t} \gamma^{t-\tau}} \sqrt{\sum_{t=1}^{T} \sum_{\tau=1}^{t} \gamma^{t-\tau} \left\|\mathcal{K}_{(s_t,a_t)}\right\|_{\widehat{W}_\tau}^2} + \gamma \sum_{t=1}^{T} \left(\zeta_t + \delta_{t+1} - \phi_{t+1}\right) \\
&\overset{(b)}{\leq} \frac{4\varepsilon T}{(1-\gamma)^2} + 2\beta \sqrt{\frac{T d_\lambda/\lambda \cdot \ln \frac{e(T+\lambda)}{\lambda}}{\ln(1+1/\lambda)(1-\gamma)^3}} + \gamma \sum_{t=1}^{T} \left(\zeta_t + \delta_{t+1}\right),
\end{aligned}$$

where (a) is by Cauchy-Schwarz and (b) is due to Lemma 5.4.12, Lemma 5.4.11, and (5.5). After rearranging the terms and using the fact $|\delta_t| \leq 1/(1-\gamma)$, we have

$$\sum_{t=1}^{T} \delta_t \leq \frac{4\varepsilon T}{(1-\gamma)^3} + 2\beta \sqrt{\frac{Td_\lambda/\lambda \cdot \ln \frac{e(T+\lambda)}{\lambda}}{\ln(1+1/\lambda)\,(1-\gamma)^5}} + \gamma \sum_{t=1}^{T} \zeta_t + \frac{2}{(1-\gamma)^2}.$$

Now note that $\{\zeta_t\}_{t=1}^{T}$ is a Martingale difference sequence where each element is $(\frac{\sigma}{1-\gamma})$-sub-Gaussian, therefore with probability at least $1-\delta/2$,

$$\sum_{t=1}^{T} \delta_t \leq \frac{4\varepsilon T}{(1-\gamma)^3} + 2\beta \sqrt{\frac{Td_\lambda/\lambda \cdot \ln \frac{e(T+\lambda)}{\lambda}}{\ln(1+1/\lambda)\,(1-\gamma)^5}} + \frac{\sigma\sqrt{2T\ln(2/p)}}{1-\gamma} + \frac{2}{(1-\gamma)^2}.$$

Finally, note that

$$\begin{aligned}
&\text{Regret}_\gamma(T) \\
&\overset{(a)}{\leq} (1-\gamma)\sum_{t=1}^{T} \Delta_t + \frac{1}{1-\gamma} \\
&\overset{(b)}{\leq} \frac{1}{1-\gamma} \cdot \sum_{t=1}^{T} \delta_t + \frac{2\varepsilon T + 1}{1-\gamma} \\
&\leq \frac{6\varepsilon T}{(1-\gamma)^2} + 2\beta \sqrt{\frac{Td_\lambda/\lambda \cdot \ln \frac{e(T+\lambda)}{\lambda}}{\ln(1+1/\lambda)\,(1-\gamma)^3}} + \frac{\sigma\sqrt{2T\ln(2/p)}}{1-\gamma} + \frac{3}{1-\gamma} \\
&\leq \left(6\rho\sqrt{\lambda} + 2\varepsilon\sqrt{Td_\lambda} + 4\sigma\sqrt{d_\lambda \ln \frac{e(T+\lambda)}{\lambda} + \ln\frac{2}{p} + c_\lambda}\right) \cdot \\
&\quad \sqrt{\frac{Td_\lambda/\lambda \cdot \ln \frac{e(T+\lambda)}{\lambda}}{\ln(1+1/\lambda)\,(1-\gamma)^5}} + \frac{6\varepsilon T}{(1-\gamma)^2} + \frac{\sigma\sqrt{2T\ln(2/p)}}{1-\gamma} + \frac{3}{1-\gamma} \\
&= \mathcal{O}\left(\sqrt{\frac{Td_\lambda \log \frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)(1-\gamma)^5}}\left(\rho + \varepsilon\sqrt{\frac{d_\lambda T}{\lambda}} + \sigma\sqrt{\frac{d_\lambda \log \frac{e(T+\lambda)}{\lambda p} + c_\lambda}{\lambda}}\right)\right) + \\
&\quad \mathcal{O}\left(\frac{\sigma\sqrt{T\log(1/p)}}{1-\gamma} + \frac{\varepsilon T}{(1-\gamma)^2} + \frac{1}{1-\gamma}\right) \\
&\overset{(c)}{=} \mathcal{O}\left(\sqrt{\frac{Td_\lambda \log \frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)(1-\gamma)^5}}\left(\rho + \varepsilon\sqrt{\frac{d_\lambda T}{\lambda}} + \sigma\sqrt{\frac{d_\lambda \log \frac{e(T+\lambda)}{\lambda p} + c_\lambda}{\lambda}}\right)\right),
\end{aligned}$$

where in (a) we used Lemma 2.3.1, in (b) we used $\Delta_t = \delta_t - \phi_t$ and (5.5), in (c) we used the fact that $d_\lambda \geq 1$, $\rho \geq 1-\gamma$, and

$$\frac{\log \frac{e(K+\lambda)}{\lambda}}{\log(1+1/\lambda)} \geq 1,$$

$$\frac{\log \frac{e(K+\lambda)}{\lambda}}{\log(1+1/\lambda)\sqrt{\lambda}} \geq 1.$$

This concludes the proof.

5.4.4 Specific Settings

The following corollary is a direct consequence of Theorem 5.4.17, Lemma 5.4.6, Lemma 5.4.14.

