

POLITECNICO
MILANO 1863

Department of Electronics, Information and Bioengineering
Doctoral Programme In Information Technology

# Theoretical Guarantees for Reinforcement Learning in Continuous State and Action Spaces

Doctoral Dissertation of:
**Davide Maran**

Advisor: Prof. Marcello Restelli
Tutor: Prof. Nicola Gatti
Year 2026 - XXXVIII Cycle

# Abstract

Adapting reinforcement learning to Markov decision processes with continuous state and action spaces is crucial for many real-world applications, particularly in robotics and control. From a theoretical perspective, however, this regime remains poorly understood: while tight guarantees exist for tabular MDPs, a general theory with provably efficient algorithms for the continuous case is still lacking. In this thesis, we introduce a new class of problems, Smooth MDPs, which provides a unifying framework for reinforcement learning in continuous environments. Smooth MDPs are defined through functional regularity assumptions on the Bellman operator and on reward and transition dynamics. This framework rigorously generalizes several models studied in the literature, including continuous bandits, LQRs and their extensions, Lipschitz and kernelized MDPs. Although Sobolev-Hölder-type assumptions are central to nonparametric regression and have recently been explored in bandit problems, prior to this work, a systematic treatment in the context of reinforcement learning was still missing. From a statistical perspective, we characterize the complexity of Smooth MDPs and demonstrate that, in the presence of a simulator, an algorithm with minimax-optimal sample complexity (with near-log factors) exists. In the online interaction scenario, we propose three algorithms for Smooth MDPs, each associated with a different learning objective, and analyze their performance by obtaining regret bounds that explicitly depend on the degree of process regularity. These results generalize classical bounds for simpler classes and improve the state of the art for kernelized MDPs with Matérn kernels.

**Keywords:** Reinforcement Learning, Smooth MDPs, Smoothness, Nonparametric Regression, Kernel Methods, Continuous Spaces, Control Theory

# Abstract in lingua italiana

Adattare l'apprendimento per rinforzo a processi decisionali di Markov con spazi di stato e azione continui è cruciale per molte applicazioni reali, in particolare in robotica e controllo. Dal punto di vista teorico, tuttavia, questo regime rimane poco compreso: mentre per gli MDP tabulari esistono garanzie nette, nel caso continuo manca ancora una teoria generale con algoritmi dimostrabilmente efficienti. In questa tesi introduciamo una nuova classe di problemi, gli MDP Lisci, che fornisce un quadro unificante per l'apprendimento per rinforzo in ambienti continui. Gli MDP Lisci sono definiti tramite ipotesi di regolarità funzionale sull'operatore di Bellman e sulle dinamiche di ricompensa e transizione. Questo framework generalizza in modo rigoroso diversi modelli studiati in letteratura, tra cui banditi continui, LQR e loro estensioni, MDP lipschitziani e kernelizzati. Nonostante ipotesi di tipo Sobolev-Hölder siano centrali nella regressione non parametrica e recentemente esplorate nei problemi bandit, prima di questo lavoro mancava una trattazione sistematica nel contesto dell'apprendimento per rinforzo. Dal punto di vista statistico, caratterizziamo la complessità degli MDP Lisci e dimostriamo che, in presenza di un simulatore, esiste un algoritmo con complessità campionaria minimax-ottimale (a fattori logaritmici). Nel regime di interazione online, proponiamo tre algoritmi per gli MDP Lisci, ciascuno associato a un diverso obiettivo di apprendimento, e ne analizziamo le prestazioni ottenendo limiti di regret che dipendono esplicitamente dal grado di regolarità del processo. Questi risultati generalizzano limiti classici per classi più semplici e migliorano lo stato dell'arte per MDP kernelizzati con kernel Matérn.

**Parole chiave:** Apprendimento per Rinforzo, MDP Lisci, Liscezza, Regressione Nonparametrica, Metodi Kernel, Spazi Continui, Teoria del Controllo

# Contents

**Abstract** **i**

**Abstract in lingua italiana** **iii**

**Contents** **v**

**1 Introduction** **1**
1.1 Thesis Structure . . . 3
1.2 Contributions . . . 5
1.2.1 Other Works Carried on During the PhD . . . 6

**2 Reinforcement Learning** **7**
2.1 Bandits: State-free Sequential Decision Making . . . 8
2.2 Markov Chains and Markov Decision Processes . . . 11
2.2.1 Value Functions and Bellman Operators . . . 14
2.3 Taxonomy of Reinforcement Learning Methods . . . 16
2.3.1 Interaction Models in Reinforcement Learning . . . 17
2.3.2 Model-based, Value-based and Policy-based Methods . . . 19

**3 Foundations of RL Theory** **21**
3.1 Standard Assumptions . . . 21
3.2 Performance Metrics . . . 23
3.2.1 Statistical Complexity . . . 23
3.2.2 No-regret Learning . . . 25
3.3 Fundamental Results in Reinforcement Learning Theory . . . 28
3.3.1 The OFU Principle . . . 29
3.3.2 Tabular MDPs . . . 31
3.3.3 MDPs with Linear Function Approximation . . . 35
3.3.4 Hardness Results for More General Classes . . . 40

**4 Learning in MDPs with Continuous State and Action Space 43**
4.1 Problem Definition and Preliminary Results . . . 43
4.1.1 The bandit case . . . 44
4.1.2 Smooth Bandits . . . 46
4.1.3 Statistical Complexity of Nonparametric Regression . . . 48
4.1.4 From Smooth Bandits to Smooth MDPs . . . 49
4.2 Existing Approaches for Continuous-State MDPs . . . 50
4.2.1 LQRs and their Generalizations . . . 51
4.2.2 Lipschitz MDPs . . . 55
4.2.3 A Note on the Dependency on $m$ . . . 58
4.2.4 Kernelized MDPs . . . 58
4.2.5 General Function Approximation . . . 61
4.2.6 State of the Art Summary and Sample Complexity . . . 65
4.3 Smooth MDPs: One Idea, Three Assumptions . . . 68
4.3.1 Mildly and Weakly Smooth MDPs . . . 71
4.3.2 Practical Relevance of Smooth MDPs . . . 77
4.4 Hierarchy of Continuous Space MDP Assumptions . . . 78
4.4.1 Non-parametric Classes . . . 81
4.4.2 Does General Function Approximation Work on Smooth MDPs? . . . 85
4.5 Research Questions on Smooth MDPs . . . 95

**5 Smooth MDPs: Sample Complexity 97**
5.1 Fourier Representation . . . 98
5.1.1 Issue with Standard LSVI . . . 100
5.1.2 Projection by Convolution . . . 102
5.2 Algorithm . . . 104
5.2.1 Choosing Datasets with Optimal Design . . . 106
5.3 Sample Complexity Analysis . . . 107
5.3.1 Upper Bound for Weakly Smooth MDPs . . . 108
5.3.2 Upper Bound for Mildly Smooth MDPs . . . 114
5.3.3 Misspecification of the Smoothness Parameter . . . 116
5.4 Final Considerations on Periodic Smooth MDPs and PLSVI . . . 117
5.4.1 Computational Complexity . . . 118
5.4.2 Are Periodic Smooth MDPs Easier than Smooth MDPs? . . . 119
5.4.3 Comparison with Non-parametric Statistics . . . 121

**6 Smooth MDPs: Regret Bounds 123**
6.1 Regret Bound for Weakly Smooth MDPs . . . 124

6.1.1 Reducing to MDPs with Low Inherent Bellman error . . . . . 125
6.1.2 Regret Upper Bound . . . . . 128
6.2 Improving the Regret Upper Bound for Mildly Smooth MDPs . . . . . 130
6.2.1 Locally Linearizable MDPs . . . . . 130
6.3 A Computationally Efficient Algorithm . . . . . 138
6.3.1 Numerical Simulations . . . . . 140
6.4 Summary of the Regret Bounds for Smooth MDPs . . . . . 142
6.4.1 Smooth and Kernelized MDPs: Which is the Optimal Regret Bound 146

**7 Conclusions 149**
7.1 Next Research Directions . . . . . 151
7.2 Conclusion of the Conclusions . . . . . 153

**A Notation Summary 155**

**B Mathematical Analysis 159**
B.1 Sobolev Spaces . . . . . 159
B.1.1 The Sobolev-Hölder Spaces $W^{\nu,\infty}$ . . . . . 160
B.2 Taylor Series . . . . . 161
B.3 Fourier Analysis . . . . . 163
B.3.1 Multi-dimensional Fourier Series . . . . . 165
B.4 Reproducing Kernel Hilbert Spaces . . . . . 165

**C Thompson Sampling on Multi-dimensional Action Spaces 169**

**D Details of the numerical simulations 171**

**E Missing Proofs 173**
E.1 Proofs from Chapter 3 . . . . . 173
E.2 Proofs from Chapter 5 . . . . . 176
E.3 Proofs from Chapter 6 . . . . . 177
E.3.1 Definition of the Quasi-optimal Solution . . . . . 177
E.3.2 Algorithm Analysis . . . . . 178
E.3.3 Decomposition of the Estimated Solution . . . . . 179
E.3.4 Failure Event . . . . . 182
E.3.5 Quasi-optimal solution and good event . . . . . 187
E.4 Regret Bound . . . . . 189
E.4.1 Reduction from Mildly Smooth to Locally Linearizable . . . . . 194

**Acknowledgements (ENG)** **199**

**Acknowledgements (IT)** **203**

**Bibliography** **207**

# 1 | Introduction

Historically, the success of Reinforcement Learning (RL) theory was built upon the tabular framework, where the system is characterized by a finite set of states and actions. In this discrete setting, the agent visits a finite number of states and takes actions from a finite set, allowing for the exact estimation of value functions. This led to a solid understanding of fundamental concepts like exploration-exploitation trade-offs and minimax regret bounds. However, most real-world problems where RL is applied are not intrinsically tabular. Applying these methods therefore requires discretizing the state space, often using a regular grid spanning all dimensions. This process can quickly lead to an explosion in the number of states, making it impractical for real-world applications such as robotics or industrial control, where states and actions are inherently continuous.

From a different lineage, Control Theory has dealt with continuous systems for decades. While RL often focuses on model-free approaches and general performance, Control Theory typically relies on specific structural assumptions, such as linearity in Differential Equations or quadratic cost functions (LQR), to ensure stability and optimality. These assumptions have a different "flavor" than those in RL; they prioritize the dynamical properties of the system over the statistical complexity of learning from samples. Several works have integrated these results in the theoretical RL literature, achieving theoretical guarantees for these problems.

Nonetheless, these structured assumptions are very specific, and only cover a relatively small, parametric, class of all the possible continuous problems where practical RL is applied. To address this challenge, a large body of work has extended RL theory to continuous spaces by imposing different forms of assumptions on the problem. Rather than forming a unified theory, these approaches have largely developed in isolation. The current landscape resembles an archipelago of disconnected islands, each corresponding to a different set of assumptions and techniques. A first line of work considers Lipschitz MDPs, where value functions or transition dynamics satisfy a Lipschitz-type constraint, i.e. $|f(x) - f(y)| \leq C\|x - y\|_2$ holds for any pair $x, y$ in their domain. These assumptions enable discretization-based methods with theoretical guarantees, but often suffer

from strong dependence on the ambient dimension. A second direction focuses on kernelized MDPs, where the value function or the transition model is assumed to lie in a reproducing kernel Hilbert space (RKHS). This framework allows for non-parametric generalization and has led to refined regret bounds, particularly for kernels such as Matérn. More broadly, recent work on function approximation in RL studies settings where value functions belong to structured function classes, such as linear spaces or more general hypothesis classes. While this line of research has made significant progress, applying it to continuous MDPs often requires hypothesis classes so large that, as we demonstrate, the resulting performance guarantees become vacuous.

All these approaches can be viewed as different ways of enabling some form of generalization. Lipschitz MDPs assume that nearby states have similar values, kernel-based methods encode smoothness through reproducing kernel Hilbert spaces, and general function approximation frameworks impose structure via restricted hypothesis classes. Yet, they rely on fundamentally different assumptions and analytical tools, making it difficult to compare results or transfer insights across frameworks. As a result, there is currently no unified theory of Reinforcement Learning in continuous spaces that captures these perspectives within a single framework.

To address these gaps, this thesis develops a unifying framework for Reinforcement Learning in MDPs with continuous state and action spaces. The goal is to bring together a broad range of assumptions from the literature under a single umbrella, by relating them to a formal notion of smoothness. By interpreting different MDP classes through the lens of function space regularity, this perspective clarifies the hierarchy of statistical difficulty across continuous models. In fact, existing approaches implicitly rely on different notions of regularity. As we show, LQR induces dynamics that are infinitely differentiable. Lipschitz MDPs impose that nearby states behave similarly and kernelized methods encode regularity through RKHS norms. In this work, we adopt a perspective rooted in functional analysis, where regularity is characterized through smoothness properties of functions. In particular, we rely on Sobolev-Hölder spaces, which provide a flexible and well-established framework to quantify smoothness. Sobolev spaces play a central role in many areas of science, particularly in the analysis of physical systems governed by partial differential equations. Heat, fluid dynamics, and electromagnetic waves are extensively studied using the theory of Sobolev spaces.

Broadly speaking, a function is in a Sobolev space if its derivatives up to a certain order are bounded in a specific norm. While derivatives capture local behavior, Sobolev theory provides powerful results that connect local smoothness to global approximation properties, moving past the "nearest point" approach characteristic of Lipschitz functions. From

this perspective, several existing models can be reinterpreted as special cases corresponding to different levels and types of smoothness. For instance, Lipschitz MDPs correspond to first-order smoothness constraints in uniform norm, while kernel methods with Matérn kernels can be linked to Sobolev spaces of fractional order in the mean-squared norm.

Building on this reinterpretation, the goal of this thesis is twofold. First, we introduce and study a broad class of continuous MDPs, referred to as Smooth MDPs, where different notions of smoothness are explicitly encoded in the model. This framework, which is designed to subsume a wide range of existing approaches, is presented in three different flavors of increasing generality (Strongly, Mildly and Weakly Smooth MDPs) which enforce Sobolev-type assumptions in different ways.

Second, we investigate whether classical tools from Sobolev function approximation are sufficient to derive statistical complexity and regret guarantees of the type studied in Reinforcement Learning theory. In particular, we aim to understand to what extent smoothness alone can characterize the difficulty of learning in continuous environments.

Through this lens, we obtain both positive and negative results. While minimax-optimal statistical guarantees are achievable across regimes, deriving regret bounds proves significantly more challenging, and sublinear guarantees are not always attainable. Moreover, our lower bounds show that, at a high level of generality, an exponential dependence on the planning horizon is unavoidable in the worst case. These findings provide a unified perspective on continuous Reinforcement Learning and highlight key open challenges toward a complete theory.

## 1.1. Thesis Structure

This thesis is organized as follows. Chapter 2 provides a self-contained introduction to Reinforcement Learning. We begin by reviewing the bandit setting as a simple instance of sequential decision making without state, and then introduce Markov chains and Markov decision processes, together with value functions and Bellman operators. The chapter concludes with a taxonomy of Reinforcement Learning methods, distinguishing between different interaction models and between model-based, value-based, and policy-based approaches. The purpose of this chapter is to establish notation and background, rather than to provide a comprehensive survey.

Chapter 3 focuses on the theoretical foundations of Reinforcement Learning. We first introduce the standard assumptions commonly adopted in the literature and discuss the main performance metrics, with particular emphasis on statistical complexity and no-

regret learning. We then review fundamental results in Reinforcement Learning theory, including the principle of optimism in the face of uncertainty, sharp guarantees for tabular MDPs, extensions to linear function approximation, and hardness results for more general function classes. This chapter highlights both what is known and the limitations of existing theory.

Chapter 4 addresses learning in MDPs with continuous state and action spaces. We start by introducing the problem setting and presenting preliminary results, beginning from the bandit case and nonparametric regression, and then discussing the challenges that arise when extending these ideas to sequential decision making. We review the main existing approaches for continuous-state MDPs, including linear-quadratic regulators, Lipschitz MDPs, kernelized methods, and general function approximation schemes, and summarize their sample complexity guarantees. The chapter then introduces Smooth MDPs as a unifying framework, presenting three levels of smoothness assumptions and organizing them into a hierarchy that clarifies their relationships with other continuous-space models.

Chapter 5 is devoted to the sample complexity of Smooth MDPs. We first discuss the limitations of standard least-squares value iteration algorithms in this setting. Then, we propose a new algorithm based on the novel technique of projection by convolution that solves the issues with LSVI in this context. In the chapter, we also briefly introduce the theory of optimal experimental design, which is an important part of the algorithm as well. The main part of the chapter is dedicated to the sample complexity analysis, where we derive upper bounds for Weakly and Mildly Smooth MDPs, study robustness to misspecification of the smoothness parameter, and compare our results with classical rates from nonparametric statistics. The chapter concludes with a discussion on periodic smooth MDPs, computational complexity, and the relationship between periodic and non-periodic smoothness assumptions.

Chapter 6 studies regret minimization in smooth MDPs. We first derive regret bounds for Weakly Smooth MDPs by reducing the problem to classes with low inherent Bellman error, and then improve the upper bounds for mildly smooth processes by exploiting local linearization properties. We also introduce a computationally efficient algorithm and validate the theoretical findings through numerical simulations. The chapter concludes with a unified summary of the regret guarantees obtained for Smooth MDPs.

Finally, chapter 7 concludes the thesis with a summary of the main contributions and a discussion of open problems and future research directions.

## 1.2. Contributions

The vast majority of the results we will present in this thesis comes from three works.

1. In [80] we introduce a class of Smooth MDPs and the corresponding algorithm for learning the optimal policy with optimal statistical complexity.
2. In [79] we define the classes of strongly and weakly smooth MDPs in a structured way, and compare them with other classes in the RL literature for continuous states and actions. Furthermore, we provide a reduction from these new classes of smooth MDPs to the already known classes of linear MDPs.
3. In [78], we present the class of Mildly Smooth MDPs as a middle ground between the Weakly and Strongly Smooth assumptions. This paper provides a detailed comparison with the Kernelized classes and Lipschitz MDPs. Furthermore, by using a feature map localization technique, we provide an algorithm with an improved regret bound compared to the one presented in the previous paper.

While writing this thesis we refined some of the results of previous publications, and introduced some completely new ones. In particular,

- In section 4.3.1, the result of including TV-Lipschitz MDPs into Mildly Smooth ones is novel.
- Theorem 32 about the lower bound on the Eluder dimension for Sobolev spaces is novel.
- While theorem 41, was already presented in [80], we provide here a novel proof that is much simpler.
- Theorem 42, on the sample complexity of Mildly Smooth MDPs, is novel.
- Section 6.1.1 contains a result that is slightly different from the original one in [79] (theorem statement does not change, but one part of the proof does). This allows to clarify the comparison with the standard analysis of the ELEANOR algorithm presented in [151].
- The proposed future work on the mixed smoothness classes in the future works is a novel idea that we had while writing this thesis.

Writing this thesis also provided the opportunity to compare the work on Smooth MDPs with the literature much more extensively.

### 1.2.1. Other Works Carried on During the PhD

In this thesis, I focus on a subset of my doctoral work, with the goal of providing a coherent overview of my contribution to Reinforcement Learning in continuous state and action spaces. For completeness, I briefly discuss other research directions I explored during my PhD.

In my work [81], I studied MDPs from a different perspective, namely the setting where the objective is to optimize the configurator rather than the policy. In this setting, the agent does not directly modify the policy, but instead acts on the transition dynamics in order to maximize a possibly different objective function.

In [82], I explored the possibility of extending SINDy (Sparse Identification of Nonlinear Dynamics), a system identification algorithm, to the control of MDPs with sparsity structure.

Although both works fall under the umbrella of Reinforcement Learning, they are not directly aligned with the main focus of this thesis.

More broadly, during my PhD I worked on several additional research directions that are not directly related to continuous-state RL. In particular, I studied bandit algorithms. In [16], I analyzed bandit problems where the reward depends on past actions via an autoregressive process, and designed learning algorithms with regret bounds that adapt to the memory and stability of the reward dynamics.

In [77], I introduced a bandit setting where feedback is provided as a ranking over actions rather than numerical rewards, and proposed algorithms with optimal regret that exploit structural properties of the ranking feedback.

# 2 | Reinforcement Learning

As anticipated, Reinforcement Learning (RL, [124]) is the study of how an agent learns to make decisions by interacting with an environment to achieve long-term goals without explicit instructions. At its core, RL formalizes the problem of learning through trial and error: the agent chooses actions, observes the resulting state, and receives feedback in the form of rewards. To fully situate RL within the broader context of Computer Science, we must first consider the principal branches of machine learning.

Machine Learning (ML) is a subfield of artificial intelligence concerned with the design and analysis of algorithms that learn patterns or behaviors from data, with the goal of improving performance on a given task without being explicitly programmed for every scenario. At its core, ML formalizes the idea of learning from experience through a statistical lens, where the learner seeks to generalize beyond the observed data.

The field of machine learning is typically divided into three main branches:

1. Supervised Learning,
2. Unsupervised Learning,
3. and Reinforcement Learning.

In Supervised Learning, the algorithm is provided with labeled examples $\{x_i, y_i\}_{i=1}^n$ and aims to learn a function that maps inputs $x$ to outputs $y$. This is historically the most studied setting, and is often divided again in two other sets depending on the nature of the response variable. If $y$ takes values in a continuous set, like the real numbers, the problem takes the name of *regression*. If $y$ takes values in a discrete set, either of numbers or of labels, it is called *classification*. The methods developed in these fields are the backbone of many Reinforcement Learning algorithms that we are going to use.

Unsupervised Learning, on the other hand, deals with unlabeled data and focuses on discovering underlying structures, such as clusters or low-dimensional embeddings. Some examples of problems where this approach is used include customer segmentation, where a business wants to group its customers into different segments based on their purchasing

habits without having any predefined labels for these groups, and image compression, where the goal is to reduce the size of an image file by representing the image with fewer pixels or features while maintaining its visual quality.

Reinforcement Learning, the primary focus of this thesis, diverges significantly from the aforementioned paradigms. Rather than relying on a static dataset, RL entails an online learning process driven by the agent's interaction with the environment. The agent executes actions and, in turn, receives feedback in the form of rewards, which it seeks to maximize over time. Crucially, each reward is explicitly associated with the agent's previous action. In its simplest formulation, where the environment lacks a notion of state, allowing the agent to repeatedly query the same reward function, the framework reduces to the *bandit problem*, which will be formally introduced in the following section.

## 2.1. Bandits: State-free Sequential Decision Making

The name "Bandit" comes from a simple, yet very practical analogy. Imagine a person is in a casino facing a row of $A$ slot machines, each with a single lever, often called a "one-armed bandit".

- Multiple Arms: the agent has $A$ different levers (the "arms", denoted with $a = 1, ...A$) to pull.
- Unknown Payouts: Each machine (arm) has a different, fixed, but unknown probability distribution of giving a reward. Some machines might pay out 10% of the time, others 20%, and one might be the best at 30%.
- The agent interacts with these machines in a sequential way, pulling a different arm in every round. The goal is to maximize their total winnings over a fixed number of rounds.

The challenge is that the agent does not know the true payout rate of any machine at the start. This creates the need to balance between a) pulling all the arms, to have a good estimate of the mean payout of each machine (exploration) and b) to pull the arm which gives the highest estimated reward (exploitation).

We formalize this process as in the survey [68]. A stochastic bandit is a collection of distributions $\{\mathscr{P}_a\}_{a\in\mathcal{A}}$, where $\mathcal{A}$ is the set of available actions ($\mathcal{A} = \{1, \dots A\}$ in the previous example). The learner and the environment interact sequentially over $K$ rounds. In each round $k$, the learner chooses an action $a_k$, corresponding to the arm to pull. The environment then samples a reward $y_k$ from distribution $\mathscr{P}_{a_k}$ and reveals it to the learner. The

interaction between the learner (or policy) and environment induces a sequence of outcomes $a_1, y_1, a_2, y_2, ..., a_K, y_K$. The following assumption, over the sequence of outcomes, characterizes the bandit setting.

**Assumption 1.** *The conditional distribution of reward $y_k$ given $a_1, y_1, a_2, y_2, ..., a_{k-1}, y_{k-1}$ is $\mathscr{P}_{a_k}$, regardless of the history of the process.*

While formally the bandit is identified by the full distribution of rewards associated to each arm, the working assumptions under most of the algorithms only concern the mean corresponding to each arm. This is referred to as **reward function**, and writes as

$$r(a) := \mathbb{E}[y_k | a_k = a].$$

As we have anticipated, the goal in a bandit problem is to maximize the expected cumulative reward in the $K$ rounds. So, calling $\pi$ the policy, i.e. the rule we use to select the action $a_k$ from the history of the process, we write this objective as follows:

$$J^\pi := \mathop{\mathbb{E}}_{\pi}\left[\sum_{k=1}^{K} y_k\right] = \mathop{\mathbb{E}}_{\pi}\left[\sum_{k=1}^{K} r(a_k)\right].$$

In the last expression, we used assumption 1, as well as the definition of reward function, to restrict the expected value to the action. As the previous expression basically grow linearly in $K$, it is convenient to introduce a different performance measure, that we call the expected regret:

$$\overline{R}_K(\pi) := K \sup_{a \in \mathcal{A}} r(a) - \mathop{\mathbb{E}}_{\pi}\left[\sum_{k=1}^{K} r(a_k)\right].$$

Since this differs only by a constant shift and sign inversion, to minimize the expected regret corresponds to maximize the cumulative reward. Lastly, if we care about the overall performance of the algorithm instead of just its expected value, we can define the **regret**, the variable that we were making the expectation of:

$$R_K := K \sup_{a \in \mathcal{A}} r(a) - \sum_{k=1}^{K} r(a_k) \tag{2.1}$$

Doing so, we consider regret with respect to the expected rewards, removing randomness due to the reward noise.

The following examples of bandit problems are the most studied in the literature.

**Example 2** (Multi-armed bandits (MAB)). *In this problem, the space of arms is discrete*

*and we take, by convention,* $\mathcal{A} := \{1, \dots A\}$. *The set of possible reward functions is* $\mathcal{F} := \{r : \mathcal{A} \to [0, 1]\}$, *that is, all functions on* $\mathcal{A}$ *taking values between* 0 *and* 1.

**Example 3** (Linear bandits). *In this problem, the space of arms is a set* $\mathcal{A} \subset \mathbb{R}^d$, *for* $d \in \mathbb{N}$. *This means that the agent cannot, in general, try all the arms even once. On the other side, the set of possible reward functions is restricted to* $\mathcal{F} := \{r(a) = \theta^\top a,\ \theta \in \mathbb{R}^d\}$. *Imposing that the reward function is linear, allows the agent to generalize between different arms.*

More generally, the several classes of bandits that we can find in the literature usually arise from different assumptions on the arm space and reward function. Indeed, beyond the classical stochastic and linear models, a rich variety of formulations has been proposed. In Combinatorial bandits, the agent selects, at each round, as arm a subset (so that $\mathcal{A} = 2^{\widetilde{A}}$, the power set of a given set $\widetilde{A}$) of a given set of "elementary arms" and the reward depends on an aggregation of the single rewards. This class captures applications such as online advertising, network routing, and recommendation systems [25]. Neural bandits [152] extend this idea further by employing neural networks as function approximators for the reward, allowing learning in highly complex and non-linear environments. Assuming $\mathcal{F}$ to be a set of parametrized neural networks brings an additional level of complexity compared to linear bandits. Other works study Lipschitz bandits [64], where arms lie in a metric space and the reward function is assumed to vary smoothly, allowing the agent to generalize across nearby actions. This corresponds to assuming that $\mathcal{F}$ is the class of Lipschitz functions (or even a smoother class, as we are going to see later in this thesis). On a higher level of abstraction, some works assume $\mathcal{F}$ to be a function class with low eluder dimension [107], which capture more general forms of structured dependencies in the reward function and allow regret bounds that scale with intrinsic complexity rather than the usual forms of function approximation.

While the bandit setting is already interesting for several applications, it remains limited by the absence of an evolving environment: each decision is independent of the past, except for the information gained through exploration. To capture richer sequential decision-making scenarios, one can move from bandits to Reinforcement Learning, which is the main topic of this work. Here the agent's actions influence not only the immediate reward but also the future opportunities it will face. This is formalized by introducing a notion of state, which evolves over time according to a Markov process controlled by the agent's actions.

## 2.2. Markov Chains and Markov Decision Processes

Let $\mathcal{S}$ be an arbitrary measurable space and $s_1, s_2, ...s_H$ be a sequence of random variables taking values in $\mathcal{S}$. We say that the random variables form a Markov chain if there is a sequence of functions $\{p_h\}_{h=1}^H$ with $p_h : \mathcal{S} \to \Delta(\mathcal{S})$ (where $\Delta(\mathcal{X})$ is the set of probability distributions over a set $\mathcal{X}$) such that, for any $\mathcal{S}' \subseteq \mathcal{S}$ measurable and any $h \in \{1, \dots H-1\}$, we have

$$\mathbb{P}(s_{h+1} \in \mathcal{S}' | s_1, \dots s_h) = \int_{\mathcal{S}'} p_h(s|s_h) ds.$$

This condition expresses the fact that the next state $s_{h+1}$ does not depend on the full history of the process, but just on the previous state $s_h$. For this reason, this process is often called *memoryless*.

Markov chains have found widespread applications across science and engineering, particularly in modeling dynamic systems that evolve over time under uncertainty. In such systems, the Markov property provides a natural abstraction: the future state depends only on the present, not on the full history. This framework is used in physics to describe particle dynamics and diffusion processes (see [95]) so that $s_h$ contains position and speed of some elements of a system; in biology to model population growth or genetic mutations; and in economics to capture market fluctuations and consumer behavior.

To generalize the Markov chain framework to the problems of stochastic control, we have to introduce an action, that influences the evolution of the chain, and a reward function, that the agent wants to maximize. As a result, we embed the bandit objective, introduced in the previous section, into the Markov chain framework, thereby defining the Markov Decision Process [103].