Corollary 5.4.25 (Regret for linear kernels)**.** *In Theorem 5.4.17, if*

$$\mathcal{K}((s_1,a_1),(s_2,a_2)) = \phi(s_1,a_1)^\intercal \phi(s_2,a_2)$$

where $\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ *for some positive integer* d *such that*

$$\|\phi(s,a)\|_2 \le 1$$

for any $(s,a) \in \mathcal{S} \times \mathcal{A}$*. Then we can choose* $d_\lambda = d$ *and some*

$$c_\lambda = \mathcal{O}\left(d^2 \log \frac{eH(T+\lambda)}{\lambda}\right)$$

such that (5.3) *is always true, and therefore with probability at least* $1-p$*,*

$$\mathrm{Regret}_\gamma(T) = \mathcal{O}\left(\sqrt{\frac{Td\log\frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)(1-\gamma)^5}} \cdot \left(\rho + \varepsilon\sqrt{\frac{dT}{\lambda}} + \sigma d \sqrt{\frac{\log\frac{eH(T+\lambda)}{\lambda p}}{\lambda}}\right)\right).$$

Remark 5.4.26. Recall that in the general setting λ should be neither too large nor too small. However, since here d_λ is bounded by d, which is independent of λ, if λ is sufficiently small such that $\lambda = \mathcal{O}(1/T)$, we can make the dependency on ρ to be $\frac{\rho\sqrt{Td}}{(1-\gamma)^{2.5}}$. In other words, in the linear setting, if the MDP is deterministic and the dynamics can be exactly represented by functions in $\mathcal{H}$, then we can choose $\lambda = \mathcal{O}(1/T)$ such that

$$\mathrm{Regret}_\gamma(T) = \mathcal{O}\left(\frac{\rho\sqrt{Td}}{(1-\gamma)^{2.5}}\right).$$

Comparing this with choosing $\lambda = \Theta(1)$, a factor of $\sqrt{\log(eT)}$ is reduced.

A perhaps more interesting example would be the regret bound for the widely-used Gaussian RBF kernel. The following corollary is a direct consequence of Theorem 5.4.17, Lemma 5.4.7, Lemma 5.4.15, Lemma 5.4.16.

Corollary 5.4.27 (Regret for Gaussian RBF kernels)**.** *In Theorem 5.4.17, if*

$$\mathcal{K}((s_1,a_1),(s_2,a_2)) = e^{-\eta\|\phi(s_1,a_1)-\phi(s_2,a_2)\|_2^2}$$

for some $\eta \geq 0$ and $\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^n$ for some positive integer n such that

$$\|\phi(s,a)\|_2 \leq 1$$

for any $(s,a) \in \mathcal{S} \times \mathcal{A}$. Then we can choose some

$$d_\lambda = \left(\mathcal{O}\left(\eta + \log\frac{e(T+\lambda)}{\lambda(1-\gamma)}\right)\right)^n, \quad c_\lambda = \mathcal{O}\left(d_\lambda^2 \log\frac{e(T+\lambda)}{\lambda(1-\gamma)}\right),$$

such that (5.3) *is always true, and therefore with probability at least $1-p$,*

$$\text{Regret}_\gamma(T) = \mathcal{O}\left(\sqrt{\frac{Td_\lambda \log\frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)(1-\gamma)^5}} \cdot \left(\rho + \varepsilon\sqrt{\frac{d_\lambda T}{\lambda}} + \sigma d_\lambda \sqrt{\frac{\log\frac{eH(T+\lambda)}{\lambda p}}{\lambda}}\right)\right).$$

We see that the above regret bound only differs from the bound in Corollary 5.4.25 in that d is replaced by $\mathcal{O}\left(\eta + \log\frac{e(T+\lambda)}{\lambda(1-\gamma)}\right)$. This suggests that this new quantity indicates the effective dimensionality under the Gaussian RBF kernel at scale λ. Note that although a smaller η would benefit the regret bound, an η too small would violate the assumption on ρ.

Remark 5.4.28. Again we can inspect how the ρ-related term in the bound,

$$\mathcal{O}\left(\frac{\rho\sqrt{T}}{(1-\gamma)^{2.5}} \cdot \sqrt{\frac{d_\lambda \log\frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)}}\right)$$

depends on λ. Consider choosing $\lambda = \Theta(1/T)$, we have

$$\frac{d_\lambda \log \frac{e(T+\lambda)}{\lambda}}{\log(1+1/\lambda)} = \mathcal{O}(d_\lambda) = \left(\mathcal{O}\left(\eta + \log \frac{eT}{1-\gamma}\right)\right)^n.$$

In other words, in the Gaussian RBF kernel setting, if the MDP is deterministic and the dynamics can be exactly represented by functions in $\mathcal{H}$, then we can choose $\lambda = \Theta(1/T)$ such that

$$\mathrm{Regret}_\gamma(T) = \frac{\rho\sqrt{T}}{(1-\gamma)^{2.5}} \cdot \left(\mathcal{O}\left(\eta + \log \frac{eT}{1-\gamma}\right)\right)^{n/2}.$$

Comparing this with choosing $\lambda = \Theta(1)$, a factor of $\sqrt{\log(eT)}$ is reduced. Recall that in Remark 5.4.26, we showed that the same thing happens in the linear setting.

5.5 Experiments

We test KQL on a suite of classic control tasks included in OpenAI Gym (Brockman et al., 2016): MOUNTAINCAR (Figure 5.1), PENDULUM (Figure 5.2), ACROBOT (Figure 5.3), and CARTPOLE (Figure 5.4). The action space of PENDULUM is discretized to $\{-1,0,1\}$, all other environments have discrete action space natively.

5.5.1 Methodology

For any state s and action a, let $[s,a]$ be the concatenation of the state vector and the one-hot embedding of the action; let l be the length of state vectors. The states are first normalized such that $||s||_\infty \leq 1$ for all s. We experiment with two types of kernels:

- Linear Kernel

$$\mathcal{K}((s_1,a_1),(s_2,a_2)) = \frac{[s_1,a_1]^\intercal[s_2,a_2]}{2l} + \frac{1}{2}.$$

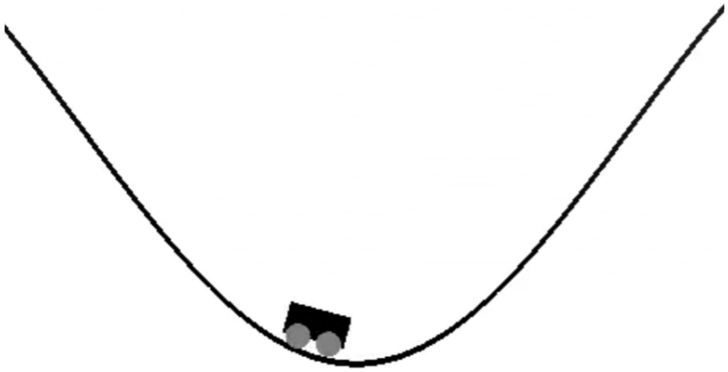

Figure 5.1. The MOUNTAINCAR environment: the algorithm needs to swing the car back and forth to gain momentum in order to reach the flag.

Figure 5.2. The PENDULUM environment: the algorithm needs to swing the rod upwards and maintain it in the upright position.

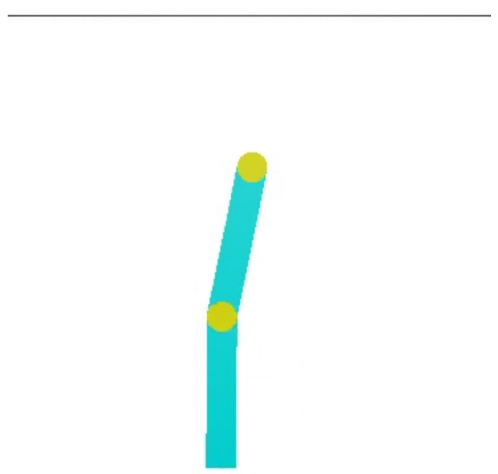

Figure 5.3. The ACROBOT environment: the algorithm needs to swing the two rods that are connected by a joint upwards above the line.

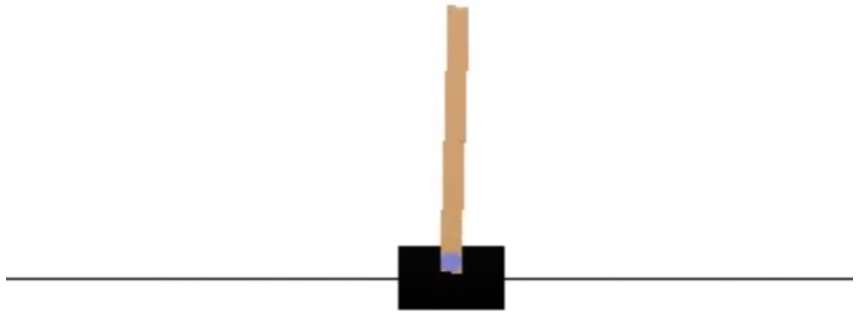

Figure 5.4. The CARTPOLE environment: the algorithm needs to maintain the rod in an upright position.

- Gaussian RBF Kernel

$$\mathcal{K}((s_1,a_1),(s_2,a_2)) = e^{-\eta\|[s_1,a_1]-[s_2,a_2]\|_2^2}.$$

where η is empirically set to 0.02 for MOUNTAINCAR, CARTPOLE and ACROBOT, and 1 for PENDULUM.

We normalize the rewards to fall within $[0,1]$. Since all environments are deterministic we can let $\sigma = 0$. We also believe that for these simple environments, if a correct kernel is used, ε can be made infinitesimally small. Our theoretical results (Theorem 5.4.17) suggest that in this case we can set

$$\beta = \frac{3\rho\sqrt{\lambda}}{1-\gamma}.$$

and our discussions in Remark 5.4.26 and Remark 5.4.28 suggest we can set

$$\lambda = \mathcal{O}\left(\frac{1}{T}\right).$$

We heuristically set $\beta = \frac{\sqrt{\lambda}}{1-\gamma}$ so that $\widetilde{Q}_t \gtrsim 1/(1-\gamma)$ at the beginning of learning, and set $\lambda = \frac{1}{10T}$ so that λ is large enough to not cause numerical issues.

We compare KQL with Deep Q-Learning (DQN) (Mnih et al., 2013), a practical, widely-used, neural network based algorithm known for its superior sample efficiency. We use the default implementation provided in Stable-Baselines3 (Raffin et al., 2021), with the environment-specific parameter overrides from RL Baselines3 Zoo (Raffin, 2020), except the following changes to accommodate to the extreme small amount of samples: batch_size=64, learning_starts=100, target_update_interval=10, train_freq=1, gradient_steps=32.

To emphasize the test on sample efficiency, for each environment, all algorithms are only allowed 1000 steps through the OpenAI Gym interface; a *reset* is only allowed in the very

Table 5.1. Evaluating different versions of Q-learning on classic control environments. Each model is trained for 1000 steps and evaluated over 100 episodes after training. The numbers are presented in the format MEAN ± STD. Larger is better.