**Definition 1** (Markov Decision Process). *We call Markov Decision Process (MDP) a tuple* $(\mathcal{S}, \mathcal{A}, H, \{p_h\}_{h=1}^H, \{r_h\}_{h=1}^H)$, *where*

- $\mathcal{S}$ *is the state space, a measurable space,*
- $\mathcal{A}$ *is the action space, another measure set,*
- $H$ *is the time horizon of the interaction,*
- $\{p_h\}_{h=1}^H$, *with* $p_h : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ *are the transition functions, one for each stage* $h$ *of the process,*
- $\{r_h\}_{h=1}^H$, *with* $r_h : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ *are the reward functions, one for each stage* $h$ *of the process,*

*such that the following equations are verified:*

$$\forall h \in \{1, \dots H\}, \mathcal{S}' \subseteq \mathcal{S} \text{ meas.} \qquad \mathbb{P}(s_{h+1} \in \mathcal{S}' | s_h, a_h, \dots s_1, a_1) = \int_{\mathcal{S}'} p_h(s|s_h, a_h) ds. \tag{2.2}$$

$$\forall h \in \{1, \dots H\} \qquad \mathbb{E}[y_h | s_h, a_h, \dots s_1, a_1] = r_h(s_h, a_h), \tag{2.3}$$

*where $y_h$ is the random reward that the agent receives at stage $h$.*

The elements in the preceding definition mirror the properties discussed earlier. Specifically, eq. (2.3) serves as the counterpart to the fundamental assumption in bandit problems: it ensures that, for a fixed state, the stochastic reward $y_h$ has an expected value of $r_h(s, a_h)$, independent of the history. The inclusion of the state, which is itself a random variable, introduces a higher level of complexity. Similarly, eq. (2.2) extends the core assumption of Markov chains; the distinction here is that the state does not evolve endogenously but is actively influenced by the agent's actions. This interaction is the what determines the true difficulty of the setting. We define the policy as the rule governing the agent's action selection at each time step, formalized as follows.

**Definition 2** (Policy). *A policy is a collection $\{\pi_h\}_{h=1}^H$ of functions $\pi_h : \mathcal{S} \to \Delta(\mathcal{A})$, that is, from the states to the probability distributions over the actions. In the following, we will write*

- *$\pi_h(\cdot|s_h)$ the probability distribution on the actions corresponding to the policy.*
- *$a_h = \pi_h(s_h)$ if the policy is deterministic.*

When the policy is fixed, we can remove the dependency on the action and obtain a genuine Markov chain. Specifically, given $\{\pi_h\}_{h=1}^H$ a series of mappings $\mathcal{S} \to \Delta(\mathcal{A})$, choosing the action as $a_h \sim \pi_h(\cdot|s_h)$, for every measurable $\mathcal{S}' \subseteq \mathcal{S}$ leads to

$$\mathbb{P}(s_{h+1} \in \mathcal{S}' \mid s_h, s_{h-1}, \dots, s_1) = \int_{\mathcal{S}} \int_{\mathcal{A}} p_h(s \mid \boxed{s_h}, a)\, \pi_h(a \mid s_h)\, \mathrm{d}a\mathrm{d}s.$$

This formula, which follows from the law of total probability, shows how the distribution of the next state depends only on that of the previous state, thus respecting the Markov property. In this scenario, where the policy is fixed, the sequence of states thus follows a Markov chain, while the corresponding reward satisfies:

$$\mathbb{E}[y_h \mid s_h, \dots, s_1] = \mathbb{E}[r_h(s_h, a_h) \mid s_h] = \int_{\mathcal{A}} r_h(\boxed{s_h}, a)\, \pi_h(a \mid s_h)\, \mathrm{d}a.$$

This structure is called a Markov reward process [83].

Before proceeding with the theory of MDPs, it is useful to give some examples to understand the variety of problems that one can model as an MDP. The first model that we are considering is a well-studied problem in *control theory* [7], which we will return to many times throughout this thesis.

**Example 4.** *[LQR with Gaussian noise] Consider a finite-horizon, discrete-time LQR problem where the state evolution follows*

$$s_{h+1} = As_h + Ba_h + w_h,$$

*with:*

- $s_h \in \mathbb{R}^{m_1} =: \mathcal{S}$ *as the state,*
- $a_h \in \mathbb{R}^{m_2} =: \mathcal{A}$ *as the control action,*
- $A$ *and* $B$ *given matrices of appropriate dimensions,*
- $w_h \sim \mathcal{N}(0, \Sigma)$ *i.i.d. Gaussian noise, where* $\Sigma$ *is an* $m_1 \times m_1$ *symmetric positive semi-definite matrix.*

*Under such a linear model for the evolution, the agent has to maximize a reward function. The per-stage reward is quadratic in both state and action:*

$$y_h = r_h(s_h, a_h) = -s_h^\top Q s_h - a_h^\top R a_h,$$

*where* $Q$ *and* $R$ *are positive semi-definite matrices capturing the cost on states and actions. It is easy to verify that such a structure satisfies both of the assumptions needed to have an MDP.*

1. *eq.* (2.2) *is satisfied, as* $p_h$ *writes as*

$$p_h(s|s_h, a_h) = \mathcal{N}(s'; As_h + Ba_h, \Sigma).$$

2. *eq.* (2.3) *is satisfied, as* $y_h = -s_h^\top Q s_h - a_h^\top R a_h$ *(the reward is deterministic, so it is not even necessary to write the expectation).*

*Crucially, both are independent on the past given* $s_h, a_h$*.*

The next example, which has discrete states and actions, was introduced in [124].

**Example 5** (Gridworld). *The Gridworld environment can be models an agent which moves on a grid, with each action having a certain probability of succeeding. The goal is to reach a certain position* $s_{target}$.

- *The state space* $\mathcal{S}$ *corresponds to the two-dimensional infinite grid* $\mathbb{Z}^2$, *where each state* $s = (i, j)$ *represents the position of the agent at row* $i$ *and column* $j$.
- *The action space is* $\mathcal{A} = \{up, down, left, right\}$, *representing the agent's possible moves.*
- *The transition probability function* $p_h(\cdot \mid s_h, a_h)$ *captures the stochastic dynamics of the environment. Typically, the agent moves in the intended direction with high probability (e.g.,* 0.8*), but with some probability (e.g.,* 0.1*) it may move perpendicular to the intended direction due to slippage:*

$$p_h(s \mid s_h, a_h) = \begin{cases} 0.7 & \text{if } s \text{ is the intended successor of } s_h \text{ under } a_h, \\ 0.1 & \text{if } s \text{ is a lateral successor of } s_h \text{ due to noise}, \\ 0 & \text{otherwise.} \end{cases}$$

- *The reward function at any stage* $h$ *writes as*

$$y_h = r_h(s, a) = \begin{cases} 1 & s = s_{target} \\ 0 & s \neq s_{target} \end{cases}.$$

*In this case, eq.* (2.2) *and eq.* (2.3) *are satisfied by definition.*

These examples are only toy models, but in general MDPs have applications in a wide variety of disciplines (see, e.g., [20]). In order to find the optimal policy in this type of problems, it is necessary to develop a theory known as *dynamic programming* [18], which defines value functions and Bellman operators.

### 2.2.1. Value Functions and Bellman Operators

Let $\pi = \{\pi_h\}_{h=1}^H$ be a fixed policy. Thanks to the Markov property, one can define a function which, given a state of the MDP, returns the expected cumulative reward for all future stages, starting from that state. For $h \in [H]$ and $s \in \mathcal{S}$, we define

$$V_h^\pi(s) := \mathbb{E}_{a_{h'} \sim \pi_{h'}(s_{h'})} \left[ \sum_{h'=h}^{H} r_{h'}(s_{h'}, a_{h'}) \,\middle|\, s_h = s \right].$$

We call this function the *state value function*. The reason for this name is that we can analogously define another value function, this time conditioning not only on the current state, but also on the action:

$$Q_h^\pi(s,a) := \mathbb{E}_{a_{h'}\sim\pi_{h'}(s_{h'})}\left[\sum_{h'=h}^{H} r_{h'}(s_{h'},a_{h'}) \,\middle|\, s_h = s, a_h = a\right].$$

The fact that these value functions contain a sum over all future states makes them difficult to compute explicitly. Fortunately, they are linked by recursive equations known as *Bellman equations*. We can fix $Q_H^\pi(s,a) = r_H(s,a)$ and write, for every $h \in [H-1]$,

$$Q_h^\pi(s,a) = r_h(s,a) + \int_{\mathcal{S}} V_{h+1}^\pi(s')\, p_h(s'|s,a)\,\mathrm{d}s', \qquad V_h^\pi(s) = \int_{\mathcal{A}} Q_h^\pi(s,a)\,\pi_h(a|s)\,\mathrm{d}a. \tag{2.4}$$

These equations express an intuitive fact: the value of a state or state-action pair equals the immediate reward plus the expected value of the next state under the policy and the transition dynamics.

While these two value functions are good for estimating the reward that a policy of going to achieve from a given state (or state action pair), the so-called *policy evaluation*, our goal is more ambitious. The goal of Reinforcement Learning is to find a policy that maximizes the expected cumulative reward. Formally, we define the optimal value functions as

$$V_h^*(s) := \sup_\pi V_h^\pi(s), \qquad Q_h^*(s,a) := \sup_\pi Q_h^\pi(s,a).$$

Under proper compactness assumptions (finite states and actions or smooth transitions, see for a detailed explanation of the topic), there is a policy $\pi^*$ that achieves the supremum. Computing $\pi^*$ by explicitly maximizing over the space of policies would be unfeasible, as this set grows exponentially in the size of $\mathcal{S}$, even if we limit to deterministic ones. Still, a simple observation makes the problem easier. Indeed, for the policy $\pi^*$ to be optimal, it is necessary that in every state $s$ it plays actions $a$ such that $Q^*(s,a) = \sup_{a'\in\mathcal{A}} Q^*(s,a')$; otherwise a strictly better policy by changing $\pi_h^*(s)$. Replacing this condition into the Bellman equations eq. (2.4) gives

$$Q_h^*(s,a) = r_h(s,a) + \int_{\mathcal{S}} V_{h+1}^*\, p_h(s'|s,a)\,\mathrm{d}s', \qquad V_h^*(s) = \sup_{a\in\mathcal{A}} Q_h^*(s,a). \tag{2.5}$$

By starting at $Q_h^*(s,a) = r_H(s,a)$, this pair of equations can efficiently compute $Q_h^*(s,a)$ at any stage. Once the procedure is complete, the optimal policy (if exists) is given by

$$\pi_h^*(s) := \arg\max_{a \in \mathcal{A}} Q_h^*(s,a).$$

Note that this procedure gives a deterministic policy. The recursive equations eq. (2.5) define a nonlinear operator known as the *Bellman optimality operator*. This operator, which is of independent interest, is formalized in the following

**Definition 3** (Bellman optimality operator)**.** *The Bellman optimality operator $\mathcal{T}_h$ is the operator acting on any function $f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ as*

$$\begin{aligned}\mathcal{T}_h f(s,a) &:= r_h(s,a) + \mathop{\mathbb{E}}_{s' \sim p_{h+1}(\cdot|s,a)} [\sup_{a \in \mathcal{A}} f(s',a')] \\ &= r_h(s,a) + \int_{\mathcal{S}} \sup_{a' \in \mathcal{A}} f(s',a') p_h(s'|s,a) ds'\end{aligned}$$

Under this lens, the optimal state-action value function (and, as a result, the optimal state value function) are given by $H$ iteration of the Bellman optimality operator starting from $Q_{h+1}^*(s,a) = 0$.

This procedure works perfectly if the transition model and reward function are assumed to be known, and the resulting algorithm takes the name of *value iteration*. However, in Reinforcement Learning the model is typically unknown, and the agent must learn the optimal policy directly from experience. This motivates the study of a much larger family of algorithms that we will introduce in the next section.

## 2.3. Taxonomy of Reinforcement Learning Methods

As we mentioned, solving an MDP means finding an optimal (or approximately optimal) policy. When the MDP is fully known (i.e., the agent has access to transition and reward function), the problem is exclusively computational. We refer to it as *planning*. Planning is not a learning problem since no experience/data is involved: as we have shown with the value iteration algorithm, it is sufficient to maximize a known function.

Differently from planning, RL is concerned with optimizing the policy when the MDP is not known a priori but it is possible to interact with the environment. We switched from the term "learning the optimal policy" to "optimizing the policy" for a reason: due to the stochastic nature of the interaction, an exact computation of the optimal policy is usually impossible. Nonetheless, we can provide various metrics of convergence of the performance

of the learned policy to the optimal one, which will be formalized in section 3.2.1. In the literature, the agent-environment interaction which allows the agent to gather samples in usually modeled in one of the following ways.

### 2.3.1. Interaction Models in Reinforcement Learning

The agent-environment interaction paradigm in Reinforcement Learning and related fields can be formulated with three principal modes for how data is gathered and used: generative model access, online interaction, and batch (offline) data collection. Each approach presents distinct theoretical and practical implications, influencing algorithm design, analysis, and empirical RL protocols.

**Online interaction** describes the traditional RL setting, where the agent interacts sequentially with the environment in real time, receiving observations, taking actions, and receiving rewards that depend on the state-action pair. In formulas, if the agent is currently on state $s_h$ and takes action $a_h$, it can observe the following signal:

$$s_{h+1} \sim p_h(\cdot|s_h, a_h) \qquad y_h = r_h(s_h, a_h) + \eta_h,$$

where, as in the bandit case, $\eta_h$ is a zero-mean noise. Then, the interaction continues from $s_{h+1}$, until reaching $H$, and then re-starts from the beginning for a new initial state $s_1$ which comes from $p_1(\cdot)$. To avoid confusion, between the states of different episodes, we use the following notation. Calling $K$ the total episodes of interaction, we note as $s_h^k$ the $h-$th state visited in the $k-$th episode, for $h \in \{1, \dots H\}$ and $k \in \{1, \dots K\}$. Therefore, the full ordering of the state that the agent can see is the following.

$$\underbrace{s_1^1, \dots s_h^1, \dots s_H^1}_{\text{First episode}}, \dots, \underbrace{s_1^k, \dots s_h^k, \dots s_H^k}_{k-\text{th episode}}, \dots, \underbrace{s_1^K, \dots s_h^K, \dots s_H^K}_{\text{Last episode}}.$$

Similarly, we note as $y_h^k$ the random reward collected at the $h-$th state visited in the $k-$th episode. Here, data collection is dynamic, as any observation $s_h^k$ can convince the agent to update its policy and indirectly influence all the following episodes. What makes this mode particularly difficult is the fact that in order to identify the "best states", the agent needs to obtain multiple samples from the transition and reward functions, which in turn requires to reach them many times. For this reason, exploration-exploitation trade-off becomes a central challenge: the agent has to visit every state sufficiently many times, but not enough to risk getting a bad reward.

Most of the RL literature focuses on this setting [124], which is particularly relevant for problems requiring adaptation and learning from ongoing interaction, such as robotic control, web recommendation, and real-world autonomous systems.

**Generative model** In the generative model setting, the agent is assumed to have access to a simulator that allows it to query any (state, action) pair and receive the corresponding transition and reward. Crucially, the agent does *not* receive the full function $p_h(\cdot|s,a)$ and the exact value $r_h(s,a)$ for the chosen pair, but rather a single sample $s' \sim p_h(\cdot|s,a)$ and the random reward $y_h$ with mean $r_h(s,a)$. In other words, the feedback is the same as in the online interaction, and the advantage is in the fact that the agent can query an arbitrary $(s,a)$ pair without having to reach $s$.

This is a powerful abstraction, particularly common in theoretical RL literature, as it allows learning a quasi-optimal policy in scenarios that would be very challenging due to the need for exploration. Having access to a simulator is not merely a convenient assumption for easing theoretical analysis, it actually reflects many real-world scenarios. In practice, the reward function is typically known, as it is defined by the designer of the problem, while the transition dynamics are often implemented through a simulator to prevent potential damage to the real system. Reinforcement Learning techniques when paired with environment simulators, have driven remarkable advances across diverse domains, enabling breakthroughs in robotics, achieving superhuman performance in board games, and mastering complex video games. (see [61], [114], [115], [19]).

**Batch RL** A rapidly emerging branch of Reinforcement Learning is known as batch RL (also referred to as offline RL) [67, 102]. Unlike online RL, in this paradigm, the agent learns exclusively from historical datasets of rewards and transitions generated by other agents or policies. The strength of this approach lies in the vast amount of historical data available today across many domains. One clear advantage over traditional online RL is that using static datasets avoids costly and potentially risky exploration in real environments, as well as the need to build accurate simulators, which can be challenging. However, this versatility also introduces difficulties, chiefly due to distributional mismatch between the historical data and the states encountered by the optimal policy [42]. Batch RL has seen notable success in automated trading systems, where algorithms like Fitted Q-Iteration (FQI) are leveraged to learn effective trading strategies from existing market data [22]. In this work, we will only focus on the former two cases, even if generalizing our results to a Batch RL scenario could be of great interest.

### 2.3.2. Model-based, Value-based and Policy-based Methods

After focusing on the different assumptions behind the agent environment interaction, in this section we describe the three main families of algorithms that are used in RL. Reinforcement Learning (RL) methods can be broadly divided into three main categories: model-based, model-free, and policy gradient approaches.

**Model-based methods** rely on learning or using a representation of the environment's dynamics to plan optimal actions. Once the model in known, the problem reduces to a pure planning task. In the generative model setting, where the agent can query for arbitrary state-action pairs, these approaches reduce to a regression problem with active samples: for every $h = 1, \dots H$ the agent chooses pairs $(s^{(i)}, a^{(i)})$, sequentially, observes one sample for the next state $s'^{(i)} \sim p_h(\cdot|s^{(i)}, a^{(i)})$ and uses the sample to train an approximate transition model $\widehat{p}_h$, that is then used in the planning phase. Model-based Reinforcement Learning approaches become more involved in online interaction scenarios, where the agent must continuously update its model while interacting with the environment. In the online setting, this procedure gets more complex, as the agent has less control on the state and action sequences it observes. Important examples of model-based RL methods include PETS (Probabilistic Ensembles with Trajectory Sampling, see [28]), which uses ensemble deep dynamics models for uncertainty-aware planning and has shown strong performance in continuous control tasks. Another well-known algorithm of this family is Dyna [123], which learns the transition model from real experience and then uses simulated experiences to improve the policy. For this reason, it is often used as a sample-augmentation technique for value-based or policy-based methods. Dyna has been successfully applied in various domains like robotics [100] and routing optimization [6].

**Value-based methods** focus on estimating the optimal state-action value function $Q_h^*(s, a)$. Once this is done, the optimal policy is just greedy with respect to $Q_h^*$. Classic algorithms in this category include SARSA and Q-learning. SARSA [139] updates action values based on the actual action taken in the next state, making it an on-policy method that learns the value of the policy being followed. Q-learning [144], instead, is an off-policy method that updates action values based on the best possible next action, allowing it to learn an optimal policy regardless of the agent's current behavior. Both algorithms are *tabular*, in the sense that they store the estimated value of $Q_h^*(s, a)$ for each state-action pair separately. More advanced and successful value-based approaches bypass this limitation using neural function approximation. A prominent example is the Deep Q-Network (DQN, see [88]), which uses neural networks to approximate the Q-function,

enabling value-based RL to scale to high-dimensional state spaces such as raw pixel inputs. DQN achieved remarkable success in domains like playing Atari games directly from screen pixels [89], illustrating the power of combining value-based methods with deep learning.

**Policy-based methods** represent one of the major families of algorithms that have shaped the history of Reinforcement Learning. In complex problems, where approximating the value function is infeasible, these approaches aim to learn the optimal policy directly by maximizing expected rewards.

The vast majority of these algorithms belong to the policy-gradient family. These optimize a parametric policy by estimating the gradient of the expected return with respect to the policy parameters, and then updating the parameters using gradient descent or its variants. The first and most straightforward algorithm in this family is REINFORCE [148], which uses Monte Carlo estimates of returns to compute the policy gradient and update the parameters after each episode. While conceptually simple, REINFORCE suffers from high variance in gradient estimates, which can slow down learning. Subsequent methods addressed these limitations. Notably, Trust Region Policy Optimization (TRPO, see [111]) introduced a constrained optimization framework to ensure updates stay within a "trust region," improving stability, while Proximal Policy Optimization (PPO, see [112]) simplified this approach with clipped objective functions, achieving similar stability with greater computational efficiency.

Policy-gradient methods such as TRPO and PPO have achieved significant milestones in real-world and simulated environments, including the aforementioned agent with superhuman performances in Dota 2 [19]. These three approaches are not mutually exclusive. For example, besides the previously mentioned Dyna method, recent advances have been achieved with Actor-Critic algorithms. These methods optimize a policy using policy-gradient techniques while employing an estimate of the $Q_h^*$ function to reduce gradient variance. Actor-Critic methods lie at the intersection of value-based and policy-based approaches. Among them, DDPG [72], SAC [48], and TD3 [44] are particularly notable.

After this overview of the variety of Reinforcement Learning algorithms developed in practice, the next chapters will focus on RL algorithms that come with theoretical guarantees of convergence and performance. Although the perspective differs significantly, the distinctions in settings and types of algorithms persist when moving from applied RL to theoretical RL, which is the focus of this thesis.

# 3 | Foundations of RL Theory

In this chapter, we transition from a heuristic description of Reinforcement Learning objectives toward a rigorous mathematical formalization. We begin by defining the core theoretical framework before examining established results for specific families of MDPs. To provide a solid foundation for these developments, we first formalize the assumptions underlying the analyses presented both in this and in subsequent chapters. A summary of the notation introduced and used in this chapter can be found in the appendix, table A.2.

## 3.1. Standard Assumptions

In this section, we define the standard assumptions underlying most results in RL theory. These will be invoked very frequently in the statements of most of the theorems.

**Assumption 6** (Bounded reward function)**.** *For any $s \in \mathcal{S}$ and $a \in \mathcal{A}$, and for any $h \in [H]$, the rewards function satisfies*

$$r_h(s,a) \in [0,1].$$

This assumption ensures that the reward function always belongs to a given interval, in this case $[0,1]$. Note that the choice of these two extremes is arbitrary, and in fact the following results can easily be generalized to any reward range. From assumption 6 it follows that for each policy and each $s \in \mathcal{S}$ and $a \in \mathcal{A}$, and for any $h \in [H]$

$$0 \leq V_h^\pi(s) \leq H \qquad 0 \leq Q_h^\pi(s,a) \leq H.$$

This consequence, which, as assumption 6 is standard in most of the literature, allows us to say that, whatever policy the agent plays during the $K$ learning episodes, its value function cannot be more than $H$ away from the optimal one.

In addition to this bound on the reward function, we also need to ensure that the stochastic reward, which we have called $y_h$, does not differ too much from its mean. If $y_h$ followed a distribution that was too heavy-tailed, a huge number of samples would be needed to

learn the reward function. Most of the RL literature overcomes this problem by assuming that the reward is sub-Gaussian.

**Assumption 7** (Sub-Gaussian noise). *Let $s_1, a_1, \dots s_h, a_h, \dots s_H, a_H$ be the sequence of states and actions for one episode, and $y_1, \dots y_h, \dots y_H$ the corresponding rewards. For any any $h \in [H]$, the random variables $y_h - r_h(s_h, a_h)$ is $\sigma-$sub-Gaussian conditionally on $s_h, a_h$. This means that*

$$\forall \lambda \in \mathbb{R} \qquad \mathbb{E}[\exp(\lambda(y_h - r_h(s_h, a_h)))|s_h, a_h] \leq \exp\left(\frac{\sigma^2\lambda^2}{2}\right),$$

*for $\sigma = 1$ (unless otherwise specified).*

While it may not be clear at first glance, from assumption 7 it follows that $\mathbb{E}[y_h|s_h, a_h] = r_h(s_h, a_h)$ as a general property of sub-Gaussian distributions. Moreover, the previous assumption implies a bound on the decay of the tails of the $y_h - r_h(s_h, a_h)$ distribution. In fact, it follows as a general property of sub-Gaussian variables that for any $\varepsilon > 0$

$$\mathbb{P}\Big(|y_h - r_h(s_h, a_h)| > \varepsilon \Big| s_h, a_h\Big) \leq 2\exp\left(-\frac{\varepsilon^2}{2\sigma^2}\right).$$

The former result can be extended recalling Markov property. In fact, $y_h - r_h(s_h, a_h)$ is not only sub-Gaussian conditionally on $s_h, a_h$, but on the entire sequence of previous states and actions. By the Markov property, $\mathbb{E}[\exp(\lambda(y_h - r_h(s_h, a_h)))|s_1, a_1, \dots s_h, a_h]$ corresponds to the same expected value conditioning only on $s_h, a_h$.

For sub-Gaussian distributions, some powerful theorems known as *concentration inequalities* hold. These are the backbone of many results in Learning Theory.

**Proposition 8** (Hoeffding's inequality). *Fix $\delta > 0$. Let $X_1, \dots X_n$ be random variables such that for any $t \in [n]$,*

$$\mathbb{E}[\exp(\lambda X_t)|X_1, \dots X_{t-1}] \leq \exp\left(\frac{\sigma^2\lambda^2}{2}\right).$$

*Then, with probability at least $1 - \delta$,*

$$\left|\sum_{t=1}^{n} X_t\right| \leq \sqrt{2n\log(2/\delta)}\sigma.$$

The former result has multiple extensions. In particular, this theorem can be generalized to vector-valued processes ([68], theorem 20.4).

## 3.2. Performance Metrics

While it is clear that a policy $\pi$ is optimal if, at any $h \in [H]$ we have $Q_h^\pi(s,a) = Q_h^*(s,a)$, for any $(s,a)$ pair, we have not yet defined formally what an $\varepsilon-$optimal policy is. From this point on, we call

$$J^\pi := \int_{\mathcal{S}} V_1^\pi(s) p_1(s) \mathrm{d}s. \tag{3.1}$$

$J^\pi$ gives what is often called the *return* of policy $\pi$, that is the total expected reward $V_1^\pi(s)$ from an initial state $s \in \mathcal{S}$ averaged over the distribution $p_1(\cdot)$ of the initial state itself. Building upon this notion, we can formally define an $\varepsilon-$optimal policy.

**Definition 4** ($\varepsilon-$optimality). *Let $\varepsilon > 0$. A policy $\pi$ is called $\varepsilon-$optimal if $J^* - J^\pi \leq \varepsilon$, where*

$$J^* := \int_{\mathcal{S}} V_1^*(s) p_1(s)\, ds,$$

*is the return of the optimal policy.*

Obviously, any optimal policy is also $\varepsilon-$optimal for any $\varepsilon > 0$. Now, as we have anticipated, the goal of Reinforcement Learning is to find an algorithm $\mathscr{A}$ that interacts with the environment for a number of stages $n$ and if able to output some policies that get closer to the optimal the larger it receives. This may be formalized in (at least) two different way, depending on what we have to enhance in the learning procedure.

### 3.2.1. Statistical Complexity

In the statistical complexity view, the performance metrics is defined as follows.

**Definition 5** (Sample complexity). *Let $\mathscr{M}$ be a class of MDPs. For $\varepsilon, \delta > 0$, the $(\varepsilon, \delta)-$sample complexity of an algorithm $\mathscr{A}$ is the number $n$ of interactions with an MDP in $\mathscr{M}$ that $\mathscr{A}$ needs to output an $\varepsilon-$optimal policy with probability at least $1-\delta$. We note this number as*

$$\mathit{SampleComplexity}(\mathscr{A}|\mathscr{M}, \varepsilon, \delta).$$

To simplify the notation, when the values of $\varepsilon, \delta$ are fixed and do not create ambiguity we will simply write SampleComplexity$(\mathscr{A}|\mathscr{M})$.

This performance metric depends both on the MDP class and the algorithm $\mathscr{A}$. Fixing the algorithm, this metric is monotone in the MDP class, meaning that, when-

ever we have two classes such that $\mathscr{M} \subset \mathscr{M}'$, then SampleComplexity$(\mathscr{A}|\mathscr{M}, \varepsilon, \delta) \leq$ SampleComplexity$(\mathscr{A}|\mathscr{M}', \varepsilon, \delta)$ for fixed $\varepsilon$ and $\delta$. On the opposite side, it is clear that the sample complexity grows when $\varepsilon$ and $\delta$ decrease, as more samples are needed to achieve a better accuracy.

Sample complexity only captures the *statistical complexity* of the problem, i.e. how complex it is to identify the optimal policy of one MDP from $n$ interactions while only knowing that it belongs to class $\mathscr{M}$. In case $\mathscr{M}$ reduces to just one element, the algorithm that outputs $\pi^*$ for the corresponding MDP has a sample complexity of 0. This is because this notion does not relate to how complex it is the optimal policy, but more on how hard is to learn from a given class $\mathscr{M}$. In fact, we can characterize the hardness of the class itself as

$$\text{MinMaxComplexity}(\mathscr{M}, \varepsilon, \delta) = \inf_{\mathscr{A}} \text{SampleComplexity}(\mathscr{A}|\mathscr{M}, \varepsilon, \delta).$$

In definition 5, the type of samples we gather is not specified, as it depends on the type of agent-environment interaction that we assume in every specific case (see section 2.3.1). When assuming a generative model, the $n$ samples correspond to $n$ queries to the model. Instead, when we assume online interaction, the samples must be collected during episodes of fixed length. Therefore, in such case we have $n = HK$ samples, with exactly $K$ samples for every stage $h = 1, \dots H$ of the MDP.

The sample complexity we have defined fits into the more general context of PAC (Provably Approximately Correct see [60]) bounds in machine learning. Giving a PAC bound for an algorithm $\mathscr{A}$ on an Hypothesis class $\mathscr{H}$ with respect to a loss function $\mathscr{L}$ corresponds to proving that

$$\mathbb{P}(\mathscr{L}(\mathrm{h}_n) - \mathscr{L}(\mathrm{h}_\star) \geq \varepsilon) \leq \delta,$$

where $\mathrm{h}_n$ is the hypothesis returned by $\mathscr{A}$ trained with $n$ samples, and $\mathrm{h}_\star$ is the minimizer of the loss. The analogy with our sample complexity is clear if we consider, as hypothesis space, the space of all policies that are optimal for an MDP of class $\mathscr{M}$ and, as loss

$$\mathscr{L}(\widehat{\pi}_n) = -J^{\widehat{\pi}_n} \qquad \mathscr{L}(\widehat{\pi}_n) - \mathscr{L}(\pi^*) = J^* - J^{\widehat{\pi}_n}.$$

Precisely because it is borrowed from general machine learning, which is traditionally offline, this performance metric does not account for suboptimality during the learning process. In the next section, we will look at a measure that's more suited to the online setting.

### 3.2.2. No-regret Learning

In eq. (2.1), we have defined the regret in the bandit setting. As we pointed out, bandits correspond to MDPs where $\mathcal{S} = \{1\}$ (one single state) and $H = 1$. In such case, the only thing that matters is the chosen action, so that the state-action value function collapses to the reward itself. We generalize the regret to RL in the following way.

$$R_K := \sum_{k=1}^{K} V_1^*(s_1^k) - V_1^{\pi^k}(s_1^k), \tag{3.2}$$

where $\pi_k$ is the policy played by the algorithm at episode $k$. Given this definition, the regret is specifically designed assuming online interactions, and does not trivially generalize to the generative model/offline learning frameworks. This notion allows to define our second performance metric.

**Definition 6** (Regret bound). *Let $\mathscr{M}$ be a class of MDPs. For $\delta > 0, K \in \mathbb{N}$, the regret bound of algorithm $\mathscr{A}$ for $K$ episodes with probability $\delta$ is the minumum value such that, for every $\mathcal{M} \in \mathscr{M}$, the following equation holds*

$$\mathbb{P}\left(R_K > RegretBound(\mathscr{A}|\mathscr{M}, K, \delta)\right) \leq \delta.$$

Also here, to simplify the notation, when the values of $K, \delta$ are fixed and do not create ambiguity we will simply write RegretBound$(\mathscr{A}|\mathscr{M})$.

Like the sample complexity, the regret bound of an algorithm depends both on $\delta$ and $K$ and on some features of the MDP class. The big difference with respect to the previous performance metric is that the regret does not only evaluate the final policy that the agent is able to learn. Conversely, to achieve a sublinear regret it is necessary that $V_1^{\pi^k}(s_1^k) \to V_1^*(s_1^k)$ with $k \to \infty$. This poses an additional challenge: on one side, the policy $\pi^k$ must explore, to gather new samples for states that were visited the less. On the other side, the same policy must have a high return $J^{\pi^k}$ (that corresponds to $\mathbb{E}[V_1^{\pi^k}(s_1^k)]$), otherwise it is not possible to bound the regret.

Under our assumption 6, the regret of any algorithm on any MDP is bounded by $HK$. Indeed, having normalized the reward between zero and one, every policy enjoys a value function $V_1^\pi$ in the range $[0, H]$. Therefore,

$$R_K := \sum_{k=1}^{K} V_1^*(s_1^k) - V_1^{\pi^k}(s_1^k) \leq \sum_{k=1}^{K} H \leq HK.$$

A regret that grows linearly in $K$ is generally considered unsatisfactory, as it implies that the return of policy $\pi^k$ is not converging to the optimal return. On the other side, if the regret grows with $K$ at a rate that is sublinear, for fixed $\delta$, we can deduce a sample complexity guarantee for the same algorithm, as the next result shows.

**Proposition 9.** *Assume* $\mathit{RegretBound}(\mathscr{A}|\mathscr{M}, K, \delta) \leq C(\mathscr{M}, \delta)K^\alpha$, *for some* $\alpha < 1$, *where* $\mathscr{A}$ *is an algorithm and* $\mathscr{M}$ *an MDP class. Defined*

$$\overline{\pi}^K(\cdot|s) := K^{-1}\sum_{k=1}^{K}\pi^k(\cdot|s)$$

*the average policy played by* $\mathscr{A}$ *on all the episodes. Then, outputting* $\overline{\pi}^K(\cdot|s)$ *at the end of the* $K$ *episodes of interaction,* $\mathscr{A}$ *enjoys a sample complexity bound of the following form*

$$\mathit{SampleComplexity}(\mathscr{A}|\mathscr{M}, \varepsilon, \delta) \leq \left\lceil H\left(\frac{2C(\mathscr{M}, \delta/2)}{\varepsilon}\right)^{\frac{1}{1-\alpha}} + \frac{32H^3\log(4/\delta)}{\varepsilon^2}\right\rceil.$$

Proposition 9 allows us to transition from a bound on the regret to one on the sample complexity. In the field of theoretical Reinforcement Learning (RL), it is proven that for all non-trivial classes of MDPs, the regret must grow at least as $\mathcal{O}(\sqrt{K})$, at least within our instance-independent formulation[1]. Therefore, in the result of proposition 9, the first term is always the dominating one. The whole result may be written in a relatively gross way as

$$\text{RegretBound}(\mathscr{A}|\mathscr{M}, K, \delta) \leq \mathcal{O}(K^\alpha) \implies \text{SampleComplexity}(\mathscr{A}|\mathscr{M}, \varepsilon, \delta) \leq \mathcal{O}(\varepsilon^{\frac{1}{1-\alpha}}).$$

Observing this equation, it is very clear how a very weak regret bound, i.e. with $\alpha \approx 1$, corresponds to a very weak sample complexity guarantee, with $1/\varepsilon$ raised to a power that goes to $+\infty$. At this point, it is natural to ask if the relationship between regret bounds and sample complexity guarantees goes also the other way round, i.e. we can derive regret bounds from sample complexity guarantees. To prove this is not trivial, as the sample complexity guarantees do not deal with the dilemma of exploration vs exploitation, that is crucial in achieving regret bounds. Nonetheless, there is a procedure to turn these guarantees into regret bounds, which takes the name of *Explore then commit* [45]. This very simple strategy works as follows.

[1]There is, in fact, a body of literature that studies instance-dependent regret bounds, where the bound is allowed to depend on characteristics of the specific MDP $\mathcal{M}$ and not only on the class $\mathscr{M}$. Regret bounds of this type are often able to scale as $\mathcal{O}(\log K)$ (see e.g., [127, 128]), but in this work we focus on the instance-independent case, where this is not possible.

1. For $K_{\text{explore}} < K$ episodes run algorithm $\mathscr{A}$, with sample complexity guarantee, getting a policy $\widehat{\pi}$.
2. For the remaining $K - K_{\text{explore}}$ play $\widehat{\pi}$, without updating it.

Using this strategy, the agent suffers linear regret in the first part, as $\mathscr{A}$ is not designed to do any exploitation. We can still bound this part of the regret as $HK_{\text{explore}} = \mathcal{O}(K)$. For the second part, assuming that the algorithm has a sample complexity guarantee of order $\mathcal{O}(\varepsilon^{-\beta})$ it follows that, after $K_{\text{explore}}$ episodes, $\widehat{\pi}$ is $\varepsilon-$optimal with $\varepsilon = K_{\text{explore}}^{-1/\beta}$. At this point, the regret corresponds to $\mathcal{O}((K - K_{\text{explore}})K_{\text{explore}}^{-1/\beta})$, and optimizing the value of $K_{\text{explore}}$ gives

$$K_{\text{explore}} \approx K^{\frac{\beta}{\beta+1}}, \qquad R_K \leq \mathcal{O}(K^{\frac{\beta}{\beta+1}}).$$

In this way we can turn sample complexity guarantees into regret bounds. The fact that with this procedure we get a sublinear regret bound (since $\beta/(\beta+1) < 1$) does not mean that the whole procedure is tight. In fact, if we try to transform a regret bound into a sample complexity guarantee with proposition 9 and then back to a regret bound with the explore-then-commit strategy, we get

$$\text{RegretBound } \mathcal{O}(K^{\alpha}) \implies \text{SampleComplexity } \mathcal{O}(\varepsilon^{\frac{1}{1-\alpha}}) \implies \text{RegretBound } \mathcal{O}(K^{\frac{1}{2-\alpha}}),$$

which is not tight as $\alpha < \frac{1}{2-\alpha}$ for every $\alpha \in (0, 1)$.

**Regret bounds in expectation or in high probability** In the previous definition 6, we formulated the bound for a fixed probability parameter $\delta$. At this point, it is natural to ask what happens if, instead of focusing on that quantity, we take the expected value of the regret, that is, the expected regret defined in the section on bandits, which writes as $\overline{R}_K = \mathbb{E}[R_K]$. Fortunately, under our assumptions, obtaining bounds in expectation from our high-probability bounds is straightforward. Indeed, it can be shown that, under assumption 7, the types of regret bounds we can prove always take the form

$$\text{RegretBound}(\mathscr{A}|\mathscr{M}, K, \delta) \leq \text{C}(\mathscr{M}, K)\sqrt{\log(1/\delta)}.$$

From this, one immediately deduces a tail bound for the distribution of the regret, which satisfies, for all $t \geq 0$,

$$\mathbb{P}(R_K \geq t) \leq \exp(-(t/\text{C}(\mathscr{M}, K))^2).$$

Thus, by the layer cake representation,

$$\begin{aligned}\overline{R}_K &= \int_0^\infty \mathbb{P}(R_K \geq t)\mathrm{d}t \leq \int_0^\infty \exp(-(t/\mathrm{C}(\mathscr{M}, K))^2)\mathrm{d}t \\ &= \mathrm{C}(\mathscr{M}, K)\int_0^\infty \exp(-\tau^2)\mathrm{d}\tau = \frac{\sqrt{\pi}}{2}\mathrm{C}(\mathscr{M}, K).\end{aligned}$$

As $\frac{\sqrt{\pi}}{2} \approx 1$, the previous passages ensure that a regret bound like $\mathrm{C}(\mathscr{M}, K)\sqrt{\log(1/\delta)}$ automatically allows for a bound on the expected regret of the same order in all the parameters.

**Any-time regret bounds** As we have clarified, all the regret bounds we will prove assume that the agent knows in advance the family of $\mathscr{M}$ that it is dealing with, but not the specific instance $\mathcal{M}$. The number of stages $H$ per episode, that we include as a parameter of $\mathscr{M}$ can be assumed to be known, as otherwise the agent only needs to wait for one episode to reveal it. Instead, we did not say anything on the knowledge of $K$. In fact, algorithms that do not assume knowledge of $K$ are formally more general, and take the name of *any-time*. Nonetheless, assuming that the agent knows $K$ is not restrictive. Indeed, a standard argument known as *doubling trick* [21] allows to convert any algorithm in an any-time algorithm. This doubling trick means, roughly, to run an algorithm $\mathscr{A}$ for batches of episodes of length $[1, 2, 4, \dots 2^\tau, \dots]$ until $K$ is reached. From the fact that the number of batches does not exceed $\lceil \log(K) \rceil$, it follows that the regret for this procedure does not exceed the one of $\mathscr{A}$ times logarithmic factors. For this reason, in the rest of this work we will always assume that the agent has access to $K$, as the regret bound generalize to the any-time case with the same order, except for logarithmic terms.

## 3.3. Fundamental Results in Reinforcement Learning Theory

In this section, we present the most celebrated results in the theoretical reinforcement learning (RL) literature. These theorems are organized according to different classes of Markov Decision Processes (MDPs), which we will define in detail later. Our focus will be on the theoretical guarantees obtained for these classes, in terms of regret bounds and sample complexity.

A detailed description of the algorithms achieving these guarantees would be beyond the scope of this introductory overview. Neveryheless, it is worth emphasizing that the underlying principle behind all these algorithms is conceptually unified and is known as

the optimism in the face of uncertainty (OFU) paradigm, that we introduce next.

### 3.3.1. The OFU Principle

At the heart of most reinforcement learning algorithms that achieve sublinear regret bounds lies the Optimism in the Face of Uncertainty (OFU) principle. The idea is simple yet powerful: when uncertain about the true dynamics or rewards of the environment, the agent should act as if the most favorable plausible scenario were true. By deliberately exploring actions that could potentially yield higher rewards, the agent finds itself in a win-win situation. Either its guess was correct, in such case it plays a near-optimal policy, or it was wrong and in such cases it receives the samples to refine its knowledge of the environment. This optimistic behavior ensures that the agent gathers enough information to reduce uncertainty over time, ultimately leading to near-optimal long-term performance.

To formalize this principle mathematically, it is convenient to start from the multi-armed bandit case, which, as previously discussed, can be seen as a special instance of an MDP with $\mathcal{S} = \{0\}$, $\mathcal{A} = \{1, \dots, A\}$, and horizon $H = 1$. The first formal application of the OFU principle appeared in the seminal paper by [10], where it was employed to derive optimal regret bounds for multi-armed bandits.

Consider the multi-armed bandit problem with arm space $\mathcal{A}$. Let us fix $a \in \mathcal{A}$, and call $k_t$ for $t = 1, \dots n_K(a)$ the steps at which arm $a$ was pulled. Under Assumption 7, the observed rewards $y_{k_t}$ satisfy that $y_{k_t} - r(a)$ is $\sigma$-sub-Gaussian. From Hoeffding's inequality (proposition 8), and a union bound over the time-steps, it follows that

$$\mathbb{P}\left(\exists n \leq K \quad \sum_{t=1}^{n} y_{k_t} + \sqrt{2n\log(K/\delta)}\sigma \leq r(a)\right) \leq \delta.$$

As, at any step $k$, the number of times $n_k(a)$ where $a$ has been pulled satisfy $n_k(a) \leq K$, the previous equation implies that

$$\mathbb{P}\left(\exists k \leq K, \quad \frac{\sum_{t=1}^{n_k(a)} y_{k_t}}{n_k(a)} + \sqrt{\frac{2\log(K/\delta)}{n_k(a)}} \leq r(a)\right) \leq \delta.$$

The first part of this equation, $\frac{\sum_{t=1}^{n_k(a)} y_{k_t}}{n_k(a)}$ corresponds to nothing else than the sample mean estimator of the reward $r(a)$, which we can call $\widehat{r}_k(a)$. In this view, the previous result ensures that adding a term $\sqrt{\frac{2\log(2K/\delta)}{n_k(a)}}$ to the sample mean, we get an estimator that

overestimates $r(a)$ at all the steps with probability at least $1-\delta$. As there are a total of $A$ arms, and we need the confidence bound to hold for each of them simultaneously, we need the error probability of each arm to be of $\delta/A$. This modifies the confidence radius as $\sqrt{\frac{2\log(2AK/\delta)}{n_k(a)}}$.

The UCB1 algorithm [10] leverages this result to build the first algorithm with a regret bound for the MAB problem. UCB1 repeats these two operations at any step $k$

1. Compute $\text{UCB}_k(a) := \widehat{r}_k(a) + \sqrt{\frac{2\log(2AK/\delta)}{n_k(a)}}$ for every $a \in \mathcal{A}$
2. Pull $a_k := \arg\max_{a\in\mathcal{A}} \text{UCB}_k(a)$ and collect $y_k$.

These simple passages are sufficient to enjoy the following regret guarantee

**Theorem 10** ([68] thm 7.2)**.** *Consider a MAB problem such that assumptions 6 and 7 (with $\sigma = 1$) hold. Then, with probability at least $1-\delta$, the regret of UCB1 is bounded by*

$$R_K \leq 8\sqrt{AK\log(AK/\delta)} + 3A.$$

As the regret is always bounded by $K$, by assumption 6, the lower order term can be rewritten as $\min\{A, K\} \leq \sqrt{AK}$. Thus, this theorem shows that the regret is bounded as $\widetilde{\mathcal{O}}(\sqrt{AK})$, a rate that can be shown to be optimal for MABs. In the next section, we will investigate if the same approach can be generalized to RL.

## Optimism in RL

Applying this optimism principle in RL is more challenging than in bandits. In the latter case, the agent is only concerned with maximizing the reward function, and the only reasonable way to enforce optimism on a function that we want to maximize, is via an upper bound. In RL the situation is more involved, due to the fact that the agent needs to learn the transition function $p_h$ at each time step, and there is no clear way to write an optimistic estimator for a probability distribution. The first solution that was invented for this problem was to estimate $p_h$ with a counting estimator $\widehat{p}_h$, to use the an bonus on the sample mean estimator $\widehat{r}_h$ (as for bandits), and to propagate the optimism over all states using the Value Iteration procedure. Assuming we have a finite number $S$ of states and of actions $A$, let us call, at episode $k \in [K]$,

$$\widehat{p}_h^k(s'|s,a) = \frac{n_h^k(s_h = s, a_h = a, s_{h+1} = s')}{n_h^k(s_h = s, a_h = a)} \qquad \widehat{r}_h^k(s,a) = \frac{\sum_{t=1}^{n_h^k(s,a)} y_{k_t}}{n_h^k(s,a)}.$$

At this point, the agent can add a bonus $b_h^k(s,a)$ to the reward, which depends on $n_h^k(s,a)$ and propagate it through the MDP with VI:

$$\widehat{Q}_{H+1}^k(s,a) = 0 \qquad \widehat{Q}_h^k(s,a) := \widehat{r}_h^k(s,a) + b_h^k(s,a) + \sum_{s' \in \mathcal{S}} \widehat{p}_h^k(s'|s,a)\widehat{V}_{h+1}^k(s') \tag{3.3}$$

where $V_h^k(s) = \max_{a \in \mathcal{A}} \widehat{Q}_h^k(s,a)$. For a proper choice of this bonus, this function is $\geq Q_h^*(s,a)$ at for any state-action pair and any stage $h$. This choice of $b_h^k(s,a)$ does not correspond to $\sqrt{\frac{2\log(2AK/\delta)}{n_h^k(s,a)}}$, which would be the direct analog to the bonus we have seen for UCB1. Instead, to guarantee optimism it is necessary to scale it with an additional factor $H$.

Once optimism is guaranteed, the agent can, at each episode, compute $\widehat{Q}_h^k(s,a)$, derive the corresponding greedy policy $\pi_h^k(s) = \arg\max_{a \in \mathcal{A}} \widehat{Q}_h^k(s,a)$ and use it to play the current episode, gathering new data and restarting this routine with $\widehat{Q}_h^{k+1}(s,a)$. This idea gave rise to the celebrated UCRL2 algorithm (see [12]), which in our setting is better known as UCBVI[2]. It can be shown that this algorithm enjoys a regret guarantee, while not optimal, for the Markov class of MDPs with a finite number of states and action. This class, known as *Tabular MDPs*, is the first that we study in the next sections.

### 3.3.2. Tabular MDPs

In this section, we define the class of Tabular MDPs, as discuss the algorithms achieving the best theoretical guarantees for this class of processes. In the following, we are starting from the more difficult case, assuming **Online Interaction** and requiring **Regret bounds** type of guarantees. Then, in the second part of this section we are focusing on the easier **Generative Model** interaction and on **Sample Complexity Guarantees**.

**Definition 7** (Tabular MDPs). *We define Tabular MDPs the class $\mathscr{M}$ of MDPs where $\mathcal{S}, \mathcal{A}$ are discrete sets of sizes $S, A$ respectively.*

The literature outlining the analysis of provably efficient methods for this class of MDPs is immense, both in a *model-based* and a *value-based* context. The first regret bound for this class was proved for the previously mentioned UCBVI algorithm, which is model-based, as it estimates the transition probability.

[2]Remarkably, that paper shows a regret guarantee even for an harder scenario than the one that we are studying in this thesis, that is when the MDP has $H = +\infty$ but finite diameter. The former condition, allows learning even in case of infinite episode horizon, as it forces every policy to keep visiting the same state infinitely often. For a precise definition of the diameter of an MDP, see the original paper [12] or the chapter *Markov Decision Processes* of [68].

**Theorem 11** (Theorem 7.1 in [4]). *Let $\mathscr{M}$ the class of Tabular MDPs, assuming online interaction and assumptions 6 and 7. The UCBVI algorithm enjoys*

$$\textit{RegretBound}(\textit{UCBVI}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^2 S\sqrt{AK}\right).$$

In this result, as we are often going to do, we are using the $\widetilde{\mathcal{O}}$ notation to ignore constants and terms that are logarithmic in the feature of the class and $K$. as we anticipated, the regret bound does not depend on $\delta$: in fact, the error probability appears as a multiplicative term $\sqrt{\log(1/\delta)}$, that is hidden in the $\widetilde{\mathcal{O}}$ notation. This dependence follows directly from the assumption of sub-gaussian distribution of the noise 7, and is the same that we are going to find in all the regret bounds of this thesis. The previous result maintains the same order of growth of theorem 10 in terms of $A, K$. As that theorem was tight for the MAB problems, which are a subclass of Tabular MDPs for $S = \{0\}, H = 1$, the former result is also tight in $A, K$. Is it also tight in the other variables, $S$ and $H$? Not really, as the following theorem shows.

**Theorem 12** (Theorem 3 in [53]). *Let $\mathscr{M}$ the class of Tabular MDPs, assuming online interaction and assumptions 6 and 7. For any algorithm $\mathscr{A}$,*

$$\textit{RegretBound}(\mathscr{A}|\mathscr{M}, K, \delta) \geq \Omega\left(\sqrt{H^3 SAK}\right).$$

Comparing theorem 12 with theorem 11, we observe that there are two sources of suboptimality. The exponent of $H$, which is $H^2$ in the upper bound and $H^{3/2}$ in the lower, and $S$, which appears as $S$ and $S^{1/2}$ in the two cases, respectively.

It is not trivial to determine if the fact that theorem 11 does not achieve the optimal regret guarantee is related to the algorithm UCBVI or to its analysis. In this case, it came out that *both* the following hypotheses are true: the regret bound was not optimal for UCBVI, and the algorithm itself was not able to achieve optimal regret guarantees. On one side, with much more work, it was proved (see Theorem 7.6 in [4]) that the same algorithm enjoys

$$\text{RegretBound}(\text{UCBVI}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^2\sqrt{SAK} + H^3 S^2 A\right).$$

This new bound on the regret assumes a lower order in the variable $S$, which decreases to $\sqrt{S}$, while remaining suboptimal in terms of $H$. This is considering only the dominant term, which scales with $\sqrt{K}$: unlike before, regret contains a particularly massive constant

whose scale reaches $H^3S^2A$.

As we mentioned, regardless of the analysis, the performance of UCBVI, is intrinsically sub-optimal. A way to improve it, is to replace the Hoeffding-type bonus that we used with a Bernstein-type one, which is tighter. The resulting algorithm takes the name of UCBVI-BF (see [15] for the original paper and [36] for a refined analysis) and enjoys the following result.

**Theorem 13.** *Let $\mathscr{M}$ the class of Tabular MDPs, assuming online interaction and assumptions 6 and 7. The UCBVI-BF algorithm enjoys*

$$\textit{RegretBound}(\textit{UCBVI-BF}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(\sqrt{H^3SAK} + H^3S^2A\right).$$

For large $K$, the dominant term $\sqrt{H^3SAK}$ corresponds to the lower bound provided in theorem 12. Still, due to the presence of the lower-order term, this only happens for $K \geq \Omega(H^3S^3A)$, which can be very big in many scenarios. Even a very recent paper [36] which studies the regret of UCBVI-BF could improve the previous regret bound only of logarithmic and constant terms, which suggests that the this undesirable constant term is intrinsic in the algorithm design.

Only very recently [153] was able to find an algorithm whose regret bound gets rid of the lower-order term, achieving the lower bound in theorem 12 for any value of $K$. This is algorithm is known as MVP (monotonic value propagation) and exploits a principle similar to the doubling trick (see [21]).

**Theorem 14** (Theorem 1 from [153])**.** *Let $\mathscr{M}$ the class of Tabular MDPs, assuming online interaction and assumptions 6 and 7. The MVP algorithm enjoys*

$$\textit{RegretBound}(\textit{MVP}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(\sqrt{H^3SAK}\right).$$

This result represents the state of the art for the regret bounds on Tabular MDPs with a model-based approach.

**value-based algorithms** represent a minority of the literature for Tabular MDPs. Nonetheless it is worth mentioning that an optimistic version of Q-Learning, perhaps the most classical and fundamental RL algorithm [124], enjoys a regret guarantee on Tabular MDPs. This algorithm, invented by [53] as Q-Learning UCB-B, merges the ideas of the original Q-Learning with the Bernstein-style bonus which allowed to improve the regret of UCBVI.

**Theorem 15** (Theorem 2 from [53]). *Let $\mathscr{M}$ the class of Tabular MDPs, assuming online interaction and assumptions 6 and 7. The Q-Learning UCB-B algorithm enjoys*

$$\textit{RegretBound}(\textit{Q-Learning UCB-B}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^2\sqrt{SAK} + \sqrt{H^9S^3A^3}\right).$$

This regret bound cannot be compared to result that are obtained for model-based algorithms. Both the dominating term is sub-optimal (extra $\sqrt{H}$ dependence) and a non-trivial lower order term $\sqrt{H^9S^3A^3}$ is present. Nonetheless, the Q-Learning UCB-B algorithm aligns more closely with empirical applications than methods such as UCBVI or MVP, which often prove computationally impractical for real-world scenarios. The fact that the regret is suboptimal reflects a well-known issue among practitioners of Q-learning-based algorithms, namely the *overestimation bias* (see [49, 66]). Although from a practical standpoint the overestimation bias has been widely addressed, for instance by the two algorithms mentioned above, an analogous method endowed with optimal theoretical guarantees does not yet exist. In recent times [70] invented a value-based algorithm named Q-ESA, which has a regret guarantee for order

$$\text{RegretBound}(\text{Q-ESA}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(\sqrt{H^3SAK} + \sqrt{H^6SA}\right).$$

The proposed method is way more involved than the original Q-Learning, and a sub-optimal lower order term is still present. Designing a variant of Q-Learning whose regret scales as $\widetilde{\mathcal{O}}(\sqrt{H^3SAK})$ (or proving that this is not possible) remains an open problem.

## Sample complexity

In this section, we focus on the other problem: to find a sample complexity guarantee for an algorithm while allwing access to a generative model. As this setting is less restrictive than the previous one, all the guarantees for the online interactions are kept for the generative model, while the converse is not true. We start from the lower bound for this setting.

**Theorem 16** (Theorem 7 from [35]). *Let $\mathscr{M}$ the class of Tabular MDPs, assuming access to a generative model. Then, any algorithm $\mathscr{A}$ enjoys*

$$\textit{SampleComplexity}(\mathscr{A}|\mathscr{M}) \geq \Omega\left(\frac{H^4SA}{\varepsilon^2}\right).$$

The presence of $H^4$ may surprise some readers familiar with the same result with $H^3$. The reason for this apparent mismatch is that the authors of [35] measure the sample

complexity in *episodes*, while we do in interaction steps. This means that, to adapt their result to our notation, multiplying by $H$, which is the number of steps in an episode, is necessary.

At this point, it would be natural to expect that the design of a specific algorithm that exploits a generative model at its full potential is necessary to achieve the former sample complexity guarantee. Surprisingly, we do not need to introduce any other algorithm. Applying proposition 9 onto the result of theorem 14 gives

$$\text{SampleComplexity}(\text{MVP}|\mathscr{M}, \varepsilon, \delta) \leq \widetilde{\mathcal{O}}\left(\frac{H^4 SA}{\varepsilon^2}\right).$$

The design of MVP, despite its stricter assumptions and different objective, automatically ensures a sample complexity bound that is optimal for tabular cases. In this sense, having access to a generative model does not substantially improve the type of guarantee that the agent is able to get. This should not imply that the generative model assumption is without merit; on the contrary, it significantly streamlines the analysis of the algorithms being studied. While the optimal sample complexity for algorithms under the generative model assumption had already been established in the early 2010s (see [14]), its online counterpart [153] was published many years later, highlighting the significantly higher complexity of the problem.

### 3.3.3. MDPs with Linear Function Approximation

In the previous section, we have defined the class of tabular MDP, showing the results for what concerns regret in the online setting and sample complexity under the generative model assumption. In this section, we will focus on a more general family of MDPs, which include arbitrary state and action spaces. The idea is to formalize the concept of MDPs that lend themselves to analysis using linear function approximation; however, this is less trivial than for the tabular case, and various definitions have been proposed in the literature. In this section, we will first define the two most commonly used notions, leaving the discussion of other similar classes of MDPs to section 3.3.4.

**Definition 8.** *[Linear MDPs, assumption A of [54]] We define Linear MDPs the class of MDPs endowed with a sequence of feature maps $\varphi_h : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ such that the following conditions hold.*

1. *For every $h \in [H]$ there is $\theta_h \in \mathbb{R}^d$ such that*

$$\forall s \in \mathcal{S}, a \in \mathcal{A} \qquad r_h(s,a) = \varphi_h(s,a)^\top \theta_h.$$

2. *For every $h \in [H]$ there is a function $\mu_h : \mathcal{S} \to \mathbb{R}^d$ such that*

$$\forall s, s' \in \mathcal{S}, a \in \mathcal{A} \qquad p_h(s'|s,a) = \varphi_h(s,a)^\top \mu_h(s'),$$

Compared to the previous notion of Tabular MDPs, many more element appear. Crucially, in the interaction this class of MDPs assumes that the agent has knowledge of all $\varphi_h$, but not of $\theta_h, \mu_h$. Linear MDPs are endowed with the following property. For every $h \in [H]$ and every $f : \mathcal{S} \to [0,1]$ there is $\omega_h \in \mathbb{R}^d$ such that

$$\forall s \in \mathcal{S}, a \in \mathcal{A} \qquad \int_{\mathcal{S}} f(s') p_h(s'|s,a) \mathrm{d}s' = \varphi_h(s,a)^\top \omega_h.$$

For the proof, see the appendix (proposition 52). In other words, the integrating over the transition function transforms bounded functions into functions that are linear in $\varphi_h$. This result entails (see Proposition 2.3. in [54]) that for any policy $\pi$ the corresponding $Q_h^\pi$ is also linear in the feature map, meaning that it corresponds to $\varphi_h(s,a)^\top \omega_h^\pi$ for some $\omega_h^\pi \in \mathbb{R}^d$. This result generalizes to the $Q_h^*$ function, which is also linear in the feature map.