	MOUNTAINCAR	PENDULUM	ACROBOT	CARTPOLE
DQN	-200.0 ± 0.0	-274.6 ± 393.0	-161.3 ± 96.1	149.6 ± 4.7
Linear	-200.0 ± 0.0	-1413.5 ± 202.1	-500.0 ± 0.0	9.3 ± 0.7
Gaussian RBF	$\mathbf{-132.5 \pm 17.0}$	$\mathbf{-167.0 \pm 97.3}$	$\mathbf{-107.2 \pm 35.2}$	$\mathbf{200.0 \pm 0.0}$

beginning and after a *done* is received. We use the same discounting factor $\gamma = 0.95$ for all algorithms and all environments.

5.5.2 Results

The results are shown in Table 5.1. As we can see, the linear kernel performs poorly on all tasks. In fact, its performance is not noticeably better than a random policy. Without handcrafted or learned feature embedding, the value function is highly non-linear; thus linear kernel suffers from huge approximation error. DQN makes progress on most tasks, with the exception of MOUNTAINCAR, which requires more than random exploration in order to succeed with a very limited amount of interaction. Remarkably, the Gaussian RBF kernel achieves high return and performs best on all tasks, compensating for its long running time by its far superior sample efficiency.

5.6 Computational Considerations

We have demonstrated that KQL is both theoretically sound and empirically promising. However, kernel methods are known to suffer from *the curse of kernelization* — their time complexity has a super-linear dependency on the number of samples, KQL is no exception.

There are two standard ways to trade sample efficiency for time complexity in kernel-based online learning: one is sparsification, which aims at keeping the number of support vectors $d \ll T$ by various strategies (see e.g. Engel et al. (2004); Sun et al. (2012); Calandriello et al. (2017) and the discussion in Lu et al. (2016, Bibliographic Notes)); the other is kernel approximation, which aims at projecting vectors in an RKHS into a Euclidean space, for example, using random Fourier features, and reduce the kernel setting to the linear setting (Lu et al., 2016; Jézéquel et al., 2019). Given its already impressive performance even without any kernel sparsification or approximation, it is not hard to believe that with proper acceleration, KQL is able to tackle more challenging tasks such as Atari games (Bellemare et al., 2013). We leave it as future work to investigate these directions.

5.7 Bibliographic Notes

Algorithms and corresponding regret analyses have previously been derived for general function classes (see e.g. Jin et al. (2020); Yang et al. (2020); Zhou et al. (2021b), among others). However, the proposed algorithms use either episodic value iteration (Jin et al., 2020; Yang et al., 2020), which is not sample-efficient in practice, or extended value iteration Zhou et al. (2021b), which is not time-efficient in practice.

There was previous work that generalizes tabular analyses to the linear model episodic setting (Wang et al., 2019; Jin et al., 2020; Zanette et al., 2020), the linear mixture model episodic setting (Cai et al., 2020; Ayoub et al., 2020; Modi et al., 2020; Yang and Wang, 2020; Zhou et al., 2021a; He et al., 2021a), and the linear mixture model discounted setting (Zhou et al., 2021b). We note here that the linear mixture model assumes the model class has finite (and small) degrees of freedom, which mitigates many challenges in model estimation (see discussion in Jin et al. (2020, Bibliographic Notes)). In particular, while Zhou et al. (2021b) gives a seemingly better regret bound than ours when the model class is linear in certain sense, they are under far stricter assumptions.

Our KQL and its analysis are most related to Yang et al. (2020), which proposes and analyzes KOVI, a kernelized version of episodic value iteration. The major difference is that Yang et al. (2020) operates on an episodic setting, where in each episode, an optimistically optimal

Q-value function is calculated based on all the historical data using episodic value iteration. On the other hand, we focus on the non-episodic setting, where in each step, a new Q-value function, which is not necessarily (optimistically) optimal based on current historical data, is calculated from the Q-value function from the previous step using one-step discounting. Such difference indeed affects theoretical analysis in that the non-episodic update introduces pathological dependencies that must be properly handled (e.g., Lemma 5.4.12). Furthermore, it also makes our algorithm significantly more applicable to real-world problems. For example, in our MOUNTAINCAR experiment (Section 5.5), the length of an episode is 200. Given a budget of 1000 interactions, KOVI can only update its value functions $1000/200-1=4$ times (since each step in an episode uses a different value function). It is very unlikely that any progress can be made after such a small number of parameter updates (indeed, we were not able to make KOVI work on MOUNTAIN-CAR after extensive parameter tuning or even hacking). On the other hand, KQL does not distinguish between different episodes, and can update its value function 999 times. In fact, we will show that KQL even outperforms DQN in the low-budget setting.

Other kernelization efforts include Xu et al. (2005), which uses kernel regression to approximate the discounted return of a Markov chain, Xu et al. (2007), which uses kernel regression as a subroutine in a policy iteration procedure, no exploration is involved thus no regret bound can be obtained; Chowdhury and Gopalan (2019), which requires very restrictive assumptions, as discussed in Yang and Wang (2020); Yang and Wang (2020), which assumes linear mixture models, therefore suffers from aforementioned limitations of such models; Ormoneit and Sen (2002); Barreto et al. (2016), which are based on local averaging and require Lipschitzness assumptions, and no exploration is involved thus no regret bound can be obtained; Domingues et al. (2021), which is similarly based on local averaging and require Lipschitzness assumption, but with Lipschitzness-based exploration.

More general function approximation classes have also been considered. Notably, Wang et al. (2020); Ayoub et al. (2020) gave regret bounds in terms of Eluder dimension and covering number of the function class. However, most of these attempts either result in generally intractable algorithms (Krishnamurthy et al., 2016; Jiang et al., 2017; Dann et al., 2018; Dong et al., 2020; Wang et al., 2020; Ayoub et al., 2020), or make very restrictive assumptions such as deterministic environment (Wen and Van Roy, 2013; 2017), or the existence of a finite latent state space (Du et al., 2019).

This chapter is primarily based on material in Liu and Su (2022), of which the dissertation author was the primary researcher and author.

Appendix A

Proofs for Covering Numbers

A.1 Proof of Lemma 5.4.14

In the case of linear kernel, for any $x \in \mathcal{H}$, $\|x\|_{\mathcal{H}} = \|x\|_* = \|x\|_2$; and for any linear operator T over X, $\|T\|_* \leq \|T\|_{\mathrm{F}} = \|T\|_{\mathrm{HS}}$. Because a n-dimensional unit ball can be ε-covered by $(1+2/\varepsilon)^n$ points, the lemma follows immediately.

A.2 Proof of Lemma 5.4.16

Let $\mathbb{N}$ be the set of non-negative integers. For any multi-index $\nu = (n_1, n_2, \cdots, n_d) \in \mathbb{N}^d$, define $|\nu| = \sum_{i=1}^{n} n_i$. The following facts will be useful.

Fact A.2.1 (Multinomial expansion)**.** *For any nonnegative integer n,*

$$\sum_{\substack{\nu \in \mathbb{N}^d \\ |\nu| = n}} \prod_{j=1}^{d} \frac{(2\eta x_j^2)^{n_j}}{n_j!} = \frac{1}{n!} \left(2\eta \|x\|_2^2\right)^n.$$

Fact A.2.2 (Taylor expansion)**.** *For any nonnegative integer n and $t \geq 0$,*

$$\sum_{n=N}^{\infty} \frac{t^n}{n!} \leq \frac{t^N}{N!} e^t.$$

Define the function

$$\begin{aligned} e_\nu : X &\to \mathbb{R} \\ x &\mapsto \prod_{j=1}^{d} e_{n_j}(x_j) \end{aligned}$$

where

$$e_{n_j}(x_j) = \sqrt{\frac{(2\eta)^{n_j}}{n_j!}} x_j^{n_j} e^{-\eta x_j^2}.$$

It is well-known that $B = \left\{e_\nu : \nu \in \mathbb{N}^d\right\}$ is an orthonormal basis for $\mathcal{H}$ (see e.g. Steinwart et al. (2006, Theorem 3.7)). Consequently, $\left\{e_{\nu_1} \otimes e_{\nu_2} : \nu_1, \nu_2 \in \mathbb{N}^d\right\}$ is an orthonormal (Schauder) basis for $\mathcal{B}_{\mathrm{HS}}(\mathcal{H})$. Now note that

$$\begin{aligned} \left\|I_{\mathrm{HS},*}\right\|^2 &= \sup_{\substack{T \in \mathcal{B}_{\mathrm{HS}}(\mathcal{H}) \\ \|T\|_{\mathrm{HS}} \le 1}} \sup_{x \in X} \langle T\mathcal{K}_x, \mathcal{K}_x \rangle_{\mathcal{H}}^2 \\ &= \sup_{x \in X} \sup_{\substack{T \in \mathcal{B}_{\mathrm{HS}}(\mathcal{H}) \\ \|T\|_{\mathrm{HS}} \le 1}} \left(\sum_{\nu_1, \nu_2 \in \mathbb{N}^d} \langle T, e_{\nu_1} \otimes e_{\nu_2} \rangle_{\mathrm{HS}} \cdot e_{\nu_1}(x) \cdot e_{\nu_2}(x) \right)^2 \\ &\overset{(a)}{=} \sup_{x \in X} \sum_{\nu_1, \nu_2 \in \mathbb{N}^d} e_{\nu_1}^2(x) \cdot e_{\nu_2}^2(x) \\ &= \sup_{x \in X} \left(\sum_{\nu \in B} e_\nu^2(x) \right)^2 \\ &= \sup_{x \in X} \left(\sum_{\nu \in \mathbb{N}^d} \prod_{j=1}^{d} \frac{(2\eta x_j^2)^{n_j}}{n_j!} \cdot e^{-2\eta \|x\|_2^2} \right)^2 \\ &\overset{(b)}{=} \sup_{x \in X} \left(\sum_{n=0}^{\infty} \frac{1}{n!} \left(2\eta \|x\|_2^2\right)^n \cdot e^{-2\eta \|x\|_2^2} \right)^2 \\ &= 1, \end{aligned} \tag{A.1}$$

where (a) is due to Cauchy-Schwarz and (b) is due to Fact A.2.1.

Next we introduce for all positive integer N two orthogonal projections: P_N, which is the projection onto span $\{e_{\nu_1} \otimes e_{\nu_2} : |\nu_1| < N, |\nu_2| < N\}$, and Q_N, which is the projection onto $P_N^{\perp}$. Note that