This class of MDPs may seem completely different from tabular MDPs, since it does not impose any conditions on the spaces $\mathcal{S}$ and $\mathcal{A}$ (which can be infinite), but only on the existence of a feature map $\varphi$ with some properties. While this is true, the following result shows that, for a particular choice of this map, Tabular MDPs are actually a subfamily of the linear ones.

**Proposition 17** (Example 2.1 from [54]). *For $d = S \cdot A$ and $\varphi_h(s,a) = \boldsymbol{e}_{s,a}$ (where $\boldsymbol{e}_{s,a}$ denotes the $(s,a)$-th element of the standard basis of $\mathbb{R}^d$). Then, the class $\mathscr{M}_{tab}$ of Tabular MDPs is a subset of $\mathscr{M}_{lin}(\varphi)$, the class of Linear MDPs with feature map $\varphi$.*

Therefore, all the regret and statistical complexity bounds for Linear MDPs still apply to the tabular class when replacing $d$ with $S \cdot A$.

The first algorithm that was proved to achieve a regret bound for Linear MDPs is known as LSVI-UCB (Least-Squares Value Iteration with UCB, algorithm 1 in [54]). We won't dwell on the design of this algorithm now. Its complete structure is reported in algorithm 6.2, as it will be useful in the final chapters of this thesis. In a nutshell, LSVI-UCB estimates the action-value function via ridge regression and adds a bonus term to promote the exploration of under-sampled regions of the feature space. For such an algorithm, the following regret bound holds

**Theorem 18** (Theorem 3.1 from [54]). *Let $\mathscr{M}$ the class of Linear MDPs, assuming online interaction and assumptions 6 and 7. Assume the following normalization conditions*

$$\forall s, s' \in \mathcal{S}, a \in \mathcal{A}, h \in [H] \qquad \|\varphi_h(s,a)\|_2 \leq 1 \qquad \max\{\|\theta_h\|_2, \|\mu_h(s')\|_2\} \leq \sqrt{d}. \tag{3.4}$$

*Then, the LSVI-UCB algorithm enjoys*

$$RegretBound(LSVI - UCB|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^2\sqrt{d^3K}\right).$$

This result shows that, under the assumption of Linear MDP, the regret can be bounded independently on the state/action space size. The corresponding sample complexity can be computed with proposition 9 to be of order $\widetilde{\mathcal{O}}(H^5d^3/\varepsilon^2)$. Differently from the tabular case, where for $H = 1$ the RL algorithms were able to match the MAB lower bound for the regret, setting $H = 1$ here does *not* give the lower bound for Linear bandits, that are the bandit restriction of Linear MDPs. The lower bound for Linear bandits writes as (theorem 24.2 in [68]) $\Omega(d\sqrt{K})$, which is lower of a $\sqrt{d}$ term. We are digging more into this problem in the next section.

## Generalization: allowing for approximation error

The results of the previous section assume that the feature map decomposition is exact, meaning that both the reward and the transition probability are linear in $\varphi_h(s,a)$. A possible generalization is given by the following class.

**Definition 9.** *[Misspecified Linear MDPs, assumption B of [54]] We define Linear MDPs the family of MDPs endowed with a sequence of feature maps $\varphi_h : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ such that the following conditions hold.*

1. *For every $h \in [H]$ there is $\theta_h \in \mathbb{R}^d$ such that*

$$\forall s \in \mathcal{S}, a \in \mathcal{A} \qquad |r_h(s,a) - \varphi_h(s,a)^\top \theta_h| \leq \mathcal{I}.$$

2. *For every $h \in [H]$ there is a function $\mu_h : \mathcal{S} \to \mathbb{R}^d$ such that*

$$\forall s \in \mathcal{S}, a \in \mathcal{A} \qquad \|p_h(\cdot|s,a) - \varphi_h(s,a)^\top \mu_h(\cdot)\|_{TV} \leq \mathcal{I},$$

*where $\mathcal{I} > 0$ is a constant.*

As for $\varphi_h$, $\mathcal{I}$ is considered a feature of the class, and the agent has access to this number.

For $\mathcal{I} = 0$, the class coincides with Linear MDPs. The same paper proves the following.

**Theorem 19.** *Let $\mathscr{M}$ the class of Misspecified Linear MDPs, assuming online interaction and assumptions 6 and 7. Assume the normalization conditions eq. (3.4). Then, the LSVI-UCB algorithm enjoys*

$$RegretBound(LSVI - UCB|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^2\sqrt{d^3K} + H^2 d\mathcal{I}K\right). \tag{3.5}$$

While the first part scales as the result of theorem 18, the misspecification now appears in the second term as $\widetilde{\mathcal{O}}(H^2 d\mathcal{I}K)$. As expected, this term scales linearly with $K$ as, due to the misspecification, it is not guaranteed that any algorithm may converge to the return of the optimal policy; even with infinite data. What may surprise is that this term also scales with $H$ and, most importantly $d$.

The presence of $d$ is particularly interesting, as it is a major source of concern in realistic scenarios. Let us consider the very common setting where the agent has a very long feature map $\overline{\varphi}$ of length $D$ at its disposal for running a RL algorithm. Choosing the actual feature map $\varphi$ of length $d \leq D$ to run an RL algorithm requires to balance a trade-off: the more feature the less bias (i.e. $\mathcal{I}$) but the higher variance. Nonetheless, as the number of samples (in this case, interactions with the environment) grows, intuition suggests that the agent can choose a very large $d$. The result of eq. (3.5) denies this intuition. Even when $K \to +\infty$, so that the second term of the regret is the dominating one, increasing the $d$ improves the regret bound only if the corresponding decreasing of $\mathcal{I}$ is such that $d\mathcal{I} = o(1)$. In the opposite case, when for example the misspecification decreases as $\mathcal{O}(d^{-1/2})$ given the length of $\varphi$, adding more feature is not guaranteed to improve the asymptotic performance.

This result is so counterintuitive that one wonders whether it is due to a flaw in the regret demonstration for LSVI-UCB. Is it possible to avoid this strange phenomenon? To understand, let's start by defining a superclass of linear MDPs in which the approximation error plays a major role.

**Definition 10** (MDP with low inherent Bellman error [151])**.** *We define MDPs with Low Inherent Bellmann error the family of MDPs endowed with a sequence of feature maps $\varphi_h : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$ such that the following condition hold.*

$$\forall s \in \mathcal{S}, a \in \mathcal{A} \qquad \sup_{\theta \in \mathcal{B}_{h+1}} \inf_{\theta' \in \mathcal{B}_h} |Q_h[\theta'](s,a) - \mathcal{T}_h Q_{h+1}[\theta](s,a)| \leq \mathcal{I},$$

*where* $\{\mathcal{B}_h\}_{h=1}^H$ *is a sequence of subsets of* $\mathbb{R}^d$ *and*

$$Q_h[\theta](s,a) := \varphi_h(s,a)^\top \theta. \tag{3.6}$$

The transition probability $p_h$ does not enter into this definition, which instead only leverages on the Bellman Optimality operator in definition 3. As for Linear MDPs, we assume that the agent has full knowledge of the feature maps $\varphi_h$ and $\mathcal{I}$, while the precise sets $\mathcal{B}_h$ do not need to be known, even if an upper bound on their diameter will be required as normalization condition.

As anticipated, taking $\mathcal{I} = 0$ shows that standard Linear MDPs are a subset of this class, as proved by Proposition 3 in [151]. The same result shows that there is an MDP in the class of definition definition 10 that is not Linear, so the inclusion is strict. ELEANOR, the algorithm proposed in [151] to deal with this larger MDP class is way more complex than LSVI-UCB, and builds on the principle of *global optimism* which cannot, for now, be implemented in a computational efficient way except for particular cases [149].

**Theorem 20** (Theorem 1 from [151]). *Let* $\mathscr{M}$ *the class of MDPs with low inherent Bellman error, assuming online interaction and assumptions 6 and 7. Assume the following normalization conditions*

$$\forall s, s' \in \mathcal{S}, a \in \mathcal{A}, h \in [H] \qquad \|\varphi_h(s,a)\|_2 \leq 1 \qquad diam(\mathcal{B}_h) =: \mathcal{R}_h \leq \sqrt{d}H. \tag{3.7}$$

*Also, assume that* $\sup_{h,\theta\in\mathcal{B}_h} \|Q_h[\theta]\|_\infty \leq 2H$. *Then, the* ELEANOR *algorithm enjoys*

$$RegretBound(\text{ELEANOR}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^2 d\sqrt{K} + H^2\sqrt{d\mathcal{I}K}\right).$$

In the original paper, the previous result wrote as $Hd\sqrt{K} + H\sqrt{d\mathcal{I}K}$, since instead of assumption 6 it was assumed that $|Q_h^\pi(s,a)| \leq 1$ for any policy $\pi$. In our setting, we have seen that $|Q_h^\pi(s,a)| \leq H$ holds, so all the regret bounds must be scaled accordingly (see paragraph "Care must be taken when comparing..." in [151]). Aside from the normalization conditions, the only assumption corresponds to $\sup_{h,\theta\in\mathcal{B}_h} \|Q_h[\theta]\|_\infty \leq 2H$. As the optimal $Q-$function does not exceed $H$ and its best approximation is at most $H\mathcal{I}-$away (see lemma 6 from [151]), this condition can be met by just restricting $\mathcal{B}_h$, since $\mathcal{I} \ll 1$. The authors of the original paper showed the result under a slightly weaker assumption, that is for any $h \in [H], \theta \in \mathcal{B}_h$ the random variable $y_h^k + \max_a Q_h[\theta](s_{h+1}^k, a)$ minus its mean is $1-$sub-Gaussian (that would become $H$ in our setting). We decided to present this theorem in this slightly stronger formulation, as it is more coherent with the rest of

the result we present.

The regret bound we just stated improves the one for Misspecified Linear MDP of a factor $\sqrt{d}$ in both terms. The problem that we discussed in the previous paragraph, namely the fact that the bias term increases with $d$, is still present, even if this dependence is less relevant. In this case, in order to be worth adding more feature, we need $\sqrt{d}\mathcal{I} \to 0$, that is, the misspecification error must decrease as $o(d^{-1/2})$. Differently from theorem 18, this theorem matches the lower bound for its class, which was proved (theorem 2 from [151]), in case the reward is normalized so that $|Q^\pi(s,a)| \leq 1$, to be

$$\forall \mathscr{A} \quad \text{RegretBound}(\mathscr{A}|\mathscr{M}) \geq \Omega\left(Hd\sqrt{K} + H\sqrt{d}\mathcal{I}K\right).$$

apart from the different dependence on $H$ given by the different normalization, which is not of great interest for the present work, this result demonstrates unequivocally that a multiplicative dependence of the misspecification of order $\sqrt{d}$ cannot be avoided. It is important to note that this lower bound holds for a specifi feature map $\varphi$: in general, it is not known if the term $\sqrt{d}\mathcal{I}K$ is necessary.

### Sample complexity guarantees

As done for Tabular MDPs, we can turn the regret bound of this section into sample complexity guarantees by applying proposition 9. The assumptions of the proposition require the regret to grow sub-linearly, as it is not possible to guarantee a $\varepsilon-$optimal policy otherwise. Therefore, we have to assume $\mathcal{I} = 0$, so that the regret is only given by the part that grows as $\sqrt{K}$. Under such condition, plus the assumptions of theorem 20, we get

$$\text{SampleComplexity}(\textsc{Eleanor}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(\frac{H^5 d^2}{\varepsilon^2}\right).$$

As before, it is not clear if the order of magnitude in $H$ is the best possible. Even papers which worked specifically under the generative model assumption [141] were only able to improve this dependency to $H^3$ under stronger assumptions. For what concerns $d, \varepsilon$, their order is tight even in the bandit case (Theorem 24.1 in [68]).

## 3.3.4. Hardness Results for More General Classes

The former results of this section show that for many classes of MDPs the Generative Model assumption is not needed. In fact, algorithms exist that are able to achieve optimal sample complexity by just interacting online.

However, there are examples showing how this statement is not always correct if we increase the generality of the considered class. Note that the previous families of Linear MDPs (definition 8) and MDPs with Low-Inherent Bellman error (definition 10) with $\mathcal{I} = 0$ are contained in a larger, more natural family of MDPs. Let us call $\mathscr{M}_{\text{q-star}}$ the family of MDPs such that there is a feature map $\varphi_h$ satisfying

$$\forall h \; \exists \theta \; \forall s \in \mathcal{S}, a \in \mathcal{A} \qquad Q_h^*(s,a) = \varphi_h(s,a)^\top \theta.$$

We have already noted that Linear MDPs are included in $\mathscr{M}_{\text{q-star}}$, as the state-action value function of *every* policy is linear. For any MDP that respects definition 10 with $\mathcal{I} = 0$, one can prove by induction that if $Q_{h+1}^*$ is linear (say $= Q_{h+1}[\theta_h^*]$), then $Q_{h+1}^*$ is also linear. In fact,

$$\exists \theta' : \quad Q_h[\theta'](s,a) = \mathcal{T}_h Q_{h+1}[\theta_h^*](s,a) = \mathcal{T}_h Q_{h+1}^*(s,a) = Q_h^*(s,a).$$

Starting from $H+1$, for which we can choose $\theta_{H+1}^\star = 0$, this proves by induction that the MDP $\in \mathscr{M}_{\text{q-star}}$.

This class is of both empirical and theoretical interest. It holds empirical interest given the simplicity of its definition, which appears to be the most natural way to model value-based learning using linear function approximation. It is of theoretical interest as it is a superclass that includes both Linear MDPs and MDPs with low inherent Bellman error. However, a fundamental barrier prevents the generalization of previous theorems to this class.

**Proposition 21** (Theorem 5.11. from [4]). *Under the Generative Model assumption, fixing the error probability to $\delta = 0.1$, there is no algorithm $\mathscr{A}$ enjoying*

$$\mathit{SampleComplexity}(\mathscr{A} | \mathscr{M}_{\mathit{q\text{-}star}}, \varepsilon, \delta = 0.1) \leq \mathcal{O}(\mathit{poly}(\varepsilon^{-1}, H, d)).$$

As the Generative model interaction is the easiest case, this result also holds for the online interaction. By proposition 9, we deduce that no algorithm can enjoy a regret bound of the form $\text{poly}(\delta, H, d)K^\alpha$ with $\alpha < 1$.

Another characteristic of this class, which contradicts what we have observed so far for the previous ones, is that, under a more stringent condition, it becomes solvable in the generative model setting, but not in the online interaction setting. Let us define the

following class of MDPs.

$$\mathscr{M}_{\Delta} := \{\mathcal{M} : \forall h \in [H], \forall s \in \mathcal{S}, a \in \mathcal{A} \quad V_h^*(s) - Q_h^*(s,a) \notin (0, \Delta)\}.$$

This definition ensures that the gaps, i.e. $V_h^*(s) - Q_h^*(s,a)$ are not small, in the sense that can either be exactly zero (for the optimal arm) or $\geq \Delta$. The following result holds for the class $\mathscr{M}_{\text{q-star}} \cap \mathscr{M}_{\Delta}$.

**Proposition 22.** *[Theorem 5.12 and following section in [4]] Let $\mathscr{M} = \mathscr{M}_{q\text{-}star} \cap \mathscr{M}_{\Delta}$. The following results hold:*

1. *Generative model: there is an algorithm $\mathscr{A}$ enjoying*

$$SampleComplexity(\mathscr{A}|\mathscr{M}, \varepsilon, \delta) \leq \mathcal{O}(poly(\varepsilon^{-1}, \log(1/\delta), H, d, \Delta)).$$

2. *Online interaction: fixing the error probability to $\delta = 0.1$, there is no algorithm $\mathscr{A}$ enjoying*

$$SampleComplexity(\mathscr{A}|\mathscr{M}, \varepsilon, \delta = 0.1) \leq \mathcal{O}(poly(\varepsilon^{-1}, H, d, \Delta)).$$

The former result was firstly proved by [145]. The takeaway message from this theorem is that, while we previously presented classes of MDPs for which the same algorithm that enjoyed no-regret guarantees in the online case was also optimal in terms of sample complexity for a generative model, the same may not be possible in general. In fact, there are classes such that getting results in the online interaction case is intrinsically more difficult than in the generative model scenario.

## Other related classes

In this section, we have examined in detail some of the most important MDP classes that have been presented in the linear literature. These are only a fraction of a vast, interconnected family of classes that rely on the interaction between an MDP and a feature map. This family notably includes the class of $Q^{\pi}$-realizable MDPs [129, 146], linear mixture MDPs [13], and the bilinear class [37]. Defining these important MDP classes is beyond the scope of this thesis, which, starting from the next section, will focus on the main topic: MDPs with continuous state and action spaces.

# 4 | Learning in MDPs with Continuous State and Action Space

In this chapter, we define the main topic of this thesis: Reinforcement Learning in MDPs with continuous state and action spaces. The transition from finite to continuous domains introduces several new challenges, both theoretical and algorithmic. When states and actions belong to continuous sets, standard tools based on tabular representations or combinatorial exploration become inapplicable, and new notions of smoothness and approximation come into play.

We will start by revisiting the simplest instance of the problem, the bandit case, and use it to highlight the intrinsic difficulty of learning when the set of possible reward functions is infinite-dimensional. We then connect these ideas to the statistical complexity of regression in Sobolev-Hölder spaces, which serves as a mathematical foundation for the rest of the analysis.

Afterwards, we provide an overview of the current literature on continuous-state MDPs, covering the most common structural assumptions: Lipschitz continuity, linear-quadratic models, kernel-based formulations, and more general representation classes. Finally, we introduce the central contribution of this work, the definition of *Smooth MDPs*, a unified framework capturing different degrees of smoothness of the transition and reward functions. We close the chapter with a comparative overview of all these assumptions, highlighting their relationships and differences. A summary of the notation introduced and used in this chapter can be found in the appendix, table A.3.

## 4.1. Problem Definition and Preliminary Results

Let us begin by defining what we mean by a continuous-space MDP. The standard assumption we will make from now on is the following:

**Definition 11.** *We define continuous spaces MDP as a Markov Decision Process with $\mathcal{S} \subset \mathbb{R}^{m_1}, \mathcal{A} \subset \mathbb{R}^{m_2}$. We call $m := m_1 + m_2$ the total dimension of the state-action space.*

We have specified continuous *spaces* MDP to distinguish from the notion of continuous MDP, which is sometimes used to describe processes where the *time* is continuous. This definition includes different phenomena. Most of the works that we consider (all except for the ones studying the LQR and its generalizations), assume that $\mathcal{S}$ and $\mathcal{A}$ are compact sets. In this setting, the authors usually put, without loss of generality,

$$\mathcal{S} = [-1, 1]^{m_1} \qquad \mathcal{A} = [-1, 1]^{m_2}. \tag{4.1}$$

This choice has many advantages, first of all the fact that the set of state-action pairs does not need to be defined as a complex cartesian product, and corresponds just to $[-1, 1]^m$. Differently from Tabular MDPs, fixing the state and action space in this way is not sufficient to build an algorithm with theoretical guarantees. A notion of smoothness is necessary to avoid the "needle in a haystack" problem, which arises already in the bandit case.

### 4.1.1. The bandit case

In the bandit case, the previous definition reduces to $m_2 = m$, as the state space is a single point of dimension zero. This scenario, often called *continuum armed bandit* was studied for the first time by [5]. Even when $m = 1$, so $\mathcal{A} = [-1, 1]$, this problem does not admit any efficient algorithm, without any additional assumption. Consider the family of bandit instances defined by

$$\mathscr{B} := \{r_x(a) = \mathbf{1}_{\{x\}}(a),\ x \in [-1, 1]\}.$$

In $K$ interactions with the environment, any bandit algorithm is going to always pull arms corresponding to zero reward, except for a set of instances of size no more than $K$. Therefore, it will suffer regret equal to $K$ in all the other instances, whose number is uncountably infinite, and this even *if the samples $y_k$ are noiseless*. The only way to prevent this phenomenon is to allow generalization between different actions, so that the agent, when observing the result of a given pull $a \in [-1, 1]$, is able to update its estimation of the reward not just for $a$, but for an infinite set of other actions that are linked to $a$.

The two seminal works [63, 64], proposed a solution for this problem under the assumption that the reward function is Lipschitz continuous.

**Definition 12.** *A function $f : \mathcal{X} \subset \mathbb{R}^m \to \mathbb{R}$ is Lipschitz continuous with constant $L > 0$ if the following inequality is respected.*

$$\forall x, y \in \mathcal{X} \quad |f(x) - f(y)| \leq L\|x - y\|_2.$$

Definition 12 is very simple to interpret: any change of $\varepsilon$ of the argument of a function corresponds to an absolute change in the value that does not exceed $L\varepsilon$. Differentiable functions on a bounded domain are always Lipschitz continuous, and their constant $L$ corresponds to the maximum of the first derivative in modulus. But the concept of Lipschitz continuity is way more general, and it could be generalized to functions between an arbitrary pair $(\mathcal{X}, d_{\mathcal{X}}) \to (\mathcal{Y}, d_{\mathcal{Y}})$ of metric spaces (see [64] for a general treatment).

Staying on the CAB problem with $\mathcal{A} = [-1, 1]^m$, several algorithms based on discretization of the action space $\mathcal{A}$ are capable of achieving the optimal regret bound, as the following result shows. Perhaps, the most popular among them is ZOOMING which dynamically discretizes the action space by maintaining a set of active, "zoomed-in" actions. It creates a new active action whenever a pulled arm yields high rewards, or adaptively refines the discretization in regions of the space that are frequently selected. By balancing exploration of uncertain regions with exploitation of highly rewarding neighborhoods, it effectively shrinks the search space around optimal actions to achieve the optimal regret bound.

**Theorem 23** (Theorem 1.1 in [64])**.** *Consider a CAB problem such that assumptions 6 and 7 hold, and the reward function is $L-$Lipschitz continuous. Then, the regret of ZOOMING is bounded as follows, with probability at least $1 - \delta$,*

$$R_K \leq \widetilde{\mathcal{O}}\left(K^{\frac{m+1}{m+2}}\right).$$

Crucially, the dimension[1] of the space changes the order of the regret in $K$, while the specific value of the $L$ only appears as constant, like $\sigma$. As mentioned, ZOOMING is just one among the many algorithms that achieve this regret bound, and other works are able to achieve the same bound in worst case and a tighter one in favorable scenarios [11]. Nonetheless, when only assuming that the reward function is Lipschitz continuous, the regret bound of the previous theorem is tight, as the following theorem shows.

**Theorem 24** (Theorem 1.1 in [64], part 2)**.** *For any $\mathscr{A}$, there is an instance of continuous armed bandits where the reward function is $L-$Lipschitz and assumptions 6 and 7 hold,*

[1] In fact, the regret of ZOOMING scales with the doubling dimension [64], which can be strictly smaller than $m$ if the reward function effectively "lives" on a lower-dimensional subset of $[-1, 1]^m$. However, analyzing such specific topological structures is beyond the scope of this work.

*such that it suffers regret*

$$R_K \geq \Omega\left(K^{\frac{m+1}{m+2}}\right).$$

While these results ensure the existence algorithms whose regret is sublinear, the regret order is very close to one for large $m$. This is more evident when passing from the result of theorem 23 to the corresponding sample complexity that is, as proposition 9 shows, $n = \widetilde{\mathcal{O}}(\varepsilon^{-2-m})$. The exponent $-m$ on $\varepsilon$ in the statistical complexity constitute a strong *curse of dimensionality*. When the size of the space is big, the number samples needed becomes extremely high. On a geometric viewpoint, this is inevitable: in order to build a grid of $[-1,1]^m$ that catches a point that is $\varepsilon-$optimal for $f$, we need a step-size of no more than $\varepsilon/L$ (as we need to guarantee $f(a^\star) - f(a) \leq L\varepsilon$), which means a total of $\approx (L/\varepsilon)^m$ points.

### 4.1.2. Smooth Bandits

Many more or less cumbersome solutions have been proposed to solve this problem by adding more assumptions. To the best of our knowledge, the only one[2] that relies solely on the smoothness of the reward function, without introducing any other assumptions, is the one proposed by [74]. This work starts from the fact that the class of Lipschitz functions is actually a particular case of much more general classes of functions $\mathbb{R}^m \to \mathbb{R}$, the Sobolev-Hölder spaces[3]. On an informal level, a function is in a Sobolev Space of order $\nu \in \mathbb{N}$ and index $p \in [1, +\infty)$ if its derivatives up to order $\nu$ are bounded in $L^p-$norm (recall that, for a function $f$, $\|f\|_{L^p} = \sqrt[p]{\int_{\mathcal{X}} |f(x)|^p dx}$).

$$\mathrm{W}^{\nu,p}(\mathcal{X}) := \{f : [-1,1]^m \to \mathbb{R} \text{ such that all weak derivatives of order } \nu \text{ are in } L^p(\mathcal{X})\}.$$

For $p = +\infty$, the $L^\infty-$norm is instead defined as the supremum of the modulus of $f$. Thus, for $\nu \in \mathbb{N}$,

$$\mathrm{W}^{\nu,\infty}(\mathcal{X}) := \{f : [-1,1]^m \to \mathbb{R} \text{ such that all weak derivatives of order } \nu \text{ are bounded}\}.$$

Surprisingly, Lipschitz functions correspond exactly to the space $\mathrm{W}^{1,\infty}$: while Lipschitz-Continuous functions they may not be differentiable in every point, they are differentiable *almost everywhere* and the corresponding weak derivative is a bounded function (see

[2]Some papers assume the reward function to be in a RKHS with Matérn Kernel. While it is not trivial to see, this assumption is strictly linked to its smoothness. We will focus to this topic in detail in section 4.2.4

[3]for the same topic in detail, see appendix B.1

appendix B.1 for more details). $\mathrm{W}^{\nu,\infty}$ is a Banach space if endowed with a proper norm. A common choice is

$$\|f\|_{\mathrm{W}^{\nu,\infty}} := \max_{|\alpha|\leq\nu} \|D^\alpha f\|_{L^\infty} = \max_{|\alpha|\leq\nu} \sup_{x\in\mathcal{X}} |D^\alpha f(x)|.$$

Once again, this norm is very close to the to the Lipschitz constant. In fact, the following equality holds

$$\|f\|_{\mathrm{W}^{1,\infty}} := \max\{\|f\|_{L^\infty}, L(f)\}. \tag{4.2}$$

While Sobolev spaces do not typically cover the fractional $\nu$ case, it is possible to generalize the former definition to the case $\nu \in [0,\infty) \setminus \mathbb{N}$ with an Hölder condition. The previous (4.2) gives an equivalent definition of the $\mathrm{W}^{1,\infty}$ norm by leveraging the Lipschitz constant. As Lipschitz functions are also a particular case of Hölder continuous functions, one can generalize $\mathrm{W}^{\nu,\infty}(\mathcal{X})$ to the fractional case by imposing that for any index $|\boldsymbol{\alpha}| = \lceil \nu - 1 \rceil$ the derivative $D^{\boldsymbol{\alpha}} f$ is $(\nu - \nu_*)-$Hölder continuous (see (B.2) for more detail). Its corresponding norm follows the same eq. (4.2), with $L(f)$ replaced by the Hölder constant.

**Remark 25.** *For $\nu \in (0,1)$, the class $W^{\nu,\infty}(\mathcal{X})$ reduces exactly to the class of Hölder-continuous functions, i.e. such that*

$$\forall x, y \in \mathcal{X} \qquad |f(x) - f(y)| \leq C\|x - y\|_2^\nu.$$

From now on, we will refer to $\mathrm{W}^{\nu,\infty}$, which is omnipresent in the rest of the thesis, as the space of *Sobolev-Hölder functions*.

Coming back to CAB, the authors of [74] proved the following regret bound, which generalizes the previous from Lipschitz bandits.

**Theorem 26** (Theorem 4. + Eq. 5 in [74])**.** *Consider a CAB problem such that assumptions 6 and 7 hold, and the reward function satisfies $\|r\|_{W^{\nu,\infty}} \leq B$ for some $B > 0$. Then, the regret of UCB-Meta-Algorithm is bounded, with probability at least $1 - \delta$, by*

$$R_K \leq \widetilde{\mathcal{O}}\left(K^{\frac{m+\nu}{m+2\nu}}\right).$$

The former result presents a different exponent than the one of Lipschitz bandit. While, as expected, they coincide for $\nu = 1$, the order of $K$ in theorem 26 writes as follows

$$\frac{m+\nu}{m+2\nu} = \frac{m/2+\nu}{m+2\nu} + \frac{m}{2m+4\nu} = \frac{1}{2} + \underbrace{\frac{m}{2m+4\nu}}_{\in(0,1/2)}.$$

The first part, 1/2 is the best that can be done in any bandit problem, while the second part interpolates between 0 and 1/2 depending on what dominates between $m$ and $\nu$. If $m$ is big, but the function is very smooth the overall exponent is still close to 1/2. Coherently, if we derive the sample complexity guarantee corresponding to theorem 26, we get $\mathcal{O}(\varepsilon^{-2-m/\nu})$. This dramatic improvement over the previous result shows that for $\nu \gg m$ the *curse of dimensionality* is broken: it is possible to learn the optimal action at an improved rate. Also, this order of magnitude on $\varepsilon$ is tight. As proved by [143], in order to find an $\varepsilon-$optimal point for a function that is in the $\mathrm{W}^{\nu,\infty}([-1,1]^m)$ class with bandit feedback, any algorithm needs a number of samples of at least

$$n = \Omega(\varepsilon^{-2-m/\nu}) \tag{4.3}$$

in the worst case. What makes theorem 26 even more remarkable is that in many practical applications, having very smooth functions, i.e., $\nu \approx \infty$, is the rule, not the exception. In fact, for any compact set $\mathcal{X}$ one has, for every $\nu > 0$, $\mathrm{C}^{\infty}(\mathcal{X}) \subset \mathrm{W}^{\nu,\infty}(\mathcal{X})$. The former space, corresponding to function that are differentiable infinitely many times, is strictly linked to the solution of important partial differential equations (PDEs). In particular, thermal, electromagnetic, and wave phenomena are governed by three well-known different equations: the heat equation, the Laplace-Poisson equation, and the D'Alembert equation, respectively [108, 119, 126]. Each of these is characterized by inherent regularity properties so that solutions are infinitely differentiable under suitable boundary conditions.