$$\begin{aligned}
\left\|I_{\mathrm{HS},*}Q_N\right\|^2 &= \sup_{\substack{T\in\mathcal{B}_{\mathrm{HS}}(\mathcal{H})\\ \|T\|_{\mathrm{HS}}\leq 1}} \sup_{x\in X} \langle Q_N T \mathcal{K}_x, \mathcal{K}_x\rangle_{\mathcal{H}}^2 \\
&= \sup_{x\in X} \sup_{\substack{T\in\mathcal{B}_{\mathrm{HS}}(\mathcal{H})\\ \|T\|_{\mathrm{HS}}\leq 1}} \left(\sum_{\substack{\nu_1,\nu_2\in\mathbb{N}^d\\ \|\nu_1\|\geq N \text{ or } \|\nu_2\|\geq N}} \langle Q_N T, e_{\nu_1}\otimes e_{\nu_2}\rangle_{\mathrm{HS}} \cdot e_{\nu_1}(x)\cdot e_{\nu_2}(x) \right)^2 \\
&\overset{(a)}{=} \sup_{x\in X} \sum_{\substack{\nu_1,\nu_2\in\mathbb{N}^d\\ \|\nu_1\|\geq N \text{ or } \|\nu_2\|\geq N}} e_{\nu_1}^2(x)\cdot e_{\nu_2}^2(x) \\
&\overset{(b)}{=} \sup_{x\in X} \left(\sum_{n=N}^{\infty} \frac{1}{n!}\left(2\eta\|x\|_2^2\right)^n \cdot e^{-2\eta\|x\|_2^2} \cdot \sum_{n=0}^{N-1} \frac{1}{n!}\left(2\eta\|x\|_2^2\right)^n \cdot e^{-2\eta\|x\|_2^2} \right. \\
&\qquad + \sum_{n=0}^{N-1} \frac{1}{n!}\left(2\eta\|x\|_2^2\right)^n \cdot e^{-2\eta\|x\|_2^2} \cdot \sum_{n=N}^{\infty} \frac{1}{n!}\left(2\eta\|x\|_2^2\right)^n \cdot e^{-2\eta\|x\|_2^2} \\
&\qquad \left. + \sum_{n=N}^{\infty} \frac{1}{n!}\left(2\eta\|x\|_2^2\right)^n \cdot e^{-2\eta\|x\|_2^2} \cdot \sum_{n=N}^{\infty} \frac{1}{n!}\left(2\eta\|x\|_2^2\right)^n \cdot e^{-2\eta\|x\|_2^2} \right) \\
&\leq 2\sup_{x\in X} \left(\sum_{n=N}^{\infty} \frac{1}{n!}\left(2\eta\|x\|_2^2\right)^n \cdot e^{-2\eta\|x\|_2^2} \right) \\
&\overset{(c)}{\leq} 2\sup_{x\in X} \left(\frac{(2\eta\|x\|_2^2)^N}{N!} \right) \\
&\leq 2\left(\frac{2e\eta}{N}\right)^N,
\end{aligned}$$

where (a) is due to Cauchy-Schwarz and (b) is due to Fact A.2.1 and (c) is due to Fact A.2.2. Note that we can choose

$$N = \left\lceil 2\left(\ln\frac{2\sqrt{2}}{\varepsilon} + e^2\eta\right)\right\rceil$$

such that $\left\| I_{\mathrm{HS},*} Q_N \right\| \leq \varepsilon/2$. Finally, note that

$$\begin{aligned}
\ln \mathcal{N}(\varepsilon, I_{\mathrm{HS},*}) &\leq \ln \left(\mathcal{N}(\varepsilon/2, I_{\mathrm{HS},*} P_N) \cdot \mathcal{N}(\varepsilon/2, I_{\mathrm{HS},*} Q_N) \right) \\
&= \ln \mathcal{N}(\varepsilon/2, I_{\mathrm{HS},*} P_N) \\
&\overset{(a)}{\leq} \operatorname{rank}(P_N) \ln \left(1 + \frac{4}{\varepsilon} \right) \\
&\leq N^{2d} \ln \left(1 + \frac{4}{\varepsilon} \right) \\
&= \left\lceil 2 \left(\ln \frac{2\sqrt{2}}{\varepsilon} + e^2 \eta \right) \right\rceil^{2d} \ln \left(1 + \frac{4}{\varepsilon} \right),
\end{aligned}$$

where in (a) we used $\| I_{\mathrm{HS},*} P_N \| \leq \| I_{\mathrm{HS},*} \| = 1$ and the fact that a n-dimensional unit ball can be ε-covered by $(1 + 2/\varepsilon)^n$ points.

Bibliography

Y. Abbasi-Yadkori, D. Pál, and C. Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24:2312–2320, 2011.

J. Altschuler, F. Bach, A. Rudi, and J. Niles-Weed. Massively scalable sinkhorn distances via the nyström method. *arXiv preprint arXiv:1812.05189*, 2018.

P. Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

A. Ayoub, Z. Jia, C. Szepesvari, M. Wang, and L. Yang. Model-based reinforcement learning with value-targeted regression. In *International Conference on Machine Learning*, pages 463–474. PMLR, 2020.

M. G. Azar, I. Osband, and R. Munos. Minimax regret bounds for reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 263–272. JMLR. org, 2017.

A. M. Barreto, D. Precup, and J. Pineau. Practical kernel-based reinforcement learning. *The Journal of Machine Learning Research*, 17(1):2372–2441, 2016.

P. L. Bartlett and A. Tewari. Regal: A regularization based algorithm for reinforcement learning in weakly communicating mdps. *arXiv preprint arXiv:1205.2661*, 2012.

M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47: 253–279, 2013.

G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba. Openai gym. *arXiv preprint arXiv:1606.01540*, 2016.

Q. Cai, Z. Yang, C. Jin, and Z. Wang. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pages 1283–1294. PMLR, 2020.

D. Calandriello, A. Lazaric, and M. Valko. Second-order kernel online convex optimization with adaptive sketching. In *International Conference on Machine Learning*, pages 645–653. PMLR, 2017.

N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

S. R. Chowdhury and A. Gopalan. On kernelized multi-armed bandits. In *International Conference on Machine Learning*, pages 844–853. PMLR, 2017.

S. R. Chowdhury and A. Gopalan. Online learning in kernelized markov decision processes. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 3197–3205. PMLR, 2019.

W. Chu, L. Li, L. Reyzin, and R. Schapire. Contextual bandits with linear payoff functions. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, pages 208–214. JMLR Workshop and Conference Proceedings, 2011.

V. Dani, T. P. Hayes, and S. M. Kakade. Stochastic linear optimization under bandit feedback. In *COLT*, 2008.

C. Dann, N. Jiang, A. Krishnamurthy, A. Agarwal, J. Langford, and R. E. Schapire. On oracle-efficient pac rl with rich observations. *arXiv preprint arXiv:1803.00606*, 2018.