The work by [74] offers a powerful regret bound for a vast class of CAB problems. In the next sections, we will take deep inspiration from its ideas, to apply them in a more general scenario. Generalizing this approach to Reinforcement Learning introduces a key challenge: instead of simply maximizing a function, as in bandit problems, we must now estimate the optimal state-action value function accurately enough to enable planning. For bandits, this problem does not arise because, having only one step and no state, the agent only needs to worry about maximizing the reward function. However, in the general case of RL, uniform estimation of the value functions becomes necessary.

### 4.1.3. Statistical Complexity of Nonparametric Regression

While it is possible to break the curse of dimensionality regarding the regret bound on bandits by assuming that the reward function is sufficiently smooth, generalizing this result to the RL world is far from obvious. Even without taking into account the time horizon, which goes from 1 to $H$, the mere presence of the state makes everything much more complex. In fact, it is no longer a question of simply approximating the action where

the reward function has its maximum, but of finding the best action for all states of the MDP that are visited sufficiently often. In the worst case, this requires approximating the optimal state-action value function uniformly over all states.

For this reason, it is useful to recall the theory of regression on smooth functions with uniform error guarantees. This problem was known in the history of statistics as *non-parametric regression*. Classic results of this discipline state that, given a fixed function $f \in \mathrm{W}^{\nu,\infty}([-1,1]^m)$ and a dataset $(x_i, y_i)_{i=1}^n$ with $x_i$ uniformly sampled over $[-1,1]^m$ and $y_i$ such that $\mathbb{E}[y_i|x_i] = f(x_i)$, there exists an algorithm that outputs a function $\widehat{f}_n$ such that

$$\lim_{n\to\infty} \mathbb{P}\left(\|f - \widehat{f}_n\|_{L^\infty} \geq (n/\log(n))^{-\frac{m}{2\nu+m}}\right) = 0. \tag{4.4}$$

This result was first presented in [122] and is valid if the tails of the noise are sufficiently light (similar to our assumption 7). Such a formulation, common in the statistical learning literature, has a slightly different focus than our statistical complexity bounds. However, if one ignores the logarithmic terms and considers $\varepsilon = n^{-\frac{\nu}{2\nu+m}}$, they deduce that the number of samples needed to achieve a given accuracy is bounded as follows:

$$n \approx \varepsilon^{-2-\frac{m}{\nu}} \implies \|f - \widehat{f}_n\|_{L^\infty} \leq \varepsilon.$$

This conclusion is encouraging, since it corresponds to the exact same order of sample complexity as the bandit case presented in theorem 26, although in this case it is necessary to approximate the entire function, and not just the maximum.

Nonetheless, equation (4.4) cannot be directly applied to our context; it provides only an asymptotic bound, whereas RL theory typically requires high-probability sample complexity bounds, a significantly more stringent requirement. Moreover, the result of eq. (4.4) assumes a very specific distribution for the samples $x_i$. While this condition may be satisfied in case of a *generative model* interaction, it is not in the *online learning* framework, where the agent is only able to see full trajectories in the MDP, without querying specific states.

### 4.1.4. From Smooth Bandits to Smooth MDPs

The remainder of this section will focus on the theoretical literature on RL in continuous spaces. Given the greater complexity compared to the bandit case, a vast literature has been produced trying to prove theoretical guarantees for suitable algorithms on the largest possible families of processes.

To what extent do the MDP classes in the literature address the curse of dimensionality

problem? What is the trade-off between having the most general assumptions possible and the possibility of proving sublinear bounds on regret? Is there any common characteristic that can pigeonhole them? These questions, which we explore in section 4.2, motivate the definition of the family of Smooth MDPs, the main focus of this thesis, in section 4.3.

## 4.2. Existing Approaches for Continuous-State MDPs

MDPs with continuous state and action spaces provide a much more realistic framework for modeling real-world decision-making problems compared to their tabular counterparts. These models naturally capture environments where variables evolve smoothly, such as robotics, control systems, or financial markets. In the context of Reinforcement Learning, they have been studied for decades, bridging ideas from control theory, function approximation, and nonparametric statistics to address the challenges posed by infinite-dimensional spaces.

The long-standing interest in these models, which predates the emergence of continuous bandits [109], has fostered a heterogeneous body of literature characterized by widely varying assumptions and settings. In this section, we will organize this work into four main clusters, which we will present in chronological order.

1. In section 4.2.1, we will discuss MDP models that rely on classic control theory assumptions. These have their roots in the study of LQRs, dating back to Kalman's seminal work [57] in the 1960s. Nonetheless, models of this type are still of widespread interest in the modern RL literature.

2. In section 4.2.2 we will focus on the concept of Lipschitz MDP. This much more general model uses a simple assumption borrowed from the bandit case for the reward, while defining the regularity of the transition function on the Wasserstein metric basis. This work has developed mainly since 2010, following the success of Lipschitz bandits.

3. In section 4.2.4 we will focus on kernelized models, i.e., models that assume rewards and transitions in a suitable kernel-reproducing Hilbert space. This branch of the literature developed ahead of the boom in kernel methods in machine learning [50].

4. Finally, in section 4.2.5 we will study the theory of continuous state-action space frameworks under the assumption of general function approximation. This literature is very recent and is based on the concept of the Eluder dimension, [107], a complexity measure for optimistic exploration with general function approximation.

While these studies cover diverse settings, they will be unified and compared against the MDP classes proposed in this thesis in a dedicated section at the end of this chapter.

### 4.2.1. LQRs and their Generalizations

The LQR model, which we have already partially explained in example 4, is obtained by assuming that $\mathcal{S} = \mathbb{R}^{m_1}, \mathcal{A} = \mathbb{R}^{m_2}$ while the state evolves according to

$$s' = As + Ba + \eta, \qquad \eta \sim \mathcal{N}(0, \Sigma)$$

where $A \in \mathbb{R}^{m_1 \times m_1}, B \in \mathbb{R}^{m_1 \times m_2}$ and $\Sigma$ is a $m_1 \times m_1$ covariance matrix. The reward is given by

$$r(s, a) = -s^\top Q s - a^\top R a,$$

for SDP matrices $Q, R$. We have omitted the dependence on $h$ since these functions are the same at any stage of the process.

Differently from all the works presented so far, the standard regret bounds for the LQR do not operate in an episodic setting. Instead, they work in the infinite-horizon average-cost regime, where instead of $K$ episodes of length $H$, there is one single episode of length $T$. This setting is generally considered more difficult, as the agent has no possibility to reset the state to $s_0$ but, to draw a comparison, one could think to $T$ as $HK$.

The first regret guarantees for adaptive LQR were obtained by Abbasi-Yadkori and Szepesvári [1], who analyze the online control problem under the following assumptions:

1. Known matrices $Q, R$
2. $\Sigma = \sigma I$

Letting $\mathscr{M}$ the family of MDPs that respect these assumptions, Abbasi-Yadkori and Szepesvári [1] designed an algorithm $\mathscr{A}$ such that

$$\text{RegretBound}(\mathscr{A}|\mathscr{M}, T, \delta) \leq \widetilde{\mathcal{O}}\left(\exp(d)\sqrt{T}\right), \qquad d = m_1^2 + m_1 m_2.$$

Here, $d$ represents the total number of unknown parameters. Their analysis relies on optimism over confidence sets for the unknown matrices $A, B$, together with delicate stability arguments ensuring that the closed-loop system remains bounded with high probability. While the regret rate achieved is of order $\widetilde{\mathcal{O}}(\sqrt{T})$, their bound suffers from an exponential dependence on the system dimension. This was the first result proving sublinear regret for LQR, but it left open the question of whether such performance could

be achieved efficiently. Moreover, $\mathscr{A}$ is very computationally inefficient, suffering also of exponential complexity in $d$.

A major step forward was made by [34], who developed the first polynomial-time and computationally efficient algorithm that enjoys high-probability regret guarantees. Under the same assumption, [34] designed an algorithm named Robust Adaptive Control Algorithm (RACA) yielding a regret bound of order

$$\text{RegretBound}(\text{RACA}|\mathscr{M}, T, \delta) \leq \widetilde{\mathcal{O}}\left(\text{poly}(d)T^{2/3}\right).$$

Their method combines robust control techniques with finite-sample system identification. The key contribution is the construction of a controller that ensures uniform stability for all model parameters within the confidence set. This method avoids the non-convex optimization required by optimism-based methods: this is the key for polynomial regret in $d$ and computational tractability. Nonetheless, this technique enforces a worse scaling in $T$, that now appears as $T^{2/3}$ instead of the optimal $\sqrt{T}$.

The previous two works left an interesting open problem, obtaining a computationally efficient algorithm with $\widetilde{\mathcal{O}}(\text{poly}(d)\sqrt{T})$ regret, which was resolved by [29]. Their algorithm is based on a semidefinite-program (SDP) relaxation of the infinite-horizon LQR objective. As [1], it yields an optimistic controller but, differently form the former work, it does not require non-convex search over stabilizing feedback matrices. The algorithm designed by [29] has the form, Optimistic Semi-definite programming for Lq cOntrol (OSLO) enjoys the following regret bound

$$\text{RegretBound}(\text{OSLO}|\mathscr{M}, T, \delta) \leq \widetilde{\mathcal{O}}\left(d^2\sqrt{T}\right),$$

which is optimal in $T$. Having solved the problem of computational complexity and eliminated the exponential dependence on the dimension $d$, all that remained was to obtain an algorithm that simultaneously satisfied all these requirements and achieved regret scaling that is minimax optimal with respect to the number of unknown parameters. This final step was accomplished by [116], who closed the problem by matching lower and upper bounds for online LQR up to logarithmic factors. On the lower-bound side, they proved that the regret of any algorithm must grow at least as

$$\text{RegretBound}(\mathscr{A}|\mathscr{M}, T, \delta) \geq \Omega\left(\sqrt{m_1^2 m_2 T}\right),$$

which, as $d = m_1^2 + m_1 m_2$ means between $\sqrt{dT}$ and $\sqrt{d^{3/2}T}$.

On the upper-bound side, they showed that a remarkably simple algorithm, certainty-equivalent control with injected Gaussian noise for the exploration (CE-GE), run over a doubling-epoch schedule, achieves the matching regret

$$\text{RegretBound}(\text{CE-GE}|\mathscr{M}, T, \delta) \leq \widetilde{\mathcal{O}}\left(\sqrt{m_1^2 m_2 T}\right).$$

The result demonstrates that naive exploration is in fact optimal for online LQR: no sophisticated exploration strategy can improve the order of regret, and no further algorithmic machinery is required to achieve minimax performance.

Together, these two contributions, an instance-dependent lower bound and a matching upper bound attained by a practical, polynomial-time controller, fully characterize the statistical difficulty of adaptive linear–quadratic control, resolving the open questions left by the earlier literature.

## Generalizations of the LQR

As we have seen, classical regret bounds on LQR operate in regimes very different from our assumptions, particularly for the non-episodic setting, which belongs more to control theory than to the RL universe. In this section, we will define some generalizations of this setting that operate episodically.

Beyond the linear–quadratic structure of the classical LQR, [56] consider a significantly more general model of controlled dynamics, which writes as follows

$$s' = W\varphi(s,a) + \eta, \qquad \eta \sim \mathscr{N}(0, \sigma I),$$

for a *known* feature map $\varphi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$. In the particular case $\varphi(s,a) = [s,a]$ and

$$W = \begin{bmatrix} A & B \end{bmatrix}, \qquad W \in \mathbb{R}^{m_1 \times (m_1+m_2)},$$

this model includes the transition of the standard LQR. An important difference from the LQR literature is the treatment of the reward function: in [56], $r_h(s,a)$ is time-dependent, non-quadratic but *bounded*, as in assumption 6.

Under this setting, [56] introduces an algorithm, LC$^3$, that achieves the following regret bound.

$$\text{RegretBound}(\text{LC}^3|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(\sqrt{d(d+m_1)H^3K}\right),$$

where, as we mentioned, $m_1$ is the dimension of the space $\mathcal{S}$ and $d$ the length of the feature

map. In the standard LQR case, where $\varphi(s,a) = [s,a]$, the previous bound writes as

$$\text{RegretBound}(\text{LC}^3|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(\sqrt{(m_1+m_2)(2m_1+m_2)H^3K}\right),$$

which may seem to break the $\Omega\left(\sqrt{m_1^2 m_2 T}\right)$ proved by [116]. In fact, this is not true, due to the boundedness assumption on the reward, which is done by [56] while it is not present in the standard LQR. Furthermore, the $\text{LC}^3$ algorithm requires an oracle for the planning part once the feature matrix $W$ has been estimated, and is therefore not computationally efficient, unlike state-of-the-art algorithms we have seen for LQR.

**Bilinear Exponential Family of MDPs** A substantially more expressive generalization of LQR models is provided by the Bilinear Exponential Family (BEF) of MDPs [27][97]. In this framework, both the transition probability and the reward distribution belong to a parametric natural exponential family whose parameter depends bilinearly on two feature vectors: $\varphi(s,a)$ for the state–action pair and $\psi(s')$ for the next state. Specifically, in their setting the transition probability function is given by

$$p(s'|s,a) = \exp\left(\psi(s')^\top W \phi(s,a) - Z(s,a)\right),$$

where $Z(s,a)$ is a constant used to normalize the probability to one. This model contains the LQR as a special case. In fact, by fixing $\varphi(s,a) = [s,a,1]$, $\psi(s') = [s', -\|s'\|_2^2]$ and

$$W = \begin{bmatrix} 2A & 2B & 0 \\ 0 & 0 & 1 \end{bmatrix}, \qquad W \in \mathbb{R}^{(m_1+1)\times(m_1+m_2+1)},$$

one gets

$$\begin{aligned} p(s'|s,a) &\propto \exp\left(\psi(s')^\top W \phi(s,a)\right) \\ &= \exp\left(2{s'}^\top(As+Ba) - \|s'\|_2^2\right) \\ &\propto \exp\left(-\|As+Ba\|_2^2 + 2{s'}^\top(As+Ba) - \|s'\|_2^2\right) \\ &= \exp\left(-\|s'-As-Ba\|_2^2\right) \propto \mathscr{N}(s'|As+Ba, I). \end{aligned}$$

For this large family of processes [97] introduce a practical algorithm, BEF-RLSVI, that achieves tractable exploration and tractable planning without relying on any oracle. The method combines penalized maximum-likelihood estimation for the BEF parameters with a randomized value-iteration scheme in which exploration is induced by injecting calibrated Gaussian noise directly into the estimated reward parameter. Fixing $d = d_\varphi \times d_\psi$, the total number of learnable parameters as in the standard LQR, and under proper

normalization assumptions, the regret of their algorithm is bounded as

$$\text{RegretBound}(\text{BEF-RLSVI}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(\sqrt{d^3H^3K}\right).$$

Despite the generality of the problem, the authors also demonstrate that the proposed algorithm has a polynomial computational complexity in all the problem parameters.

### 4.2.2. Lipschitz MDPs

In the previous section, we looked at LQR and its variants, which constitute a structured family of MDPs with continuous state-action spaces. Despite the model's historical importance, these processes capture only a small fraction of continuous-space MDPs.

A much wider family can be defined by just assuming that small variations in the state-action pair $(s, a)$ lead to small variations in the reward function $r(s, a)$ and small variations in the transition function $p(\cdot|s, a)$. This is precisely the idea behind the setting of *Lipschitz MDPs* [104].

**Definition 13** (Lipschitz MDPs). *The class $\mathscr{M}$ of Lipschitz MDPs is the set of MDPs where $\mathcal{S} = [-1, 1]^{m_1}, \mathcal{A} = [-1, 1]^{m_2}$ and there are constants $L_r, L_p$ such that, for all $h \in [H], s, s' \in \mathcal{S}, a, a' \in \mathcal{A}$*

$$|r_h(s, a) - r_h(s', a')| \leq L_r(\|s - s'\|_2 + \|a - a'\|_2) \tag{4.5}$$
$$D(p_h(\cdot|s, a), p_h(\cdot|s', a')) \leq L_p(\|s - s'\|_2 + \|a - a'\|_2), \tag{4.6}$$

*where $D$ denotes a notion of distance between probability distributions.*

Lipschitz MDPs assume that the transition function, as well as the reward function, are Lipschitz continuous with constants $L_p$ and $L_r$, respectively. For the reward function, this is just expressed by enforcing that for all pairs of state-action couples the difference of the reward function is bounded by $L_r(\|s - s'\|_2 + \|a - a'\|_2)$. As the transition function maps a state-action pair into a probability distribution, to find a metric in the co-domain of this function is less trivial. In the most common formulation of these MDPs, D is chosen as the Wasserstein metric, which is defined, for two probability distributions $\mu, \omega$ over a probability metric space $\mathcal{X}$, as

$$\mathcal{W}(\mu, \omega) := \sup_{\|f\|_{\mathrm{W}^{1,\infty}} \leq 1} \int_{\mathcal{X}} f(x)(\mu(x) - \omega(x))dx \tag{4.7}$$

Where the maximization is on the set of Lipschitz functions (which we have seen coincides

with $\mathrm{W}^{1,\infty}(\mathcal{X})$, appendix B.1) with norm bounded by one. Nonetheless, a formulation similar to Lipschitz MDPs also exists for the total variation distance. It is sufficient to replace (4.6) with the following

$$\mathrm{TV}(p_h(\cdot|s,a), p_h(\cdot|s',a')) \leq L_p(\|s-s'\|_2 + \|a-a'\|_2). \tag{4.8}$$

The reason why many works prefer the formulation with the Wasserstein distance over the one with the total variation is that the latter does not admit deterministic MDPs. In fact, taking a deterministic MDP with a transition given by $s' = f(s,a)$, one has

$$\mathrm{TV}(p_h(\cdot|s,a), p_h(\cdot|s',a')) = \mathrm{TV}(\delta_{f(s,a)}, \delta_{f(s',a')}) = 1,$$

regardless how close $(s',a')$ is to $(s,a)$. Thus, it is not possible to find a constant $L_p$ satisfying eq. (4.8).

Lipschitz MDPs have been applied to a number of different settings, both because of their simple definition and of their notable flexibility. Not only do they allow developing policy gradient methods that enjoy a theoretical guarantee [9, 87, 101], they also help to tackle generalizations of standard RL, such as RL with delayed feedback [73] or configurable RL [86], and auxiliary tasks for imitation learning [31, 76].

We will now review the main algorithms and regret bounds obtained for Lipschitz MDPs in the reinforcement learning literature. For the whole section, $\mathscr{M}_{\mathcal{W}}$ is going to denote the class of Lipschitz MDP defined in definition 13 with the Wasserstein distance in eq. (4.7), while $\mathscr{M}_{\mathrm{TV}}$ will indicate the same class but with the assumption on the total variation distance. In the same way, we will call $m = m_1 + m_2$, the dimension of the state-action space, which often appears in the regret bounds. All the algorithms leverage the geometric smoothness of the process to discretize the state–action space in a statistically efficient manner.

The first provably efficient algorithm to be invented for this problem is UCCRL from [96]. This algorithm is a direct generalization of the well-known UCRL2 which makes use of a uniform discretization of the state-action space. The regret of this algorithm is obtained in the non-episodic setting, and writes as

$$\mathrm{RegretBound}(\mathrm{UCCRL}|\mathscr{M}_{\mathrm{TV}}, T, \delta) \leq \widetilde{\mathcal{O}}\left(T^{\frac{2m+1}{2m+2}}\right) \qquad m = m_1 + m_2.$$

This result only holds for the class $\mathscr{M}_{\mathrm{TV}}$, and not for its more general version $\mathscr{M}_{\mathcal{W}}$. Moreover, the regret bound does not reach the lower bound for Lipschitz bandits, where

$\widetilde{\mathcal{O}}(T^{\frac{m+1}{m+2}})$ was achieved (note that Lipschitz bandits are indeed a special case for this problem for $m_1 = 0$ and $H = 1$).

Many have attempted to extend and improve this result. In particular, it has been useful to move from the model-based approach of [96] to a value-based one, as done by [117, 120, 130]. All these algorithms move along the line of discretization of the state-action space, and are more or less inspired by the bandit world. In particular, [130] implements an adaptive discretization that exactly mirrors that of ZOOMING. The worst-case regret bounds coincide in all the three cases with the one NB-QL by [120]. Under our base assumptions 6 and 7, the former algorithm enjoys the following regret bound

$$\text{RegretBound}(\text{NB-QL}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^{5/2} L_Q^{\frac{d}{d+2}} K^{\frac{m+1}{m+2}}\right).$$

This result is both valid when $\mathscr{M} = \mathscr{M}_{\mathcal{W}}$ and for $\mathscr{M}_{\text{TV}}$.

This result has the same order as the bandit case, and is therefore optimal. However, the regret bound includes a quantity we had not defined, namely $L_Q$, the Lipschitz constant of the optimal policy. While it is clearly related to the constants $L_r$ and $L_p$ that appear in definition 13, it also contains a dependence on $H$ that is hidden by this symbol.

For Lipschitz MDP with the total variation assumption (the class $\mathscr{M}_{\text{TV}}$), this dependence on $H$ is polynomial: Proposition 2.2 from [120] proves that $L_Q = \mathcal{O}(HL_rL_p)$, so that the regret bound writes as

$$\text{RegretBound}(\text{NB-QL}|\mathscr{M}_{\text{TV}}) \leq \widetilde{\mathcal{O}}\left(H^{5/2+\frac{m}{m+2}} K^{\frac{m+1}{m+2}}\right),$$

which, despite massive, is still a polynomial dependence on the horizon. For the more general case of $\mathscr{M}_{\mathcal{W}}$, the situation is much worse: the same proof strategy based on iterating the Bellman optimality operator is only able to prove that $L_Q = \mathcal{O}(L_p^H L_r)$. This results in the following regret bound

$$\text{RegretBound}(\text{NB-QL}|\mathscr{M}_{\mathcal{W}}) \leq \widetilde{\mathcal{O}}\left(H^{5/2} L_p^{\frac{dH}{d+2}} L_r^{\frac{d}{d+2}} K^{\frac{m+1}{m+2}}\right) = \widetilde{\mathcal{O}}(\exp(H) K^{\frac{m+1}{m+2}}).$$

This result is not due to a suboptimal approach in the algorithm's analysis, nor to an imperfection of the algorithm itself. On the contrary, Theorem 12 of [79] shows that, surprisingly,

$$\forall \mathscr{A} \qquad \text{RegretBound}(\mathscr{A}|\mathscr{M}_{\mathcal{W}}) \geq \Omega\left(L_p^{\frac{d(H-2)}{d+2}} K^{\frac{m+1}{m+2}}\right).$$

This lower bound shows that the $\exp(H)$ regret is necessary for this family of MDP, which constitutes a serious obstacle to the possibility of having satisfactory theoretical guarantees.

### 4.2.3. A Note on the Dependency on $m$

Although we have not explicitly reported it, all previous bounds on the regret scale with an exponential constant in $m$, which is hidden by the notation $\mathcal{O}(\cdot)$.

While this phenomenon may be scary, this dependence is negligible w.r.t. the one on $K$ in any learnable regime. To see this, note that, the regret lower bound (even in the bandit case) writes as

$$R_K = \Omega\left(K^{\frac{m+1}{m+2}}\right).$$

Now, let us assume that $m \gtrsim \beta \log(K)$ for some constant $\beta > 0$. As a consequence,

$$\begin{aligned} R_K &= \Omega\left(K^{\frac{m+1}{m+1}}\right) = \Omega\left(K^{\frac{\beta\log(K)+1}{\beta\log(K)+2}}\right) = \Omega\left(K^{1-\frac{1}{\beta\log(K)+2}}\right) \\ &\geq \Omega\left(K^{1-\frac{1}{\beta\log(K)}}\right) = \Omega\left(K \cdot e^{\frac{-\log(K)}{\beta\log(K)}}\right) = \Omega\left(K \cdot e^{-1/\beta}\right) = \Omega\left(K\right). \end{aligned}$$

This lower bound shows that the Lipschitz MDP problem, as well as similar nonparametric classes that we study, is not learnable unless $m = o(\log(K))$. Therefore, from now on, for all the non-parametric classes we treat $m$ as a small constant, which can be hidden in the regret/sample complexity bounds unless it is at the exponent of $H, K, \varepsilon$.

### 4.2.4. Kernelized MDPs

As we have seen, LQRs offer strong regret guarantees at the cost of a very limited flexibility. On the other side, Lipschitz MDP are extremely general, but suffer a regret that grows almost linearly, as $K^{\frac{m+1}{m+2}}$ and, when using the definition with the Wasserstein metric, exponentially in $H$. In this section, we see another family of processes that stands between the former two in terms of generality, that is the one of kernelized MDPs.

A natural way to impose smoothness on continuous MDPs is to assume that rewards and transitions belong to a Reproducing Kernel Hilbert Space (RKHS, for this topic in detail see appendix B.4). This viewpoint connects RL with the well–developed theory of kernel methods and Gaussian processes, which have achieved great success in the history of

machine learning. In this section we review three representative works that progressively sharpen our understanding of learning in kernelized MDPs. This family of MDPs is defined as follows [26, 134, 150].

**Definition 14** (Kernelized MDPs)**.** *The class $\mathscr{M}$ of Kernelized MDPs is the set of MDPs where, calling $\mathcal{Z} = \mathcal{S} \times \mathcal{A}$, there is a kernel function $\mathscr{K} : \mathcal{Z}^2 \to \mathbb{R}$ such that, for all $h \in [H], s' \in \mathcal{S}$ the functions $r_h$ and $p_h(s'|\cdot)$ belong to the RKHS $\mathcal{H}(\mathscr{K})$ induced by $\mathscr{K}$ with*

$$\|r_h(\cdot)\|_{\mathcal{H}} \leq 1, \qquad \|p_h(s'|\cdot)\|_{\mathcal{H}} \leq 1 \qquad \mathcal{H} = \mathcal{H}(\mathscr{K}). \tag{4.9}$$

In this section, $\mathscr{M}$ will always denote this family of MDPs. While in the most general formulation the spaces $\mathcal{S}, \mathcal{A}$ can be arbitrary, given the topic of this thesis, we focus, as most of the papers on the topic, on the case where $\mathcal{S} = [-1, 1]^{m_1}, \mathcal{A} = [-1, 1]^{m_2}$, so that $\mathcal{Z} = [-1, 1]^m$. In this setting, the most popular choice of $\mathscr{K}$ is given by the Matérn kernel (B.11) of parameter $\nu \in (0, \infty)$.

The first algorithms to deal with this family of MDPs are the model-based approaches of [26]. Their algorithms GP-UCRL and GP-PSRL generalize UCRL2 and posterior sampling (PSRL) to this nonparametric setting: they 1) maintain Gaussian process posteriors for both the transition and reward function, 2) construct confidence sets for these functions using kernel ridge regression, 3) at the beginning of each episode solve a planning problem in an optimistic (or sampled) MDP consistent with the confidence sets. Leveraging these techniques, under the usual assumptions 6 and 7, [26] is able to show that

$$\text{RegretBound}(\text{GP-UCRL}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^{5/2}\gamma_K\sqrt{K}\right).$$

Note that in the original paper a regret bound of order $\widetilde{\mathcal{O}}\left(H^{3/2}\gamma_K\sqrt{K}\right)$ is shown for the case where transition and reward do not depend on $h$, i.e. $r_h(s, a) = r(s, a)$ for every $h$. We have multiplied the result by $H$ to take into account the overhead of learning each function separately, to guarantee a fair comparison with the other regret bounds of this chapter. In this bound, $\gamma_K$ is the Maximum Information Gain, a measure of complexity for an RKHS that, intuitively, quantifies the reduction in uncertainty about the a target function in the space after collecting $K$ data. Conceptually, the work of [26] establishes that the assumption 14 of Kernelized MDP suffice to obtain sublinear regret in continuous MDPs. However, the price is that each episode requires solving a hard planning problem in a continuous MDP, so the algorithm is computationally inefficient.

Another seminal paper for the literature of Kernelized methods in RL is [150]. This work takes a different route and ask whether one can obtain provably efficient algorithms that

are both computationally and statistically tractable when using a value-based algorithm. They propose an optimistic least-squares value iteration scheme, inspired from the Linear MDP algoritm LSVI-UCB that we discussed around definition 8, and instantiate it in the kernel setting as KOVI (Kernel Optimistic Value Iteration).

Here the kernel is used to approximate the state-action value functions $Q_h^*(s,a)$ directly. At each episode, the algorithm performs a backward pass over stages $h = H, \dots, 1$ and for each stage, it fits a kernel ridge regression that predicts the Bellman update. Then, as in LSVI-UCB, it adds an upper-confidence bonus derived from kernel bandits to obtain an optimistic estimate of the function.

The resulting regret bounds has the form

$$\text{RegretBound}(\text{KOVI}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^2\sqrt{(\gamma_K + \log \mathcal{N}_\infty(1/\sqrt{K}, \mathcal{B}_1(\mathcal{H}))\gamma_K T}\right).$$

While the result is valid for an algorithm with much lower computational complexity than its predecessor, the bound on regret not only does not improve, but it contains an additional term that did not appear before. We define it formally, since we will need it later.