O. D. Domingues, P. Ménard, M. Pirotta, E. Kaufmann, and M. Valko. Kernel-based reinforcement learning: A finite-time analysis. In *International Conference on Machine Learning*, pages 2783–2792. PMLR, 2021.

K. Dong, Y. Wang, X. Chen, and L. Wang. Q-learning with ucb exploration is sample efficient for infinite-horizon mdp. *arXiv preprint arXiv:1901.09311*, 2019.

K. Dong, J. Peng, Y. Wang, and Y. Zhou. Root-n-regret for learning in markov decision processes with function approximation and low bellman rank. In *Conference on Learning Theory*, pages 1554–1557. PMLR, 2020.

S. Du, A. Krishnamurthy, N. Jiang, A. Agarwal, M. Dudik, and J. Langford. Provably efficient rl with rich observations via latent state decoding. In *International Conference on Machine Learning*, pages 1665–1674. PMLR, 2019.

Y. Engel, S. Mannor, and R. Meir. The kernel recursive least-squares algorithm. *IEEE Transactions on signal processing*, 52(8):2275–2285, 2004.

R. Fruit, M. Pirotta, A. Lazaric, and R. Ortner. Efficient bias-span-constrained exploration-exploitation in reinforcement learning. In *ICML 2018-The 35th International Conference on Machine Learning*, volume 80, pages 1578–1586, 2018.

H. V. Hasselt. Double q-learning. In *Advances in neural information processing systems*, pages 2613–2621, 2010.

H. V. Hasselt, A. Guez, and D. Silver. Deep reinforcement learning with double q-learning. In *Thirtieth AAAI conference on artificial intelligence*, 2016.

T. Hastie, R. Tibshirani, and J. H. Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction, 2nd Edition*. Springer Series in Statistics. Springer, 2009.

J. He, D. Zhou, and Q. Gu. Nearly minimax optimal reinforcement learning for discounted mdps. *arXiv preprint arXiv:2010.00587*, 2020.

J. He, D. Zhou, and Q. Gu. Logarithmic regret for reinforcement learning with linear function approximation. In *International Conference on Machine Learning*, pages 4171–4180. PMLR, 2021a.

J. He, D. Zhou, and Q. Gu. Nearly minimax optimal reinforcement learning for discounted MDPs. In A. Beygelzimer, Y. Dauphin, P. Liang, and J. W. Vaughan, editors, *Advances in Neural Information Processing Systems*, 2021b.

M. Hessel, J. Modayil, H. V. Hasselt, T. Schaul, G. Ostrovski, W. Dabney, D. Horgan, B. Piot, M. Azar, and D. Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural networks*, 2(5):359–366, 1989.

T. Jaksch, R. Ortner, and P. Auer. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(Apr):1563–1600, 2010.

R. Jézéquel, P. Gaillard, and A. Rudi. Efficient online learning with kernels for adversarial large scale problems. *arXiv preprint arXiv:1902.09917*, 2019.

N. Jiang, A. Krishnamurthy, A. Agarwal, J. Langford, and R. E. Schapire. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*, pages 1704–1713. PMLR, 2017.

C. Jin, Z. Allen-Zhu, S. Bubeck, and M. I. Jordan. Is q-learning provably efficient? In *Advances in Neural Information Processing Systems*, pages 4863–4873, 2018.

C. Jin, Z. Yang, Z. Wang, and M. I. Jordan. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pages 2137–2143, 2020.

S. M. Kakade. *On the sample complexity of reinforcement learning*. PhD thesis, University

College London, 2003.

A. Krause and C. S. Ong. Contextual gaussian process bandit optimization. In *Nips*, pages 2447–2455, 2011.

A. Krishnamurthy, A. Agarwal, and J. Langford. Pac reinforcement learning with rich observations. *arXiv preprint arXiv:1602.02722*, 2016.

T. Kühn. Covering numbers of gaussian reproducing kernel hilbert spaces. *Journal of Complexity*, 27(5):489–499, 2011.

T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

T. Lattimore and M. Hutter. Pac bounds for discounted mdps. In *International Conference on Algorithmic Learning Theory*, pages 320–334. Springer, 2012.

L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

Y. Li, Y. Wang, and Y. Zhou. Nearly minimax-optimal regret for linearly parameterized bandits. In *Conference on Learning Theory*, pages 2173–2174. PMLR, 2019.

Y. Li, Y. Wang, X. Chen, and Y. Zhou. Tight regret bounds for infinite-armed linear contextual bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 370–378. PMLR, 2021.

S. Liu and H. Su. Regret bounds for discounted mdps. *arXiv preprint arXiv:2002.05138*, 2021.

S. Liu and H. Su. Provably efficient kernelized q-learning. *arXiv preprint arXiv:2204.10349*, 2022.

J. Lu, S. C. Hoi, J. Wang, P. Zhao, and Z.-Y. Liu. Large scale online kernel learning. *Journal of Machine Learning Research*, 17(47):1, 2016.

V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra, and M. Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

A. Modi, N. Jiang, A. Tewari, and S. Singh. Sample complexity of reinforcement learning using linearly combined model ensembles. In *International Conference on Artificial Intelligence and Statistics*, pages 2010–2020. PMLR, 2020.

D. Ormoneit and Ś. Sen. Kernel-based reinforcement learning. *Machine learning*, 49(2):161–178, 2002.

R. Ortner. Regret bounds for reinforcement learning via markov chain concentration. *Journal of Artificial Intelligence Research*, 67:115–128, 2020.