**Definition 15** (Covering number). *Let $\mathcal{F}$ be a class of functions $\mathcal{X} \to \mathbb{R}$, and $\varepsilon > 0$. We define the $\infty-$covering number, write $\mathcal{N}_\infty(\varepsilon, \mathcal{F})$, of the function class as the minimum $N$ such that there are functions $f_1, \dots f_N \in \mathcal{F}$ such that*

$$\forall f \in \mathcal{F} \quad \exists n \in [N] \qquad \|f - f_n\|_\infty \leq \varepsilon.$$

In the particular case of the previous result, $\varepsilon = 1/\sqrt{K}$ and $\mathcal{F} = \mathcal{B}_1(\mathcal{H})$, i.e. the 1-norm ball in the RKHS. Both $\gamma_K$ and this covering number are measures of complexity for the RKHS, and very often they grow at the same rate. Thus, the regret bound for KOVI is not much different from the one of GP-UCRL.

Unfortunately, in this setting even a regret bound that grows as $\widetilde{\mathcal{O}}(\gamma_K\sqrt{K})$ is not fully satisfactory. In fact, for the family of Matérn kernels and the SE, [137] proved the following *tight* bounds on the information gain

$$\text{Matérn}: \ \gamma_K = \mathcal{O}\left(K^{\frac{m}{m+2\nu}}\right) \qquad \text{SE}: \ \gamma_K = \mathcal{O}\left(\log^{m+1}(K)\right).$$

When trying to replace these values in a regret bound of the form $\widetilde{\mathcal{O}}(\gamma_K\sqrt{K})$ one gets

$$\text{Matérn}: \quad R_K = \widetilde{\mathcal{O}}\left(K^{\frac{3/2m+\nu}{m+2\nu}}\right). \tag{4.10}$$

For a Matérn kernel with moderate $\nu$ in a space where $m \geq 2\nu$, this bound grows **super-linearly** in $K$. This means that the result is trivially vacuous and no convergence to the value of the optimal policy is guaranteed. As the bound over the information gain is tight, this is only an issue of the algorithms or their analysis.

A similar issue used to exist in the field of Kernelized bandits, which may be seen as definition 14 with space of dimension 0 and $H = 1$. Subsequent research [71, 136, 138] introduced algorithms attaining the lower bound of $\Omega(\sqrt{\gamma_K K})$ by employing methodologies that transcend standard Gaussian process optimization. Notably, the optimality of the latter remains an open question in the current literature [147].

The question that naturally arises at this point is whether, even in the RL context, which is much more general than a simple bandit, it is still possible to obtain regret of order $\widetilde{\mathcal{O}}(\sqrt{\gamma_K K})$. Such a result would imply, for the Matérn kernel family, regret of order $R_K = \widetilde{\mathcal{O}}\left(K^{\frac{m+\nu}{m+2\nu}}\right)$, which is always sublinear.

Several attempts have been made to achieve this, among which [134] is particularly noteworthy. This work attempts to address the problem by generalizing an algorithm from [51] that relies on domain splitting to the RL case. Unfortunately, the central proof contains an error that undermines the validity of the final result (see Appendix F in [78]).

Obtaining an algorithm that achieves optimal (or at least sublinear) regret for Kernelized MDPs is still an open problem recognized by the world's leading learning theory conference [133].

### 4.2.5. General Function Approximation

The last category of algorithms we will consider comes from a different family, namely RL under general function approximation.

In recent years, RL theory has seen a rush in searching for the weakest assumptions under which sample-efficient RL is possible. These works abandon explicit parametric models of the MDP and instead characterize the environment through the assumption that either the transition dynamics or the optimal value functions lie in a given function class. Rather than relying on linearity, kernel structure, or Lipschitz continuity, the only structural requirement is that the true functions governing the MDP belong to a rich

hypothesis space $\mathcal{F}$. Learning then reduces to identifying the correct function in $\mathcal{F}$ by interacting with the environment. The statistical difficulty of this identification problem is captured by a notion originating from bandit theory: the Eluder dimension [107].

**Definition 16** (Eluder dimension)**.** *Let $\mathcal{F}$ be a class of functions $\mathcal{X} \to \mathbb{R}$. For fixed $\varepsilon > 0$ we call Eluder dimension of $\mathcal{F}$, write $dim_E(\varepsilon, \mathcal{F})$, the maximum length of a sequence of points $x_1, \dots x_d \subset \mathcal{X}$ such that for any $d_0 < d$ there exist $\varepsilon_0 \geq \varepsilon$ and $f, g \in \mathcal{F}$ such that*

$$\sqrt{\sum_{i=1}^{d_0}(f(x_i) - g(x_i))^2} \leq \varepsilon_0 \qquad f(x_{d_0+1}) - g(x_{d_0+1}) \geq \varepsilon_0.$$

Intuitively, the Eluder dimension $\dim_{\mathrm{E}}(\varepsilon, \mathcal{F})$ measures how many observations are required to eliminate functions that behave differently in "independent" ways. In bandits, a small Eluder dimension implies that disagreement in $\mathcal{F}$ collapses quickly as data is collected, enabling algorithms to identify near-optimal actions with few samples. The works presented in this section aim to generalize this powerful idea to the reinforcement learning setting, offering two complementary interpretations of what the underlying function class $\mathcal{F}$ should represent. In the first line of work, $\mathcal{F}$ is used to model the transition dynamics themselves, capturing the structure of how states evolve under actions. In the second, the notion of general function approximation is applied directly to the optimal value functions, viewing reinforcement learning as the problem of identifying the correct value function within a rich hypothesis space.

It is worth mentioning that Bellman-Eluder dimension is not the only framework proposed for studying reinforcement learning with general function approximation. Another recent line of work seeks to characterize the difficulty of reinforcement learning problems through an abstract complexity measure known as the Decision–Estimation Coefficient (DEC) [41, 43]. The DEC framework recovers several previously proposed settings as special cases and can be viewed as complementary to Bellman-Eluder dimension. Moreover, DEC can itself be bounded in terms of Bellman-Eluder dimension and other complexity measures introduced in the literature [41]. Since, as we will show, Bellman-Eluder dimension admits explicit characterizations for several function classes of interest in this thesis, we will focus on results based on this notion.

The work of [105] develops the first regret guarantees for episodic RL in general continuous state and action spaces under the sole assumptions that the transition dynamics lie in a prescribed function class $\mathcal{F}$ and the noise is Gaussian.

**Definition 17** (Gaussian Transition-Realizable MDPs)**.** *Given a set $\mathcal{F}$ of bounded func-*

*tions $\mathbb{R}^m \to \mathbb{R}^{m_2}$, the class $\mathscr{M}_{GTR}$ of Gaussian Transition-Realizable MDPs is the set of MDPs where $\mathcal{S} = \mathbb{R}^{m_1}, \mathcal{A} = \mathbb{R}^{m_2}$, with $m = m_1 + m_2$ and for any $h \in [H]$ the transition takes the form*

$$s' = f^\star(s, a) + \eta \qquad \eta \sim \mathscr{N}(0, \sigma^2 I).$$

As anticipated, the main assumption is that $f^\star \in \mathcal{F}$, a known class of bounded functions $\mathbb{R}^m \to \mathbb{R}^{m_1}$. As usual, we note $m = m_1 + m_2$.

Importantly, the Gaussian noise plays a structural role: it ensures that the stochastic transition can be expressed as a linear functional of $f^\star$ in a proper RKHS, enabling the problem to be cast as a linear MDP in an infinite-dimensional feature space. The algorithm builds on the principle of maintaining a confidence set over candidate transition functions $f \in \mathcal{F}$ that are statistically consistent with past observations. At each episode, the agent selects the most optimistic model in this set and plans greedily with respect to it. The key technical contribution is to show that this RKHS embedding is possible precisely because of the presence of the additive Gaussian noise (The "free lunch" in the title of the paper).

Under definition 17, assuming that the reward function is known and respects assumption 6, the authors establish the following regret guarantee for their SPEDE algorithm.

$$\text{RegretBound}(\text{SPEDE}|\mathscr{M}_{\text{GTR}}) \leq \widetilde{\mathcal{O}}\left(H^{3/2}\sqrt{\dim_{\text{E}}((HK)^{-1/2}, \mathcal{F}) \log \mathcal{N}_\infty((HK)^{-1/2}, \mathcal{F})K}\right).$$

It is worth mentioning that

- the previous regret bound depends equally on the two class complexity measures, covering number and Eluder dimension
- The dependence on $H$, of the form $H^{3/2}$ does not arise from an analysis via the Bernstein bound, but from the fact that by definition of the class the transition function is the same at all steps $h \in [H]$.

This bound shows that, when the noise is Gaussian, sample-efficient RL is achievable in continuous domains as long as the function class describing the transition dynamics has controlled complexity: the dependence on $\mathcal{F}$ enters only through its (log-)covering number and Eluder dimension, and no additional structural assumptions is required.

The second contribution [55] we consider introduces a fundamentally different perspective on general function approximation in RL. Instead of modelling the transition dynamics, this line of work assumes that the optimal value functions themselves lie in a prescribed function class $\mathcal{F} = \mathcal{F}_1 \times \ldots \mathcal{F}_h \cdots \times \mathcal{F}_H$. The only structural assumption of the setting is that, whenever making a backup with the Bellmann optimality operator $\mathcal{T}_h$, the function

class $\mathcal{F}_h$ is mapped into an auxiliari function class $\mathcal{G}_h$ known.

**Definition 18** (MDPs with bounded Bellman-Eluder dimension)**.** *Given two classes $\mathcal{F} = \mathcal{F}_1 \times \dots \mathcal{F}_h \cdots \times \mathcal{F}_H$ and $\mathcal{G} = \mathcal{G}_1 \times \dots \mathcal{G}_h \cdots \times \mathcal{G}_H$ of bounded functions $\mathcal{S} \times \mathcal{A} \to \mathbb{R}$, the class $\mathscr{M}_{BE}$ of MDPs is the set of MDPs wtih state space $\mathcal{S}$ and action space $\mathcal{A}$ such that, for any $h \in [H]$, the corresponding Bellman optimality operator $\mathcal{T}_h$ satisfies*

1. *$\mathcal{T}_h(\mathcal{F}_{h+1}) \subseteq \mathcal{G}_h$*
2. *for any $\varepsilon > 0$, $dim_{BE}(\varepsilon, \mathcal{F}, \mathcal{P}) := \max_{h\in[H]} dim_{DE}(\varepsilon, (I - \mathcal{T}_h)\mathcal{F}_h, \mathcal{P}) < +\infty$ (see definition 19), for $\mathcal{P}$ that is either the set of Dirac's deltas over $\mathcal{S} \times \mathcal{A}$ or the set of probability distributions over the state action couples that may result when executing a greedy policy with respect to some $f \in \mathcal{F}$.*

The class name, Bellmann-Eluder, is relative to the complexity measure used to demonstrate the regret bound of this family of processes.

**Definition 19** (Distributional Eluder dimension)**.** *Let $\mathcal{F}$ be a class of functions $\mathcal{X} \to \mathbb{R}$ and $\mathcal{P}$ a set of probability distributions over $\mathcal{X}$. For fixed $\varepsilon > 0$ we call Distributional Eluder dimension of $\mathcal{F}$ (note $dim_{DE}(\varepsilon, \mathcal{F}, \mathcal{P})$) the maximum length of a sequence of probability distributions $\mu_1, \dots \mu_d \subset \mathcal{P}$ such that for any $d_0 < d$ there exist $\varepsilon_0 \geq \varepsilon$ and $f \in \mathcal{F}$ such that*

$$\sqrt{\sum_{i=1}^{d_0} \mathop{\mathbb{E}}_{x\sim\mu_i} [f(x)]^2} \leq \varepsilon_0 \qquad \mathop{\mathbb{E}}_{x\sim\mu_{d_0+1}} [f(x)] \geq \varepsilon_0.$$

This notion of dimension is strictly linked to the standard Eluder dimension when $\mathcal{P}$ is the set of all Dirac's delta over $\mathcal{X}$. Indeed, letting $\mathcal{F}$ be an arbitrary function class and $\mathcal{F} - \mathcal{F} := \{g = f_1 - f_2,\ f_1, f_2 \in \mathcal{F}\}$, it follows that

$$\dim_{\mathrm{DE}}(\varepsilon, \mathcal{F}, \mathcal{P}) = \dim_{\mathrm{E}}(\varepsilon, \mathcal{F} - \mathcal{F}) \qquad \mathcal{P} = \text{deltas on } \mathcal{X}.$$

Despite its inherent complexity, definition 18 provides a unified framework for a vast class of function approximation problems, the specifics of which exceed the scope of this thesis [52, 142]. The algorithm proposed by [55], named GOLF, maintains confidence sets over candidate Q-functions, selects the optimistic one, and acts greedily with respect to it. The key innovation is the introduction of the Bellman Eluder dimension, a refinement of the classical Eluder dimension tailored to value-function learning, which generalizes the standard Eluder dimension to probability distributions and applies it to the Bellman operator. This quantity governs how quickly ambiguity in value estimates can be resolved,

capturing the intrinsic complexity of planning under general function approximation.

Under the usual assumptions 6 and 7, the authors are able to prove the following regret bound.

$$\text{RegretBound}(\text{GOLF}|\mathscr{M}_{\text{BE}}) \leq \widetilde{\mathcal{O}}\left(H^2\sqrt{d_{\text{BE}}\log\mathcal{N}_\infty(K^{-1},\mathcal{F}\cup\mathcal{G})K}\right)$$
$$d_{\text{BE}} = \min\left\{\dim_{\text{BE}}(K^{-1/2},\mathcal{F},\mathcal{P}_\delta),\dim_{\text{BE}}(K^{-1/2},\mathcal{F},\mathcal{P}_\pi)\right\},$$

where $\mathcal{P}_\delta, \mathcal{P}_\pi$ are the set of Dirac's deltas over $\mathcal{S}\times\mathcal{A}$ and the set of probability distributions over the state action couples that may result when executing a greedy policy with respect to some $f \in \mathcal{F}$.

Whenever some particular structure is present, $\dim_{\text{BE}}(K^{-1/2},\mathcal{F},\mathcal{P}_\pi) \ll \dim_{\text{BE}}(K^{-1/2},\mathcal{F},\mathcal{P}_\delta)$; but in worst-case, as every action can be optimal for some $f \in \mathcal{F}$ and every state can be visited, the two quantities coincide. Typically, $d_{\text{BE}}$ reduces to $\dim_{\text{BE}}(K^{-1/2},\mathcal{F},\mathcal{P}_\delta) = \max_{h\in[H]}\dim_{\text{DE}}(K^{-1/2},(I-\mathcal{T}_h)\mathcal{F}_h,\mathcal{P}_\delta)$. As we have seen the former can be reduced again to

$$\max_{h\in[H]}\dim_{\text{DE}}(K^{-1/2},(I-\mathcal{T}_h)\mathcal{F}_h,\mathcal{P}_\delta) = \max_{h\in[H]}\dim_{\text{E}}(K^{-1/2},(I-\mathcal{T}_h)(\mathcal{F}_h-\mathcal{F}_h)),$$

which recovers the standard Eluder dimension.

The results by [55] show that sample-efficient RL is possible in extremely rich settings, where the transitions are entirely unstructured, as long as the value function class has controlled Bellman Eluder dimension, making it one of the most general positive results in RL theory to date.

### 4.2.6. State of the Art Summary and Sample Complexity

Throughout this chapter, we have explored much of the RL literature on continuous state and action spaces. The discussion has focused on online interaction, the most widely studied setting in the literature as it provides the strongest theoretical guarantees, namely regret bounds. In this short section, we will attempt to summarize all the results and verify which sample complexity guarantees correspond to each regret bound.

As previously done, we will use proposition 9 to convert the existing regret bounds of the best-performing algorithms for continuous MDP families into sample complexity guarantees. The corresponding rate is presented in the *S.C.* column of table 4.1. In the same table, we have also added a *Tractable* column to indicate whether the algorithm is (relatively) computationally efficient or requires a planning oracle. Finally, we included the *Known reward* column, which specifies whether the algorithm requires prior knowledge of

the reward function.

Let us describe the results of table 4.1 starting from the sample complexity column. The presented bounds depend on some quantities of interest: $m_1$ denotes the dimension of the state space, $m_2$ that of the actions, and $m = m_1 + m_2$. Furthermore, in lines 2, 3, the MDP class is equipped with a feature map, whose dimension we call $d$, which is completely decoupled from $m$. Other complexity measures are the information gain $\gamma_K$ relative to the kernels and the log covering number $N_\infty$ of the class of functions that characterize the process (computed for different values of $\varepsilon$, as we specify in the notes). Finally, we have denoted $\dim_{\mathrm{E}}$ as the Eluder dimension and $d_{\mathrm{BE}}$ as the Bellmann-Eluder (both for $\varepsilon \approx K^{-1/2}$).

The results in the table exhibit varying sample complexities, an expected outcome given the broad generality of the MDP classes considered. For the LQR, the Nonlinear Regulator [56] and the Bilinear Exponential Family [97], the order of the sample complexity is $\widetilde{\mathcal{O}}(\varepsilon^{-2})$, regardless of the dimension $m$ of the state-action space. When the generality is significantly increased by moving to the unstructured class of Lipschitz MDPs (whether defined by the total variation or the Wasserstein metric), things change radically. The order of the sample complexity then becomes $\widetilde{\mathcal{O}}(\varepsilon^{-2-m})$, which grows exponentially with the dimension. When we finally move to even more flexible models, such as Kernelized MDPs or the general function approximation settings defined by [105] and [55] no finite sample complexity guarantee is possible. The reason lies in the dependence of the regret bounds on complexity measures like $\gamma_K$, $N_\infty$, and the Eluder dimension. The unpredictable dependence of these measures on $K$ prevents us from applying proposition 9 to obtain a finite guarantee. This is not a problem caused by the proposition we are trying to apply: in the next chapters we are seeing some choices of the function class (that are compulsory to deal with some MDPs) such that the regret bounds of GOLF and SPEDE grow super-linealy in $K$ (for what concerns the regret of KOVI, the same was anticipated in the section about Kernelized MDPs). As convergence to the optimal return is not granted, vacuous regret bounds do not correspond to any sample complexity guarantee.

Although computational complexity is not a central theme of this thesis, we have also included a *Polynomial Time* column in the table, which expresses whether the defined algorithm is computationally efficient or requires a planning oracle. For non-parametric classes, to put ✓, we require polynomial time in $H, K$, but not in $m$ and the absence of planning oracle. Importantly, section 4.2.3 shows that all learnable families of non-parametric MDPs (Lipchitz or more general) have $m = o(\log(K))$, a complexity that grows as $\exp(m)$ is acceptable. Instead, for parametric classes like LQRs, where no such condition exists, we also require polynomial dependency in $m$ and, when present, in the

| Class | Algorithm | Regret Bound | S. C. | Polynomial Time | Unknown reward |
|---|---|---|---|---|---|
| LQR | CE-GE | $\sqrt{m_1^2 m_2 T}$ | $\frac{m_1^2 m_2}{\varepsilon^2}$ | ✓ | ✗ |
| NR | LC$^3$ | $\sqrt{d(d+m_1)H^3K}$ | $\frac{d(d+m_1)H^4}{\varepsilon^2}$ | ✗ | ✗ |
| BEF | BEF-RLSVI | $\sqrt{d^3H^3K}$ | $\frac{d^3H^4}{\varepsilon^2}$ | ✓ | ✗ |
| TV-Lips. | NB-QL | $\text{poly}(H)K^{\frac{m+1}{m+2}}$ | $\frac{\text{poly}(H)}{\varepsilon^{m+2}}$ | ✓ | ✓ |
| $\mathcal{W}$-Lips. | NB-QL | $\exp(H)K^{\frac{m+1}{m+2}}$ | $\frac{\exp(H)}{\varepsilon^{m+2}}$ | ✓ | ✓ |
| Kernelized | KOVI | $H^2\gamma_K\sqrt{K}$ | N/A | ✓ | ✗ |
| GTR | SPEDE | $H^{3/2}\sqrt{\dim_E N_\infty^\dagger K}$ | N/A | ✓ | ✗ |
| BE | GOLF | $H^2\sqrt{d_{BE}N_\infty^\ddagger K}$ | N/A | ✗ | ✗ |

Table 4.1: In this table, we have reported the main state-of-the-art results presented in the chapter. The tractable column indicates whether the corresponding algorithm is computationally efficient (✓) or requires a planning oracle (✗). The unknown reward column indicates whether the algorithm works for noisy rewards (✓) or requires seeing the deterministic reward for each state (✗).
† The log-covering number is computed with $\varepsilon = K^{-1/2}$.
‡ The log-covering number is computed with $\varepsilon = K^{-1}$.

length $d$ of the feature map.

As mentioned, the state-of-the-art of LQRs, CE-GE, is computationally efficient. Instead, LC$^3$ requires a planning oracle because it is model-based and lacks a specific structure. GTR is efficient, because of the kernel trick. In contrast, GOLF requires a minimization oracle to find the hypothesis within the class that minimizes the empirical Bellman error. Without the oracle, the computational complexity of this algorithm scales linearly with the size of $\mathcal{F}$. As we will see later, the application of GOLF to continuous MDPs often requires a hypothesis class whose covering number is essentially $\exp(K^{m/2})$.

Finally, in the *Unknown reward* column, we consider whether the algorithm functions even when the reward is noisy and accessible only through noisy samples, or if complete knowledge is instead required. This column is merely a technical detail, as it is widely accepted in the literature that having an unknown transition function constitutes the major obstacle, and generalizing to a noisy reward is a simple exercise.

We conclude the section with two fundamental considerations. The first is that it is not yet known whether it is actually possible to obtain a sample complexity guarantee for some of

the mentioned classes. All works assume online interaction, but working within the more convenient generative model scenario could lead to more significant results. Secondly, there is still no work that successfully leverages the smoothness of the process in a way similar to what is done by [74] in order to mitigate the exponential sample complexity in $m$. These facts motivate for the introduction of the classes of Smooth MDPs, which are the main topic of this thesis.

## 4.3. Smooth MDPs: One Idea, Three Assumptions

As we have seen, the Reinforcement Learning literature concerning MDPs with continuous spaces relies on a vast array of assumptions. These inspirations appear to stem from multiple, disparate sources that often lack a unifying framework. Currently, the state of the art does not clearly elucidate the role of smoothness in determining whether a specific class of Markov Decision Processes (MDPs) admits no-regret algorithms.

In this section, we aim to define the family of Smooth MDPs to establish a common thread between these diverse approaches. As anticipated, Sobolev-Hölder spaces introduced in eq. (4.2) (and appendix B.1 in a more extensive form) are at the core of their definition.

The first formalization of the concept of smoothness in MDPs is simply based on imposing that both the reward and the transition function belong to the class of functions just described. The class of processes defined in this way is called *Strongly* Smooth MDPs.

**Definition 20** (*Strongly* Smooth MDPs)**.** *The class $\mathscr{M}_{SS}$ of Strongly Smooth MDPs is the set of MDPs where $\mathcal{S} = [-1,1]^{m_1}, \mathcal{A} = [-1,1]^{m_2}$ and, for some $\nu \geq 0$ and for all $h \in [H], s' \in \mathcal{S}$ the functions $r_h$ and $p_h(s'|\cdot)$ belong to $W^{\nu,\infty}(\mathcal{Z})$ with*

$$\|r_h\|_{W^{\nu,\infty}} \leq C_r^{SS}, \qquad \|p_h(s'|\cdot)\|_{W^{\nu,\infty}} \leq C_p^{SS}, \tag{4.11}$$

*for proper constants $C_r^{SS}$, $C_p^{SS}$.*

As one can imagine, from the inclusion $\mathrm{W}^{\nu,\infty}(\mathcal{Z}) \subset \mathrm{W}^{\nu',\infty}(\mathcal{Z})$ for $\nu' \leq \nu$, it follows that the class gets smaller the bigger $\nu$. This assumption brings to mind, the class of Kernelized MDPs, in definition 14. In particular, the difference with respect to the latter lies in the class of functions used: no longer a given RKHS but rather a suitable Sobolev-Hölder space. Also, for the kernelized case the bound on the norm of the reward and transition is fixed to one, while here and in the original assumption of the paper [79] we use given constants $\mathrm{C}_\mathrm{r}^{\mathrm{SS}}, \mathrm{C}_\mathrm{p}^{\mathrm{SS}}$.

As the name implies, this assumption is more stringent than the other notions of Smooth

MDPs discussed hereafter. We shall begin by briefly outlining the implications of this regularity requirement on both reward and transition functions. First, it's interesting to note that if we choose $h = 1$, our assumption reduces to the smooth bandits introduced at the beginning of the chapter. This fundamental property will be shared by all variants of the Smooth MDP that we are introducing later. The peculiarity of this class of MDPs is that it imposes a transition function that is smooth over the entire state-action space $\mathcal{Z}$. In the most intuitive case of a continuous-space MDP, where the transition maintains the form

$$s_{h+1} = f_h(s_h, a_h) + \eta_{h+1} \qquad \eta_{h+1} \sim \mathscr{P}(\cdot),$$

where $\mathscr{P}$ is the noise density function, the transition function takes the form

$$p_h(s'|s, a) = \mathscr{P}(s' - f_h(s_h, a_h)).$$

For this function to be smooth with respect to $s, a$, the composition between $\mathscr{P}$ and $f_h$ must be smooth. As the condition must hold for every $s' \in \mathcal{S}$, is constant, both functions must be smooth (except for trivial cases where $\mathscr{P}$ is constant).

Requiring that the noise density function also be smooth is a strong assumption and, as we will see, unnecessary to obtain most of the regret bounds. For this reason, we call this class of processes Strongly Smooth. Nonetheless, this fact holds true in some practical cases, such as the aforementioned LQR section 4.2.1. In fact, having

$$f(s, a) = As + Ba \qquad \eta \sim \mathscr{N}(0, \Sigma) \propto \exp(-x^\top \Sigma x),$$

ensures that both functions are infinitely differentiable, belonging to $\mathrm{W}^{\nu,\infty}(\mathcal{Z})$ for every $\nu > 0$. While this may seem like a subtlety, it should be noted that LQR is not comparable to Smooth MDPs due to the different assumptions about the state-action space, which here is considered a bounded hypercube, while in LQR it was the entire space $\mathbb{R}^m$. To include LQR in the family of Smooth MDPs, it is necessary to project it onto a compact set, as in the following example.

**Example 27.** *In this example we show that an LQR projected on a bounded space through a smooth function $\varphi : \mathbb{R}^{m_1} \to B_1(0) \subset \mathbb{R}^{m_1}$ gives rise to a Strongly Smooth MDP. Let $\mathcal{M}$ be an MDP that evolves according to*

$$s_{h+1} = \varphi(As_h + Ba_h + \eta) \qquad \eta \sim \mathscr{N}(0, \Sigma),\ \varphi(x) := \frac{x}{\sqrt{1 + \|x\|_2^2}}.$$

*There is no doubt that the state of the system cannot escape from $B_1(0)$, the ball of vectors*

*in $\mathbb{R}^{m_1}$ whose norm does not exceed one. Indeed, $\|\varphi(x)\|_2 < 1$ for any $x$. Now, let us try to compute the transition function. First, note that $\varphi$ is a biijection from $\mathbb{R}^{m_1}$ to $B_1(0)$, meaning that it admits a unique inverse $\varphi^{-1}(s') = \frac{s'}{\sqrt{1-\|s'\|_2^2}}$, which is also smooth for $\|s'\|_2 < 1$. Therefore, calling $p_{LQR}$ the density function for the LQR with transition $s_{h+1} = As_h + Ba_h + \eta$, the following approximation holds*

$$\begin{aligned} p_h(s'|s,a) &= |\det J_\varphi(\varphi^{-1}(s'))|^{-1} p_{LQR}(\varphi^{-1}(s')|s,a) \\ &= |\det J_\varphi(\varphi^{-1}(s'))|^{-1} \mathcal{N}(\varphi^{-1}(s'); As+Ba, \Sigma) \\ &\propto |\det J_\varphi(\varphi^{-1}(s'))|^{-1} \exp(-x^\top \Sigma^{-1} x/2) \quad x = \varphi^{-1}(s') - As - Ba, \end{aligned}$$

*where $J_\varphi$ denotes the Jacobian of the function $\varphi$. In this case, we have*

$$|\det J_\varphi(\varphi^{-1}(s'))|^{-1} = \left(1 + \|\varphi^{-1}(s')\|_2^2\right)^{\frac{m_1+2}{2}}.$$

*Therefore, the total formula gets*

$$p_h(s'|s,a) \propto \frac{\exp(-x^\top \Sigma^{-1} x/2)}{(1-\|s'\|_2^2)^{(m_1+2)/2}} \qquad x = \varphi^{-1}(s') - As - Ba.$$

*This function is differentiable an infinite number of times in $s, a$. We have also to ensure that $\|p_h(s'|\cdot)\|_{W^{\nu,\infty}}$ is bounded uniformly over $s'$. The only problem could arise for $\|s'\|_2 \to 1$, as the denominator converges to zero. Nonetheless, using the fact that $\varphi^{-1}(s') = \frac{s'}{\sqrt{1-\|s'\|_2^2}}$, we have*

$$\begin{aligned} \lim_{\|s'\|_2 \to 1} D^\alpha p_h(s'|s,a) &= \lim_{\|s'\|_2 \to 1} poly(s,a) p_h(s'|s,a) \\ &\propto \lim_{\|s'\|_2 \to 1} \frac{\exp(-x^\top \Sigma^{-1} x/2)}{(1-\|s'\|_2^2)^{(m_1+2)/2}} \qquad x = \varphi^{-1}(s') - As - Ba \\ &= \lim_{\|s'\|_2 \to 1} \frac{\exp\left(-\frac{s'}{\sqrt{1-\|s'\|_2^2}}^\top \Sigma^{-1} \frac{s'}{\sqrt{1-\|s'\|_2^2}}/2\right)}{(1-\|s'\|_2^2)^{(m_1+2)/2}} \\ &\leq \lim_{\|s'\|_2 \to 1} \frac{\exp\left(-\frac{\|s'\|_2^2}{1-\|s'\|_2^2} \lambda_{max}(\Sigma)^{-1}/2\right)}{(1-\|s'\|_2^2)^{(m_1+2)/2}} \\ &= \lim_{\|s'\|_2 \to 1} \frac{\exp\left(-\frac{1}{1-\|s'\|_2^2} \lambda_{max}(\Sigma)^{-1}/2\right)}{(1-\|s'\|_2^2)^{(m_1+2)/2}} \\ &= \lim_{t \to +\infty} t^{(m_1+2)/2} \exp\left(-t \lambda_{max}(\Sigma)^{-1}/2\right) = 0. \end{aligned}$$

*This proves that all the derivatives are bounded uniformly over $s'$, which means that for*

*every* $\nu > 0$ *we can find a constant* $C_p^{SS}$ *such that* $\|p_h(s'|\cdot)\|_{W^{\nu,\infty}} \leq C_p^{SS}$. *The condition on the reward follows from the fact that the function* $s^\top Qs$ *is infinitely many times differentiable and has all its derivatives bounded over* $B_1(0)$.