I. Osband and B. Van Roy. On lower bounds for regret in reinforcement learning. *arXiv preprint arXiv:1608.02732*, 2016.

I. Osband, B. Van Roy, and Z. Wen. Generalization and exploration via randomized value functions. In *International Conference on Machine Learning*, pages 2377–2386, 2016.

A. Raffin. Rl baselines3 zoo. https://github.com/DLR-RM/rl-baselines3-zoo, 2020.

A. Raffin, A. Hill, A. Gleave, A. Kanervisto, M. Ernestus, and N. Dormann. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 2021.

P. Rusmevichientong and J. N. Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010.

D. Russo. Worst-case regret bounds for exploration via randomized value functions. *arXiv preprint arXiv:1906.02870*, 2019.

M. Simchowitz and K. G. Jamieson. Non-asymptotic gap-dependent regret bounds for tabular mdps. *Advances in Neural Information Processing Systems*, 32:1153–1162, 2019.

N. Srinivas, A. Krause, S. M. Kakade, and M. Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

N. Srinivas, A. Krause, S. M. Kakade, and M. W. Seeger. Information-theoretic regret bounds for gaussian process optimization in the bandit setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, 2012.

B. K. Sriperumbudur, K. Fukumizu, and G. R. Lanckriet. Universality, characteristic kernels and rkhs embedding of measures. *Journal of Machine Learning Research*, 12(7), 2011.

I. Steinwart, D. Hush, and C. Scovel. An explicit description of the reproducing kernel hilbert spaces of gaussian rbf kernels. *IEEE Transactions on Information Theory*, 52(10):4635–4643, 2006.

Y. Sun, F. Gomez, and J. Schmidhuber. On the size of the online kernel sparsification dictionary. *arXiv preprint arXiv:1206.4623*, 2012.

R. S. Sutton, D. McAllester, S. Singh, and Y. Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

I. Szita and C. Szepesvári. Model-based reinforcement learning with nearly tight exploration complexity bounds. In *Proceedings of the Twenty-seventh International Conference on Machine Learning*, pages 1031–1038, 2010.

M. Valko, N. Korda, R. Munos, I. Flaounas, and N. Cristianini. Finite-time analysis of kernelised contextual bandits. *arXiv preprint arXiv:1309.6869*, 2013.

R. Wang, R. Salakhutdinov, and L. F. Yang. Reinforcement learning with general value function approximation: Provably efficient approach via bounded eluder dimension. *arXiv preprint arXiv:2005.10804*, 2020.

Y. Wang, R. Wang, S. S. Du, and A. Krishnamurthy. Optimism in reinforcement learning with generalized linear function approximation. *arXiv preprint arXiv:1912.04136*, 2019.

C. J. Watkins and P. Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

Z. Wen and B. Van Roy. Efficient exploration and value function generalization in deterministic systems. *Advances in Neural Information Processing Systems*, 26:3021–3029, 2013.

Z. Wen and B. Van Roy. Efficient reinforcement learning in deterministic systems with value function generalization. *Mathematics of Operations Research*, 42(3):762–782, 2017.

X. Xu, T. Xie, D. Hu, and X. Lu. Kernel least-squares temporal difference learning. *International Journal of Information Technology*, 11(9):54–63, 2005.

X. Xu, D. Hu, and X. Lu. Kernel-based least squares policy iteration for reinforcement learning. *IEEE transactions on neural networks*, 18(4):973–992, 2007.

L. Yang and M. Wang. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. In *International Conference on Machine Learning*, pages 10746–10756. PMLR, 2020.

Z. Yang, C. Jin, Z. Wang, M. Wang, and M. I. Jordan. On function approximation in reinforcement learning: Optimism in the face of large state spaces. *arXiv preprint arXiv:2011.04622*, 2020.

A. Zanette and E. Brunskill. Tighter problem-dependent regret bounds in reinforcement learning without domain knowledge using value function bounds. In *International Conference on Machine Learning*, pages 7304–7312, 2019.

A. Zanette, D. Brandfonbrener, E. Brunskill, M. Pirotta, and A. Lazaric. Frequentist regret bounds for randomized least-squares value iteration. In *International Conference on Artificial*

Intelligence and Statistics, pages 1954–1964. PMLR, 2020.

T. Zhang. Learning bounds for kernel regression using effective data dimensionality. *Neural Computation*, 17(9):2077–2098, 2005.

Z. Zhang and X. Ji. Regret minimization for reinforcement learning by evaluating the optimal bias function. *arXiv preprint arXiv:1906.05110*, 2019.

Z. Zhang, Y. Zhou, and X. Ji. Almost optimal model-free reinforcement learning via reference-advantage decomposition. *arXiv preprint arXiv:2004.10019*, 2020.

Z. Zhang, X. Ji, and S. Du. Is reinforcement learning more difficult than bandits? a near-optimal algorithm escaping the curse of horizon. In *Conference on Learning Theory*, pages 4528–4531. PMLR, 2021.

D. Zhou, L. Li, and Q. Gu. Neural contextual bandits with ucb-based exploration. In *International Conference on Machine Learning*, pages 11492–11502. PMLR, 2020.

D. Zhou, Q. Gu, and C. Szepesvari. Nearly minimax optimal reinforcement learning for linear mixture markov decision processes. In M. Belkin and S. Kpotufe, editors, *Proceedings of Thirty Fourth Conference on Learning Theory*, volume 134 of *Proceedings of Machine Learning Research*, pages 4532–4576. PMLR, 15–19 Aug 2021a.

D. Zhou, J. He, and Q. Gu. Provably efficient reinforcement learning for discounted mdps with feature mapping. In M. Meila and T. Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 12793–12802. PMLR, 18–24 Jul 2021b.