While definition 20 is simple and intuitive, the idea of smoothness can be made more general by applying it directly to the Bellman operator, rather than separately to transitions and rewards. This is the subject of the next section.

### 4.3.1. Mildly and Weakly Smooth MDPs

As we saw in the introduction, in order to construct the optimal policy of an MDP, what is really necessary to know is the Bellman optimality operator, which we call $\mathcal{T}_h$ at the corresponding step. Knowing the reward and transition functions implies knowledge of this operator, but the vice versa is not necessarily true.

Therefore, in this section we will formalize the classes of mildly and weakly Smooth MDPs based on assumptions that only apply to this operator, to increase generality. As with definition 20, it will be necessary to enforce the smoothness of this operator through a norm. While a standard Sobolev-Hölder norm suffices for the reward functions, characterizing the transitions $\mathcal{T}_h$ necessitates the use of an appropriate operator norm.

For a linear operator $\mathcal{T} : X \to Y$, where $X, Y$ are normed spaces of norms $\|\cdot\|_X, \|\cdot\|_Y$ respectively, the operator norm writes as follows.

$$\|\mathcal{T}\|_{X\to Y} := \sup_{x\in X\setminus\{\mathbf{0}\}} \frac{\|\mathcal{T}x\|_Y}{\|x\|_X}.$$

However, the Bellman operator is *not* linear. In fact, for the zero function $\mathbf{0}$, $\mathcal{T}_h\mathbf{0} = r_h$ which is, in general, non-zero. Thus, the previous definition would imply that all Bellman operators have infinite norm. To solve this problem, we introduce the following "growth functional",

$$\|\mathcal{T}\|^\circ_{X\to Y} := \sup_{x\in X} \frac{\|\mathcal{T}x\|_Y}{\|x\|_X + 1}. \tag{4.12}$$

We still use $\|\cdot\|$ to recall where its definition comes from, but we add $\circ$ because formally the former is not of a norm (axioms of linearity is not respected, nor is the triangular inequality). Having a bound on $\|\mathcal{T}\|^\circ_{X\to Y}$ means that the operator does not transform element of a given norm to other elements whose norm exceeds $\|\mathcal{T}\|^\circ_{X\to Y}$ times one plus the initial quantity.

The norm of an operator depends on the norms $\|\cdot\|_X, \|\cdot\|_Y$ that one uses for the starting and arrival spaces. For what concerns the Bellman optimality operator, classical results in RL leverage the boundedness of this operator w.r.t. the $L^\infty(\mathcal{Z})$ norm. In this context however, we consider norms that come from Sobolev-Hölder spaces.

**Definition 21** (*Mildly* Smooth MDPs)**.** *The class $\mathscr{M}_{MS}$ of Mildly Smooth MDPs is the set of MDPs where $\mathcal{S} = [-1,1]^{m_1}, \mathcal{A} = [-1,1]^{m_2}$ and, for some $\nu \geq 0$ and for all $h \in [H]$ the Bellman optimality operator is bounded over $L^\infty(\mathcal{Z}) \to W^{\nu,\infty}(\mathcal{Z})$ so that*

$$\|\mathcal{T}_h\|^\circ_{L^\infty \to W^{\nu,\infty}} \leq C_{\mathcal{T}}^{MS}, \tag{4.13}$$

*for a proper constant $C_{\mathcal{T}}^{MS}$.*

Boundedness over $L^\infty(\mathcal{Z}) \to \mathrm{W}^{\nu,\infty}(\mathcal{Z})$ means that the operator transforms functions that are bounded (i.e., belong to $L^\infty(\mathcal{Z})$) into functions that are $\nu$-times differentiable (i.e., belong to $\mathrm{W}^{\nu,\infty}(\mathcal{Z})$). Moreover, there exists a constant $\mathrm{C}_{\mathcal{T}}^{\mathrm{MS}} < +\infty$ such that $\|\mathcal{T}_h f\|_{\mathrm{W}^{\nu,\infty}} \leq \mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}(\|f\|_{L^\infty} + 1)$ for every $h \in [H]$ and every function $f \in L^\infty(\mathcal{Z})$. Intuitively, this condition means that by applying the Bellman operator to bounded (possibly non-smooth) functions, we always get functions that are smooth.

It is not immediately clear whether this new class introduced in definition 21 is more or less general than the Strongly Smooth class that we introduced in definition 20. Let us start by proving the bandit case, that is, if $H = 1$, the imposed condition reduces to having a smooth reward. By definition,

$$\mathcal{T}_1 f(s,a) = r(s,a),$$

since the process terminates after only one time step. It follows that

$$\|\mathcal{T}_h f\|_{\mathrm{W}^{\nu,\infty}} = \|r\|_{\mathrm{W}^{\nu,\infty}} \leq \underbrace{\|r\|_{\mathrm{W}^{\nu,\infty}}}_{=:\mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}}(\|f\|_{L^\infty} + 1).$$

This means that this novel class of Mildly Smooth MDPs contains the Smooth bandits as a special case, which was a condition that also held for Strongly Smooth MDPs. In the next we show that actually the whole class of Strongly Smooth MDPs is included in the Mildly Smooth one.

**For fixed $\nu > 0$, every Strongly Smooth MDP of parameter $\nu$ is Mildly Smooth with the same parameter** Let us assume an MDP is Strongly Smooth. By definition,

for every function $f : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$:

$$\begin{aligned}\mathcal{T}_h f(s,a) &= r_h(s,a) + \mathop{\mathbb{E}}_{s' \sim p_h(\cdot|s,a)}[\max_{a' \in \mathcal{A}} f(s',a')] \\ &= r_h(s,a) + \int_{\mathcal{S}} \max_{a' \in \mathcal{A}} f(s',a') p_h(s'|s,a)\ ds'.\end{aligned}$$

By triangular inequality, this entails:

$$\|\mathcal{T}_h f\|_{\mathrm{W}^{\nu,\infty}} \leq \|r\|_{\mathrm{W}^{\nu,\infty}} + \left\| \int_{\mathcal{S}} \max_{a' \in \mathcal{A}} f(s',a') p_h(s'|\cdot)\ ds' \right\|_{\mathrm{W}^{\nu,\infty}},$$

where the first term $\|r\|_{\mathrm{W}^{\nu,\infty}}$ is bounded by assumption and so we can focus on the second one. As all the functions involved are bounded, the theorem of exchange between integral and derivative applies, proving that, for every multi-index with $|\alpha| \leq \nu$:

$$D^\alpha \int_{\mathcal{S}} \max_{a' \in \mathcal{A}} f(s',a') p_h(s'|\cdot)\ ds' = \int_{\mathcal{S}} \max_{a' \in \mathcal{A}} f(s',a') D^\alpha p_h(s'|\cdot)\ ds'. \tag{4.14}$$

Using the abbreviation $\tilde{f}(s') = \max_{a' \in \mathcal{A}} f(s',a')$, the following inequalities follow:

$$\begin{aligned}\left\| \int_{\mathcal{S}} \tilde{f}(s') p_h(s'|\cdot)\ ds' \right\|_{\mathrm{W}^{\nu,\infty}} &= \max_{|\alpha| \leq \nu} \left\| D^\alpha \int_{\mathcal{S}} \tilde{f}(s') p_h(s'|\cdot)\ ds' \right\|_{L^\infty} \\ &= \max_{|\alpha| \leq \nu} \left\| \int_{\mathcal{S}} \tilde{f}(s') D^\alpha p_h(s'|\cdot)\ ds' \right\|_{L^\infty} \\ &\leq \max_{|\alpha| \leq \nu} \mathrm{Vol}(\mathcal{S}) \sup_{s' \in \mathcal{S}} \|\tilde{f}(s') D^\alpha p_h(s'|\cdot)\|_{L^\infty} \\ &\leq \max_{|\alpha| \leq \nu} \mathrm{Vol}(\mathcal{S}) \sup_{s' \in \mathcal{S}} \|D^\alpha p_h(s'|\cdot)\|_{L^\infty} \|\tilde{f}\|_{L^\infty} \\ &\leq \max_{|\alpha| \leq \nu} \mathrm{Vol}(\mathcal{S}) \sup_{s' \in \mathcal{S}} \|D^\alpha p_h(s'|\cdot)\|_{L^\infty} \|f\|_{L^\infty} \\ &\leq 2^{m_1} \mathrm{C}_{\mathrm{p}}^{\mathrm{SS}} \|f\|_{L^\infty}.\end{aligned}$$

Where the first step is the definition of $\|\cdot\|_{\mathrm{W}^{\nu,\infty}}$, the second comes from exchanging integral and derivative, the third one by just bounding the integral with the infinity norm times the volume of the set, and the last ones by definitions of $\tilde{f}$ and $\mathrm{C}_{\mathrm{p}}^{\mathrm{SS}}$. Together, this proves that

$$\|\mathcal{T}_h f\|_{\mathrm{W}^{\nu,\infty}} \leq \mathrm{C}_{\mathrm{r}}^{\mathrm{SS}} + 2^{m_1} \mathrm{C}_{\mathrm{p}}^{\mathrm{SS}} \|f\|_{L^\infty} = \underbrace{\max(\mathrm{C}_{\mathrm{r}}^{\mathrm{SS}}, 2^{m_1} \mathrm{C}_{\mathrm{p}}^{\mathrm{SS}})}_{=: \mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}} (1 + \|f\|_{L^\infty}).$$

The former equation proves that every Strongly Smooth MDP is also Mildly Smooth for the same parameter $\nu$. The exponential constant $2^{m_1}$ comes from the normalization over $[-1,1]^m$, whose volume grows exponentially with $m$. As pointed in section 4.2.3, in this context an exponential scaling in $m$ is still considered acceptable.

After proving this inclusion relationship between these two classes of smooth processes, it is natural to ask as follow-up question if the inclusion is strict. The answer is yes, as the following paragraph shows.

**For $\nu = 1$, there is an MDP that is $\nu-$Mildly Smooth but not $\nu-$Strongly Smooth.** Define an MDP where

- $\mathcal{S} = \mathcal{A} = [-1,1]$, so that $\mathcal{Z} = [-1,1]^2$.
- $r_h(z) =$ any smooth function.
- $p_h(\cdot|s,a) = \text{Unif}(\beta s, \beta s + 1 - \beta)$ for some $\beta \in (0,1)$.

Note that this transition function is well defined, as both $\beta s, \beta s + 1 - \beta$ are in $[-1,1]$ but it is not even continuous, as its density corresponds to

$$p_h(s'|s,a) = \frac{\mathbf{1}_{[\beta s, \beta s+1-\beta]}(s')}{1-\beta}.$$

Therefore, the corresponding MDP is *not* Strongly Smooth. Still, we can show that this process is Mildly smooth for $\nu = 1$. Indeed, take every $f \in L^\infty$,

$$\|\mathcal{T}_h f\|_{\mathrm{W}^{1,\infty}} = \max_{|\alpha|\leq 1} \|D^\alpha \mathcal{T}_h f\|_{L^\infty} = \max\left\{\|\mathcal{T}_h f\|_{L^\infty}, L_1(\mathcal{T}_h f)\right\},$$

where $L_1(\mathcal{T}_h f)$ denotes the Lipschitz constant. The first part $\|\mathcal{T}_h f\|_{L^\infty}$ is bounded as

$$\|\mathcal{T}_h f\|_{L^\infty} \leq \|r_h\|_{L^\infty} + \|f\|_{L^\infty}.$$

The second one corresponds to

$$L_1(\mathcal{T}_h f) = \sup_{z_1, z_2 \in \mathcal{Z}} \frac{\mathcal{T}_h f(z_1) - \mathcal{T}_h f(z_2)}{\|z_1 - z_2\|_\infty}.$$

Calling $\tilde{f}(s') = \max_{a' \in \mathcal{A}} f(s', a')$ (recall that reward is constant),

$$\sup_{z_1,z_2\in\mathcal{Z}} \frac{\mathcal{T}_h f(z_1) - \mathcal{T}_h f(z_2)}{\|z_1 - z_2\|_\infty} = \sup_{z_1,z_2\in\mathcal{Z}} \frac{\int_{\mathcal{S}} \tilde{f}(s') p_h(s'|z_1)\, ds' - \int_{\mathcal{S}} \tilde{f}(s') p_h(s'|z_2)\, ds'}{\|z_1 - z_2\|_\infty}$$
$$= \sup_{z_1,z_2\in\mathcal{Z}} \frac{\int_{\mathcal{S}} \tilde{f}(s')(p_h(s'|z_1) - p_h(s'|z_2))\, ds'}{\|z_1 - z_2\|_\infty}.$$

The former allows to evaluate the numerator explicitly:

$$\int_{\mathcal{S}} \tilde{f}(s')(p_h(s'|s_1,a_1) - p_h(s'|s_2,a_2))\, ds' = \int_{\mathcal{S}} \tilde{f}(s') \left( \frac{\mathbf{1}_{[\beta s_1, \beta s_1 + 1 - \beta]}(s')}{1-\beta} - \frac{\mathbf{1}_{[\beta s_2, \beta s_2 + 1 - \beta]}(s')}{1-\beta} \right) ds'$$
$$= \int_{\mathcal{S}} \tilde{f}(s') \frac{\mathbf{1}_{[\beta s_1, \beta s_1 + 1 - \beta]}(s')}{1-\beta}\, ds'$$
$$- \int_{\mathcal{S}} \tilde{f}(s') \frac{\mathbf{1}_{[\beta s_2, \beta s_2 + 1 - \beta]}(s')}{1-\beta}\, ds'$$
$$= \frac{1}{1-\beta} \left( \int_{\beta s_1}^{\beta s_1 + 1 - \beta} \tilde{f}(s')\, ds' - \int_{\beta s_2}^{\beta s_2 + 1 - \beta} \tilde{f}(s')\, ds' \right).$$

Now, note that the function $g(s) := \int_{\beta s}^{\beta s + 1 - \beta} \tilde{f}(s')\, ds'$ has a derivative bounded by $2\|\tilde{f}\|_{L^\infty}$ in absolute value: from the fundamental theorem of calculus,

$$|g'(s)| = |\tilde{f}(\beta s + 1 - \beta) - \tilde{f}(\beta s)| \leq 2\|\tilde{f}\|_{L^\infty},$$

so that it is $2\|\tilde{f}\|_{L^\infty}$ Lipschitz continuous. Substituting in the previous equation we get

$$\int_{\mathcal{S}} \tilde{f}(s')(p_h(s'|s_1,a_1) - p_h(s'|s_2,a_2))\, ds' \leq \frac{1}{1-\beta} \left( \int_{\beta s_1}^{\beta s_1 + 1 - \beta} \tilde{f}(s')\, ds' - \int_{\beta s_2}^{\beta s_2 + 1 - \beta} \tilde{f}(s')\, ds' \right)$$
$$= \frac{1}{1-\beta} (g(s_1) - g(s_2)) \leq \frac{2\|\tilde{f}\|_{L^\infty}}{1-\beta} \|s_1 - s_2\|_{L^\infty}.$$

This proves that

$$\sup_{z_1,z_2\in\mathcal{Z}} \frac{\mathcal{T}_h f(z_1) - \mathcal{T}_h f(z_1)}{\|z_1 - z_2\|_\infty} = \sup_{z_1,z_2\in\mathcal{Z}} \frac{\int_{\mathcal{S}} \tilde{f}(s')(p_h(s'|z_1) - p_h(s'|z_2))\, ds'}{\|z_1 - z_2\|_\infty}$$
$$\leq \sup_{z_1,z_2\in\mathcal{Z}} \frac{\frac{2\|\tilde{f}\|_{L^\infty}}{1-\beta} \|s_1 - s_2\|_\infty}{\|z_1 - z_2\|_\infty}$$
$$= \frac{2\|\tilde{f}\|_{L^\infty}}{1-\beta} \leq \frac{2}{1-\beta} \|f\|_{L^\infty}.$$

Where the passage after the first inequality holds since $\|s_1 - s_2\|_\infty \leq \|z_1 - z_2\|_\infty$ (equality holds taking the supremum over $z_1, z_2$ as it is sufficient to have $a_1 = a_2$ to enforce exactly $\|s_1 - s_2\|_\infty = \|z_1 - z_2\|_\infty$) and the second is due to the fact that being $\tilde{f}(s') = \max_{a' \in \mathcal{A}} f(s', a')$ we have $\|\tilde{f}\|_{L^\infty} \leq \|f\|_{L^\infty}$. These passages show that

$$\begin{aligned}\|\mathcal{T}_h f\|_{\mathrm{W}^{1,\infty}} &\leq \max\left\{\|r_h\|_{L^\infty} + \|f\|_{L^\infty}, \frac{2}{1-\beta}\|f\|_{L^\infty}\right\} \\ &\leq \max\left\{\|r_h\|_{L^\infty}, \frac{2}{1-\beta}\right\}(1 + \|f\|_{L^\infty}).\end{aligned}$$

This proves that the MDP is Mildly Smooth for $\nu = 1$, while we have seen that this is not Strongly Smooth for any $\nu > 0$.

The Mildly Smooth MDP family is of even broader interest than its "strong" counterpart. As shown in the previous example, based on an MDP with simple, uniform density noise, the strong smoothness assumption fails when the density function is discontinuous. One the other hand, the Mildly Smooth condition can be satisfied even in these cases. Is it possible to generalize further than Mildly Smooth processes without compromising our core idea of smoothness? The answer is yes, by further relaxing the fundamental assumption in definition 21.

**Definition 22** (*Weakly* Smooth MDPs)**.** *The class $\mathscr{M}$ of Weakly Smooth MDPs is the set of MDPs where $\mathcal{S} = [-1, 1]^{m_1}, \mathcal{A} = [-1, 1]^{m_2}$ and, for some $\nu \geq 0$ and for all $h \in [H]$ the Bellman optimality operator is bounded over $W^{\nu,\infty}(\mathcal{Z}) \to W^{\nu,\infty}(\mathcal{Z})$ so that*

$$\|\mathcal{T}_h\|^\circ_{W^{\nu,\infty} \to W^{\nu,\infty}} \leq C^{WS}_{\mathcal{T}}, \tag{4.15}$$

*for a proper constant $C^{WS}_{\mathcal{T}}$.*

At first glance, not much has changed. The very subtle, but crucial, difference lies in the pair of norms under which the assumption of boundedness of the operator norm holds. No longer $L^\infty(\mathcal{Z})$ as the starting space, but $\mathrm{W}^{\nu,\infty}(\mathcal{Z})$. Boundedness over $\mathrm{W}^{\nu,\infty}(\mathcal{Z}) \to \mathrm{W}^{\nu,\infty}(\mathcal{Z})$ means that the operator must output a function that is $\nu$ times differentiable when receiving a function from the same set.

As before, we now need to prove that this class is more general than the previous ones, that is, that every MDP belonging to the Mildly Smooth class also belongs to this one. Fortunately, this time the task is very easy: by definition we know that for every $f$ we have $\|f\|_{L^\infty} \leq \|f\|_{\mathrm{W}^{\nu,\infty}}$. At this point, given any MDP with Bellman operator $\mathcal{T}$ that

satisfies definition 21, the following inequalities hold at any stage $h$

$$\begin{aligned}\|\mathcal{T}_h\|^{\circ}_{\mathrm{W}^{\nu,\infty}\to\mathrm{W}^{\nu,\infty}} &= \sup_{f\in\mathrm{W}^{\nu,\infty}(\mathcal{Z})} \frac{\|\mathcal{T}_h f\|_{\mathrm{W}^{\nu,\infty}}}{\|f\|_{\mathrm{W}^{\nu,\infty}}+1} \\ &\leq \sup_{f\in\mathrm{W}^{\nu,\infty}(\mathcal{Z})} \frac{\|\mathcal{T}_h f\|_{\mathrm{W}^{\nu,\infty}}}{\|f\|_{L^\infty}+1} = \|\mathcal{T}_h\|^{\circ}_{L^\infty\to\mathrm{W}^{\nu,\infty}}.\end{aligned}$$

Therefore, not only any MDP that is Mildly Smooth is also Weakly Smooth, but when restricting to the smaller class the constant does not change, $\mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} = \mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}$.

The inclusion between these two families of MDPs is strict, as will be shown in the next section. There, we will compare our smoothness assumptions with all classes of state-space and continuous-action MDPs found in the literature, and the difference between the Mildly and Weakly Smooth assumptions will become clear.

Before proceeding, it is worth clarifying a unifying property that characterizes all variants of the Smooth MDP family. We have already established that for $H = 1$, Definitions 20, 21, and 22 all collapse to the Smooth Bandit case. Furthermore, for an arbitrary horizon $h$, these families share the fundamental property of reward smoothness. In the Strongly Smooth case (definition 20), it follows directly from the definition that $\|r_h\|_{\mathrm{W}^{\nu,\infty}}$ is bounded by a constant (specifically, $\mathrm{C}_{\mathrm{r}}^{\mathrm{SS}}$). In the Mildly Smooth case, the same result is obtained by considering the zero function $\mathbf{0}$:

$$\|r_h\|_{\mathrm{W}^{\nu,\infty}} = \|\mathcal{T}_h\mathbf{0}\|_{\mathrm{W}^{\nu,\infty}} \leq \mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}(1+\|\mathbf{0}\|_{L^\infty}) = \mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}.$$

The same reasoning applies to the Weakly Smooth family, as the zero function has a null norm in any normed space. From this perspective, all proposed variants of Smooth MDPs can be viewed as intermediate classes, positioned between Smooth Bandits (which, as shown, admit efficient learning) and the broader class of MDPs with smooth rewards (which, conversely, are too general to allow for theoretical guarantees).

$$\text{Smooth bandits} \subset \begin{bmatrix}\text{Strongly}\\ \text{Mildly}\\ \text{Weakly}\end{bmatrix} \text{Smooth MDPs} \subset \text{MDPs with smooth reward.}$$

### 4.3.2. Practical Relevance of Smooth MDPs

While this treatment focuses on the theoretical study of Smooth MDPs, it is worth noting that assumptions closely related to those considered in this work are frequently implicit in practical Reinforcement Learning (RL) research. In particular, in domains such as

autonomous aircraft and underwater vehicle control [23, 40], the inherent smoothness of the underlying dynamics is often exploited to train policies that, beyond maximizing rewards, enforce a certain degree of regularity. This synergy arises naturally from the fundamental physics governing such systems.

As shown in [93], learning smooth policies can lead to improved aircraft engine performance and substantial fuel savings. Their approach relies on penalizing high-frequency components of the controller's Fourier transform, a technique known to be equivalent to bounding the function's norm in a Sobolev space.

More generally, recent works on policy gradient methods attribute their strong empirical performance to the smoothness of the objective function in parameter space [90, 91, 99]. This smoothness is often derived from regularity assumptions on the underlying MDP (see Assumption 4.2 in [90]), which closely resemble the Strongly Smooth MDP framework.

These results suggest that the theory of Smooth MDPs captures key structural properties of real-world control problems, providing a principled explanation for empirical phenomena observed in modern RL.

## 4.4. Hierarchy of Continuous Space MDP Assumptions

In the remainder of this chapter, we have explored most of the settings in the literature that address the problem of reinforcement learning on continuous spaces, proposing algorithms with theoretical performance guarantees.

In this section, we try to sort out this ocean of diverse assumptions to determine which areas are covered by which others. If one were to analyze each setting in its entirety, the result would be that most of them are not comparable. For this reason, the comparison will be made under some simplifications. In particular:

- Differences in the state/action space are not going to be considered. This only concerns the generalizations of the LQR, which assume $\mathcal{S} = \mathbb{R}^{m_1}$, while the other classes (including Smooth MDPs) usually assume compactness (i.e. $\mathcal{Z} = \mathcal{S} \times \mathcal{A} = [-1, 1]^m$).
- For MDP classes involving feature maps, we sometimes assume the mapping is smooth. Fortunately, this is largely true for maps used for practical problems. Maps used in classical statistics, such as polynomials, Fourier features, and wavelets, are infinitely differentiable, so they belong to $\mathrm{W}^{\nu,\infty}$ for every $\nu > 0$. The same does not

hold for splines; nevertheless, they are differentiable up to a given degree (usually 2). If we look at modern machine learning, feature extraction is typically done via a neural network. If we consider a feature map given by a feed-forward neural network, the smoothness depends solely on the activation functions. `sigmoid` and `tanh` are infinitely differentiable, while `relu` is not, but still belongs to $W^{1,\infty}$ because it is Lipschitz. Since the linear layers are also infinitely differentiable, as are the post-training `Batch Normalization` layers, the network inherits the same level of smoothness.

- As for Kernelized methods, our focus will be on the Matérn kernel family, which are the most used in the case of problems on continuous spaces. This is not uncommon in the literature, and there are papers on kernelized MDPs that specifically focus on this class [51].

Let us start by analyzing the LQR and its generalizations. As already observed, the transition function of the LQR takes the form

$$p_h(s'|s,a) \propto \exp\left(-\frac{(s'-As-Ba)^\top \Sigma^{-1}(s'-As-Ba)}{2}\right),$$

which is infinitely differentiable. Similarly, the reward function, which the LQR assumes to be quadratic, is also infinitely times differentiable. Therefore, the LQR model satisfies definition 20 of Smooth MDP.

Are the generalizations of this assumption about transition equally smooth? The setting presented as NR (Nonlinear Regulator, defined by [56]) assumes the following form for the transition distribution.

$$p_h(s'|s,a) \propto \exp\left(-\frac{(s'-W\varphi(s,a))^\top \Sigma^{-1}(s'-W\varphi(s,a))}{2}\right).$$

For every $s'$, the exterior function is infinitely smooth. Furthermore, the product by the matrix $W$ is a linear function. Therefore, the smoothness of the global transition function is equal to that of the feature map $\varphi$. For the most common choices, this transition function thus satisfies eq. (4.11) for every $\nu \geq 0$.

Studying the Bilinear Exponential Family, which is the subject of the papers [27][97] is more complex. In this case, unlike the previous two, the Gaussian noise assumption, which guaranteed smoothness in the previous cases, is dropped. Here, the transition function

takes the following form:

$$p(s'|s,a) = \exp\left(\psi(s')^\top W\phi(s,a) - Z(s,a)\right),$$

where $Z(s,a)$ is a normalization constant. While the function $\exp\left(\psi(s')^\top W\phi(s,a) - Z(s,a)\right)$ in as smooth as $\phi$ in $s,a$, there is an additional term $\exp(-Z(s,a))$ to take into consideration. Letting $\alpha$ be any multi index,

$$\begin{aligned} D^\alpha e^{Z(s,a)} &= D^\alpha \int_{\mathcal{S}} \exp\left(\psi(s')^\top W\phi(s,a)\right) ds' \\ &= \int_{\mathcal{S}} D^\alpha \exp\left(\psi(s')^\top W\phi(s,a)\right) ds' \\ &= \int_{\mathcal{S}} \psi(s')^\top W(D^\alpha\phi(s,a)) \exp\left(\psi(s')^\top W\phi(s,a)\right) ds' \\ &= \left(\int_{\mathcal{S}} \psi(s') \exp\left(\psi(s')^\top W\phi(s,a)\right) ds'\right)^\top W D^\alpha\phi(s,a). \end{aligned}$$

Therefore, unless $\left\|\int_{\mathcal{S}} \psi(s') \exp\left(\psi(s')^\top W\phi(s,a)\right) ds'\right\|_2 = +\infty$, the function $e^{Z(s,a)}$ is differentiable as many times as $\phi$. If one assumes finite $\|\psi(s')\|_2$ for every $s'$, the other term is $\exp\left(\psi(s')^\top W\phi(s,a)\right)$, which is positive for every $s'$ and integrates to $e^{Z(s,a)} < +\infty$ by assumption. Now, as the exponential map is positive, strictly increasing and infinitely times differentiable $e^{-Z(s,a)}$ and $e^{Z(s,a)}$ have the same number of bounded derivatives. Therefore, it follows that, whenever $\phi \in \mathrm{W}^{\nu,\infty}(\mathcal{Z})$ and $\|\psi(s')\|_2$ is bounded,

$$p(s'|s,a) = \underbrace{\exp\left(\psi(s')^\top W\phi(s,a)\right)}_{\mathrm{W}^{\nu,\infty}(\mathcal{Z})} \underbrace{\exp(-Z(s,a))}_{\mathrm{W}^{\nu,\infty}(\mathcal{Z})} \in \mathrm{W}^{\nu,\infty}(\mathcal{Z}).$$

Interestingly, the parameter $\nu$ depends only on the smoothness of $\phi$. Therefore, while it is necessary to assume the smoothness of $\phi$, the other map $\psi$ is allowed to be a general bounded function.

We have thus shown that in both cases the MDP is as smooth as its feature map $\phi$. In the cases we mentioned earlier, such as polynomials, Fourier features, and neural networks with `tanh` activation, the map is differentiable an arbitrary number of times. However, note that the precise value of $\mathrm{C}_{\mathrm{p}}^{\mathrm{SS}}$ depends only on $\varphi$ and the norm of all the matrices involved, and could grow very quickly with $\nu \to \infty$. Therefore, even for infinitely differentiable maps, taking the largest possible parameter $\nu$is not always the best choice.

### 4.4.1. Non-parametric Classes

The previous section demonstrated that LQRs and their generalizations satisfy the Strongly Smooth MDP assumption, provided the feature maps meet mild regularity conditions.We now shift focus to non-parametric classes: Lipschitz and Kernelized MDPs. These offer a more compelling theoretical comparison; unlike the unbounded state spaces ($\mathbb{R}^{m_1}$) typical of the previous models, Lipschitz and Kernelized MDPs are defined on compact spaces (i.e., $[-1,1]^m$). This alignment makes them natively compatible with our framework.

**Kernelized MDPs are Strongly Smooth** First, it is interesting to determine the relation between Kernelized and Strongly Smooth MDPs. As already pointed out, the two assumptions write in a very similar way. A Kernelized MDP is defined by

$$\|r_h(\cdot)\|_{\mathcal{H}} \leq 1, \qquad \|p_h(s'|\cdot)\|_{\mathcal{H}} \leq 1 \qquad \mathcal{H} = \mathcal{H}(\mathscr{K}_\nu), \tag{4.16}$$

where $\mathscr{K}_\nu$ is the Matérn Kernel of parameter $\nu$. Instead our Strongly Smooth MDP class assumes

$$\|r_h\|_{\mathrm{W}^{\nu,\infty}} \leq \mathrm{C}_{\mathrm{r}}^{\mathrm{SS}}, \qquad \|p_h(s'|\cdot)\|_{\mathrm{W}^{\nu,\infty}} \leq \mathrm{C}_{\mathrm{p}}^{\mathrm{SS}}.$$

Given the definitions of the two MDP families, this point reduces to proving that the RKHS generated by the Matérn kernel of parameter $\nu > 0$ is a subspace of $\mathrm{W}^{\nu,\infty}(\mathcal{Z})$, with a bound on the norm.

We start proving that there is an inclusion between the spaces. Theorem 8, part (3) from [30] ensures that, for isotropic kernels (that write as $\mathscr{K}(x,y)$ only depends on the distance of $x,y$), in order to have sample path $\in \mathrm{W}^{\nu,\infty}(\mathcal{Z})$, a sufficient condition is for the kernel to be in $\mathrm{W}^{\nu,\infty}(\mathcal{Z}^2)$. The same paper also shows (proof of Proposition 10) that the Matérn kernel can be written as

$$\mathscr{K}_\nu(x,y) \propto (f_1(\|x-y\|_2^2) + \|x-y\|_2^{2\nu} f_2(\|x-y\|_2^2)),$$

where $f_1, f_2 \in \mathcal{C}^\infty(\mathbb{R})$. Note that,

$$\|x-y\|_2^2 = \sum_{i=1}^{m} (x_i - y_i)^2 \in \mathcal{C}^\infty(\mathcal{Z}^2),$$

so that $f_1(\|\cdot - \cdot\|_2^2), f_2(\|\cdot - \cdot\|_2^2) \in \mathcal{C}^\infty(\mathcal{Z}^2)$. Therefore, these two terms do not affect the overall smoothness of the kernel: the order of smoothness of $\mathscr{K}_\nu(x,y)$ corresponds to the one of $\|x-y\|_2^{2\nu}$ which is indeed in $\mathrm{W}^{2\nu,\infty}(\mathcal{Z}^2)$.

After proving that $\mathcal{H}(\mathscr{K}_\nu) \subset \mathrm{W}^{2\nu,\infty}(\mathcal{Z}^2)$, it is necessary to show norm upper bound. Two well-known facts are the key for this proof: i) both are Banach Spaces ii) both norms are stronger than the $L^2$ one. Now, let us take the identity operator $I : \mathcal{H}(\mathscr{K}_\nu) \to \mathrm{W}^{2\nu,\infty}(\mathcal{Z}^2)$. Let $f_n \subset \mathcal{H}(\mathscr{K}_\nu)$ be a sequence of functions such that $f_n \overset{\mathcal{H}(\mathscr{K}_\nu)}{\to} f$ and $If_n \overset{\mathrm{W}^{2\nu,\infty}}{\to} g$. By definition of the identity operator, $If_n = f_n$, so that both $\|f_n - f\|_{\mathcal{H}(\mathscr{K}_\nu)} \to 0$ and $\|f_n - g\|_{\mathrm{W}^{2\nu,\infty}} \to 0$. Using (ii), this entails that $\|f_n - f\|_{L^2} \to 0$ and $\|f_n - g\|_{L^2} \to 0$, so that we must have $f = g$.

In this way, we have just proved that the identity operator $I : \mathcal{H}(\mathscr{K}_\nu) \to \mathrm{W}^{2\nu,\infty}(\mathcal{Z}^2)$ enjoys the *closed graph property*. As by (i), both $\mathcal{H}(\mathscr{K}_\nu), \mathrm{W}^{2\nu,\infty}(\mathcal{Z}^2)$ are Banach spaces, the closed graph theorem ensures that the identity operator is continuous, namely that there is some constant $\mathrm{C}_\nu$ such that $\|f\|_{\mathrm{W}^{2\nu,\infty}} = \|If\|_{\mathrm{W}^{2\nu,\infty}} \leq \mathrm{C}_\nu \|f\|_{\mathcal{H}(\mathscr{K}_\nu)}$. With this argument, we have just proved the following proposition.

**Proposition 28.** *Let us fix $\nu > 0$ and call $\mathscr{M}_{kernelized}$ the class of Kernelized MDPs with Matérn kernel of parameter $\nu$. Then, $\mathscr{M}_{kernelized}$ is a subset of the class $\mathscr{M}_{SS}$ of Strongly Smooth MDPs with the same parameter $\nu$ and constant $C_\nu$.*

One could ask if the opposite property also holds, so that the two classes coincide. It reveals that this is not the case, even for very simple MDPs. Corollary A.6 from [132] shows that, for $m = 2$, the RKHS $\mathcal{H}_{\mathscr{K}_1}(\mathcal{Z}) = \mathrm{W}^{2,2}(\mathcal{Z})$ the functions with square-integrable second derivative. Take an MDP $\mathcal{M}$ with $\mathcal{S} = \mathcal{A} = [-1, 1]$, smooth transition and reward function

$$r_h(s, a) = \sqrt{s^2 + a^2} = \|z\|_2.$$

$\mathcal{M}$ is Strongly Smooth MDP for $\nu = 1$, as $r_h$ is Lipschitz continuous. However, the second derivative of $r_h \notin \mathrm{W}^{2,2}(\mathcal{Z})$, as the square Hessian matrix determinant behaves as $\frac{1}{x^2+y^2}$, which is not integrable on $[-1, 1]^2$. Thus, $\mathcal{M}$ is not a Kernelized MDP for $\nu = 1$.

**TV-Lipschitz MDPs are Mildly Smooth with $\nu = 1$** As seen in the section about Lipschitz MDPs, there is a significant difference in generality between TV-Lipschitz and $\mathcal{W}-$Lipschitz MDPs. Let us start with the first, less general case.

We previously proved that for $\nu = 1$, there is an MDP that is Mildly Smooth but not Strongly Smooth. The same example shows that there exist TV-Lipschitz MDPs that are not Strongly Smooth for any $\nu > 0$. However, inclusion occurs in the broader class of Mildly Smooth MDPs.

Consider a TV-Lipschitz MDP with reward $r_h$ and transition $p_h$. By assumption, there

are constants $L_r, L_p$ such that, for all $h \in [H], s, s' \in \mathcal{S}, a, a' \in \mathcal{A}$

$$|r_h(s,a) - r_h(s',a')| \leq L_r(\|s-s'\|_2 + \|a-a'\|_2), \tag{4.17}$$
$$\mathrm{TV}(p_h(\cdot|s,a), p_h(\cdot|s',a')) \leq L_p(\|s-s'\|_2 + \|a-a'\|_2). \tag{4.18}$$

To prove that $\mathcal{M}$ is Mildly Smooth means upper-bounding $\|\mathcal{T}_h\|^{\circ}_{L^\infty \to \mathrm{W}^{1,\infty}}$.

$$\begin{aligned}\|\mathcal{T}_h\|^{\circ}_{L^\infty \to \mathrm{W}^{1,\infty}} &= \sup_{f \in L^\infty(\mathcal{Z})} \frac{\left\| r_h(\cdot) + \int_{\mathcal{S}} \max_{a' \in \mathcal{A}} f(s',a') p_h(s'|\cdot) ds' \right\|_{\mathrm{W}^{1,\infty}}}{1 + \|f\|_{L^\infty}} \\ &\leq \sup_{f \in L^\infty(\mathcal{Z})} \frac{\|r_h\|_{\mathrm{W}^{1,\infty}} + \left\| \int_{\mathcal{S}} \max_{a' \in \mathcal{A}} f(s',a') p_h(s'|\cdot) ds' \right\|_{\mathrm{W}^{1,\infty}}}{1 + \|f\|_{L^\infty}}.\end{aligned}$$

Observe that the $\|r_h\|_{\mathrm{W}^{1,\infty}}$ term is bounded by $L_r$, by assumption. For what concerns the second one, said $g(s') := \max_{a' \in \mathcal{A}} f(s',a')$, we have $\|g\|_{L^\infty} \leq \|f\|_{L^\infty}$. The previous becomes

$$\|\mathcal{T}_h\|^{\circ}_{L^\infty \to \mathrm{W}^{1,\infty}} \leq \sup_{g \in L^\infty(\mathcal{S})} \frac{\|r_h\|_{\mathrm{W}^{1,\infty}} + \left\| \int_{\mathcal{S}} g(s') p_h(s'|\cdot) ds' \right\|_{\mathrm{W}^{1,\infty}}}{1 + \|g\|_{L^\infty}}.$$

Focus now on the term $\left\| \int_{\mathcal{S}} g(s') p_h(s'|\cdot) ds' \right\|_{\mathrm{W}^{\nu,\infty}}$. By definition of the norm, it follows that (see B.1) $\|f\|_{\mathrm{W}^{1,\infty}} = \max\{\|f\|_{L^\infty}, \|D^\alpha f\|_{L^\infty}\}$, where $\alpha$ denotes a multi-index whose values sum up to one. The following upper bound holds for the first part.

$$\left\| \int_{\mathcal{S}} g(s') p_h(s'|\cdot) ds' \right\|_{L^\infty} \leq \|g\|_{L^\infty} \left\| \int_{\mathcal{S}} p_h(s'|\cdot) ds' \right\|_{L^\infty} = \|g\|_{L^\infty}.$$

Then, we upper bound the second one.

$$\begin{aligned}\left\| D^\alpha \int_{\mathcal{S}} g(s') p_h(s'|\cdot) ds' \right\|_{L^\infty} &\leq \sup_{z \in \mathcal{Z}} \sup_{\Delta z} \frac{\left| \int_{\mathcal{S}} g(s') \left( p_h(s'|z) - p_h(s'|z+\Delta z) \right) ds' \right|}{\|\Delta z\|_2} \\ &\leq \sup_{z \in \mathcal{Z}} \sup_{\Delta z} \frac{\|g\|_{L^\infty} \mathrm{TV}\left( p_h(\cdot|z), p_h(\cdot|z+\Delta z) \right)}{\|\Delta z\|_2} \\ &\leq \sup_{z \in \mathcal{Z}} \sup_{\Delta z} \frac{\|g\|_{L^\infty} \|\Delta z\|_2 L_p}{\|\Delta z\|_2} = \|g\|_{L^\infty} L_p.\end{aligned}$$

The crucial passage is the second one: from the dual formulation of the total variation distance, it follows that for probability distributions $\mu, \rho$ over a set $\mathcal{X}$ the equation $\mathrm{TV}(\mu,\rho) = \sup_{\|f\|_{L^\infty}=1} \int_{\mathcal{X}} f(x)(\mu(x) - \rho(x)) dx$ holds. The next passage is just due to the definition of TV-Lipschitz MDP. This proves that

$$\|\mathcal{T}_h\|^{\circ}_{L^\infty \to \mathrm{W}^{1,\infty}} \leq \sup_{g \in L^\infty(\mathcal{S})} \frac{L_r + \|g\|_{L^\infty}(1 + \max\{1, L_p\})}{1 + \|g\|_{L^\infty}} = \max\{1, L_r, L_p\}.$$

This result is formalized in the following proposition.

**Proposition 29.** *The class $\mathscr{M}_{\mathrm{TV}}$ of Lipschitz MDPs with total variation distance is a subset of the Mildly Smooth MDPs with parameter $\nu = 1$ and constant $C_{\mathcal{T}}^{MS} = \max\{1, L_r, L_p\}$.*

Another wide family of well-studied MDPs with continuous state and action spaces is included into the Mildly Smooth family. We now turn to the overall most general case: the $\mathcal{W}-$Lipschitz MDPs.

**$\mathcal{W}-$Lipschitz MDPs are Weakly Smooth with $\nu = 1$** The previous result leaves us with the message that the class of Mildly Smooth MDPs contains $\mathscr{M}_{\mathrm{TV}}$. Moreover, it was already known that the same class is also contained in $\mathscr{M}_{\mathcal{W}}$, i.e., Lipschitz MDPs using the Wasserstein norm. $\mathscr{M}_{\mathcal{W}}$ is too large to belong to the Mildly Smooth MDPs. While providing a counterexample is complex, we will later demonstrate that the upper bound on regret for the Mildly Smooth class violates the lower bound established for $\mathscr{M}_{\mathcal{W}}$.

Nonetheless, we can prove that $\mathscr{M}_{\mathcal{W}}$ is included in the larger class of Weakly Smooth MDPs, again for $\nu = 1$. As before, this means upper-bounding $\|\mathcal{T}_h\|^{\circ}_{\mathrm{W}^{1,\infty}\to\mathrm{W}^{1,\infty}}$.

$$\begin{aligned}\|\mathcal{T}_h\|^{\circ}_{\mathrm{W}^{1,\infty}\to\mathrm{W}^{1,\infty}} &= \sup_{f\in\mathrm{W}^{1,\infty}(\mathcal{Z})} \frac{\left\|r_h(\cdot) + \int_{\mathcal{S}} \max_{a'\in\mathcal{A}} f(s',a')p_h(s'|\cdot)ds'\right\|_{\mathrm{W}^{1,\infty}}}{1+\|f\|_{\mathrm{W}^{1,\infty}}} \\ &\leq \sup_{f\in\mathrm{W}^{1,\infty}(\mathcal{Z})} \frac{\|r_h\|_{\mathrm{W}^{1,\infty}} + \left\|\int_{\mathcal{S}} \max_{a'\in\mathcal{A}} f(s',a')p_h(s'|\cdot)ds'\right\|_{\mathrm{W}^{1,\infty}}}{1+\|f\|_{\mathrm{W}^{1,\infty}}}.\end{aligned}$$

Observe that the $\|r_h\|_{\mathrm{W}^{1,\infty}}$ term is bounded by $L_r$, by assumption and, for what concerns the second one, said $g(s') := \max_{a'\in\mathcal{A}} f(s',a')$, one has $\|g\|_{\mathrm{W}^{1,\infty}} \leq \|f\|_{\mathrm{W}^{1,\infty}}$ (this property holds since $\mathrm{W}^{1,\infty}$ is a lattice, but would fail in $\mathrm{W}^{\nu,\infty}$ for $\nu > 1$). The previous becomes

$$\|\mathcal{T}_h\|^{\circ}_{\mathrm{W}^{1,\infty}\to\mathrm{W}^{1,\infty}} \leq \sup_{g\in L^{\infty}(\mathcal{S})} \frac{\|r_h\|_{\mathrm{W}^{1,\infty}} + \left\|\int_{\mathcal{S}} g(s')p_h(s'|\cdot)ds'\right\|_{\mathrm{W}^{1,\infty}}}{1+\|g\|_{\mathrm{W}^{1,\infty}}}.$$

As before, we use the definition of the norm $\mathrm{W}^{1,\infty}$ to bound the second part of the numerator. First, The following is an upper bound for the first part.

$$\left\|\int_{\mathcal{S}} g(s')p_h(s'|\cdot)ds'\right\|_{L^{\infty}} \leq \|g\|_{L^{\infty}} \left\|\int_{\mathcal{S}} p_h(s'|\cdot)ds'\right\|_{L^{\infty}} = \|g\|_{L^{\infty}} \leq \|g\|_{\mathrm{W}^{1,\infty}}.$$

Then, we upper bound the second one.

$$\begin{aligned}\left\| D^\alpha \int_{\mathcal{S}} g(s') p_h(s'|\cdot) ds' \right\|_{L^\infty} &\leq \sup_{z \in \mathcal{Z}} \sup_{\Delta z} \frac{\left| \int_{\mathcal{S}} g(s') \left( p_h(s'|z) - p_h(s'|z + \Delta z) \right) ds' \right|}{\|\Delta z\|_2} \\ &\leq \sup_{z \in \mathcal{Z}} \sup_{\Delta z} \frac{L(g) \mathcal{W}\left( p_h(\cdot|z), p_h(\cdot|z + \Delta z) \right)}{\|\Delta z\|_2} \\ &\leq \sup_{z \in \mathcal{Z}} \sup_{\Delta z} \frac{\|g\|_{\mathrm{W}^{1,\infty}} \|\Delta z\|_2 L_p}{\|\Delta z\|_2} = \|g\|_{\mathrm{W}^{1,\infty}} L_p.\end{aligned}$$

The second passage relies on the definition of Wasserstein distance and the third one on the fact that $\|g\|_{\mathrm{W}^{1,\infty}}$ upper bounds the Lipschitz constant. This proves that

$$\|\mathcal{T}_h\|^{\circ}_{\mathrm{W}^{1,\infty} \to \mathrm{W}^{1,\infty}} \leq \sup_{g \in L^\infty(\mathcal{S})} \frac{L_r + \max\{1, L_p\} \|g\|_{\mathrm{W}^{1,\infty}}}{1 + \|g\|_{\mathrm{W}^{1,\infty}}} = \max\{1, L_r, L_p\}.$$

We can turn this result into another proposition.

**Proposition 30.** *The class $\mathscr{M}_{\mathcal{W}}$ of Lipschitz MDPs with Wasserstein distance is a subset of the Weakly Smooth MDPs with parameter $\nu = 1$ and constant $C_{\mathcal{T}}^{WS} = \max\{1, L_r, L_p\}$.*

Notably, the constant we get is exactly the same in the generalization $\mathscr{M}_{\mathrm{TV}} \to$ Mildly Smooth and in the one $\mathscr{M}_{\mathcal{W}} \to$ Weakly Smooth. This enforces the idea that the assumptions of smoothness that are imposed over the probability transition in classical papers find a natural generalization in our approach, which only cares about the Bellman optimality operator.

In this section, we have proved three results that reduce well-known classes of non-parametric MDPs with continuous spaces to our classes of Strongly, Mildly and Weakly Smooth MDPs respectively. In addition to demonstrating the generality of our assumptions, this discussion shows that all three are noteworthy in their own way. Indeed, the differences in generality between the three classes are capable of separating three well-known and well-studied families in the literature. For a visual representation of the results in this section, the reader is invited to look at fig. 4.1.

### 4.4.2. Does General Function Approximation Work on Smooth MDPs?

General function approximation provides a very flexible model for a large variety of RL environments. As we have seen in section 4.2.5, the regret bounds under this assumption depends on some complexity measures of the hypothesis space $\mathcal{F}$. The two powerful algorithms in this field that we have introduced, SPEDE and GOLF interpret $\mathcal{F}$ in

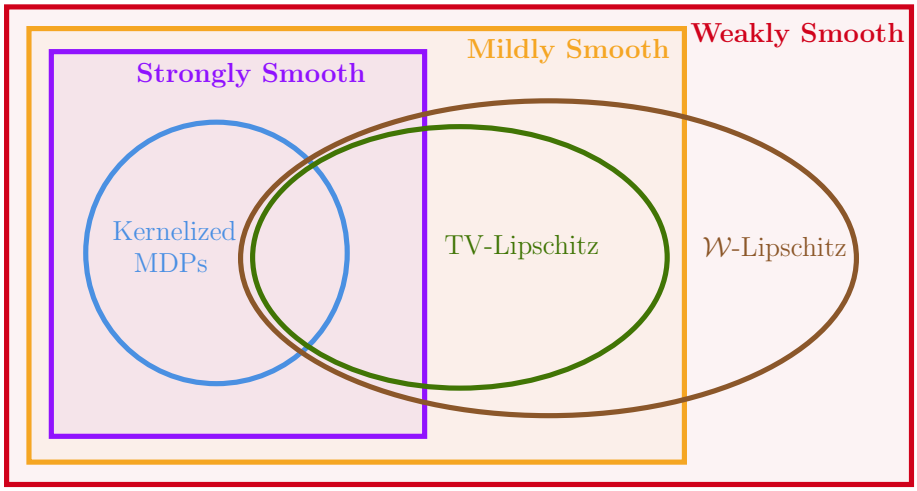



Figure 4.1: Comparison between the non-parametric classes of continuous spaces MDPs described in this section. Strongly, Mildly and Weakly Smooth MDPs follow the hierarchy suggested by their names. Importantly, not all the inclusion follow exactly the same logic. The one between Kernelized MDPs, like the ones between the Smooth family, hold for a fixed and common $\nu$. Thus, a Kernelized MDP with Matén kernel of order $\nu$ is not, in general, Weakly Smooth with parameter $\nu'$ if $\nu' > \nu$. Differently, the inclusion concerning the Lipschitz MDPs work for $\nu = 1$.

different ways. The former, which deals with Gaussian Transition Realizable MDPs adopts a model-based approach, assuming that the transition is given by $s' = f(s, a)$, for $f \in \mathcal{F}$, plus a Gaussian noise. The latter, uses $\mathcal{F}$ to model directly the sequence of $Q^*$ functions.

In both cases, two complexity measures of $\mathcal{F}$ enter in the regret bounds of the corresponding algorithm, the log-covering number and the Eluder dimension (or Bellman-Eluder). All Smooth MDP classes are linked by the function class $\mathrm{W}^{\nu,\infty}(\mathcal{Z})$, which provides the necessary functions to characterize the process. Therefore, it is of paramount interest to understand the value of these complexity measures for $\mathcal{F}(B) := \{f \in \mathrm{W}^{\nu,\infty}(\mathcal{Z}) : \|f\|_{\mathrm{W}^{\nu,\infty}} \leq B\}$. For what concerns the log-covering number, the following result is well-known:

$$\log \mathcal{N}_\infty(\varepsilon, \mathcal{F}(B)) = \Theta\left(B^{m/\nu}\varepsilon^{-m/\nu}\right). \tag{4.19}$$

Unfortunately, an equally simple and clear result about the Eluder dimension does not exist in the literature, as the latter notion of complexity is much more recent. In the next paragraph, we fill this gap.

**What is the Eluder dimension of the Smooth Functions?** For this paragraph, we always let $\mathcal{Z} = [-1, 1]^m$ and, for $B > 0$

$$\mathcal{F}(B) := \{f \in \mathrm{W}^{\nu,\infty}(\mathcal{Z}) : \|f\|_{\mathrm{W}^{\nu,\infty}} \leq B\}.$$

When $B$ is fixed, the former class will just be named $\mathcal{F}$, assuming the parameter is clear from the context. The Eluder dimension $\dim_{\mathrm{E}}(\varepsilon, \mathcal{F})$ (definition 16) of this class is the the maximum length of a sequence of points $x_1, \dots x_d \subset \mathcal{X}$ such that for any $d_0 < d$ there exist $\varepsilon_0 \geq \varepsilon$ and $f, g \in \mathcal{F}$ such that

$$\sqrt{\sum_{i=1}^{d_0}(f(x_i) - g(x_i))^2} \leq \varepsilon_0 \qquad f(x_{d_0+1}) - g(x_{d_0+1}) \geq \varepsilon_0.$$

We are interested in the dependence of this dimension on $\varepsilon$ and $B$.

**Theorem 31.** *The Eluder dimension of the function class $\mathcal{F}$ is bounded as follows, in terms of $\varepsilon$ and $B$.*

$$dim_E(\mathcal{F}, \varepsilon) \leq \widetilde{\mathcal{O}}(B^{m/\nu}\varepsilon^{-m/\nu}).$$

Due to its complexity, the complete proof of this result (see [79], Theorem 15) is omitted. Instead, the following discussion provides the intuition behind the result. The idea exploits a result that dates back to the first paper that introduced the Eluder dimension. Example 4 from [107] shows that, for vector space of dimension $d$, the Eluder dimension grows as $\widetilde{\mathcal{O}}(d \log(1/\varepsilon))$. This result does not directly apply on $\mathcal{F}$, that comes form an infinite dimensional vector space, but a sharper argument can still reduce the problem to the linear case.

Taken any function $f \in \mathcal{F}$, its Taylor polynomial (see appendix B.2) in a fixed point $y$ writes as

$$T_y^{\nu_*}[f](z) = \sum_{\|\alpha\|_1 \leq \nu_*} \frac{D^\alpha f(y)}{\alpha!}(z - y)^\alpha =: \boldsymbol{w}(f)^\top \psi_y^{\nu_*}(z).$$

Here, the feature map $\psi_y^{\nu_*}(x)$ does not depend on $f$; the dependence is contained in the coefficients $\boldsymbol{w}(f)$. The set of possible Taylor polynomials centered in a given point thus forms a vector space given by of finite dimension $\binom{\nu_*+m}{m}$. The properties of this polynomials ensure (theorem 51) that for $f \in \mathcal{F}$

$$\forall z \in \mathcal{Z},\ |f(z) - T_y^{\nu_*}[f](z)| \leq B\|z - y\|_\infty^\nu.$$

The domain $\mathcal{Z} = [-1, 1]^m$ can be partitioned as follows. Fixed $\varepsilon_{\text{cover}} > 0$, we build a

$\varepsilon_{\text{cover}}-$net $\mathcal{Z}^{\varepsilon_{\text{cover}}}$ in the infinity norm, whose size can be proved to be $|\mathcal{Z}^{\varepsilon_{\text{cover}}}| =: N \leq (2/\varepsilon_{\text{cover}})^m$. Now, for every $z^n \in \mathcal{Z}^{\varepsilon_{\text{cover}}}$, we define recursively $\mathcal{Z}_n$ to be the set of points which are near to $z^n$, formally:

$$\mathcal{Z}_1 := \{z \in \mathcal{Z} : \|z - z^1\|_\infty \leq \varepsilon_{\text{cover}}\}, \qquad \mathcal{Z}_n := \{z \in \mathcal{Z} : \|z - z^n\|_\infty \leq \varepsilon_{\text{cover}}\} \setminus \cup_{\ell=1}^{n-1} \mathcal{Z}_\ell. \tag{4.20}$$

By definition, every point of $\mathcal{Z}$ is matched with a point $z^n$ of the cover $\mathcal{Z}^{\varepsilon_{\text{cover}}}$ and, importantly, is assigned to exactly one subset $\mathcal{Z}_n$ of $\mathcal{Z}$. This way, $\{\mathcal{Z}_n\}_{n=1}^N$ defines a partition of $\mathcal{Z}$ where each set has diameter not larger than $\varepsilon_{\text{cover}}$.

The set-up is completed to proceed with our argument. We start bounding the length of the longest independent sequence when intersected with $\mathcal{Z}_n$ for some $n \in [N]$. Define the following space of functions

$$\mathcal{F}_n := \left\{ f : \mathcal{Z}_n \to \mathbb{R} : \ f(z) = w_n^\top \psi_{z^n}^{\nu_*}(z) \right\}.$$

These functions arise from linear combination of a basis of dimension $\text{len}(\psi_{z^n}^{\nu_*}(z)) := d$. Therefore, $\dim_{\text{E}}(\mathcal{F}_n, \varepsilon) \leq \widetilde{\mathcal{O}}(d \log(1/\varepsilon))$. By definition of the cover and from the approximation property of Taylor polynomials, it follows that

$$\forall f \in \mathcal{F}, \exists f_0 \in \mathcal{F}_n : \quad \sup_{z \in \mathcal{Z}_n} |f(z) - f_0(z)| \leq B\varepsilon_{\text{cover}}^\nu.$$

Fix a sequence $\{\tilde{z}_i\}_{i=1}^{d_0+1}$ of length $d_0+1$ and let $B\varepsilon_{\text{cover}}^\nu \ll \varepsilon/\sqrt{d}$. The dependent sequences in $\mathcal{Z}_n$ w.r.t. $\mathcal{F}$ are the same that are dependent w.r.t. $\mathcal{F}_n$. Indeed, for any sequence $\{\tilde{z}_i\}_{i=1}^{d_0} \subset \mathcal{Z}_n$ and any $f, g \in \mathcal{F}$

$$\begin{aligned}
\sqrt{\sum_{i=1}^{d_0} (f(\tilde{z}_i) - g(\tilde{z}_i))^2} &\geq \sqrt{\sum_{i=1}^{d_0} (f_0(\tilde{z}_i) - g_0(\tilde{z}_i))^2} - 2\sqrt{\sum_{i=1}^{d_0} B^2 \varepsilon_{\text{cover}}^{2\nu}} \\
&\geq \sqrt{\sum_{i=1}^{d_0} (f_0(\tilde{z}_i) - g_0(\tilde{z}_i))^2} - 2\sqrt{d_0} B\varepsilon_{\text{cover}}^\nu \\
&= \sqrt{\sum_{i=1}^{d_0} (f_0(\tilde{z}_i) - g_0(\tilde{z}_i))^2} - o(\varepsilon).
\end{aligned}$$

At the same time, if $d_0$ exceeds $\widetilde{\mathcal{O}}(d \log(1/\varepsilon))$, the bound $\dim_{\text{E}}(\mathcal{F}_n, \varepsilon) \leq \widetilde{\mathcal{O}}(d \log(1/\varepsilon))$ on the Eluder dimension of linear function classes implies that $|f_0(\tilde{z}_{d_0+1}) - g_0(\tilde{z}_{d_0+1})| < \varepsilon$.

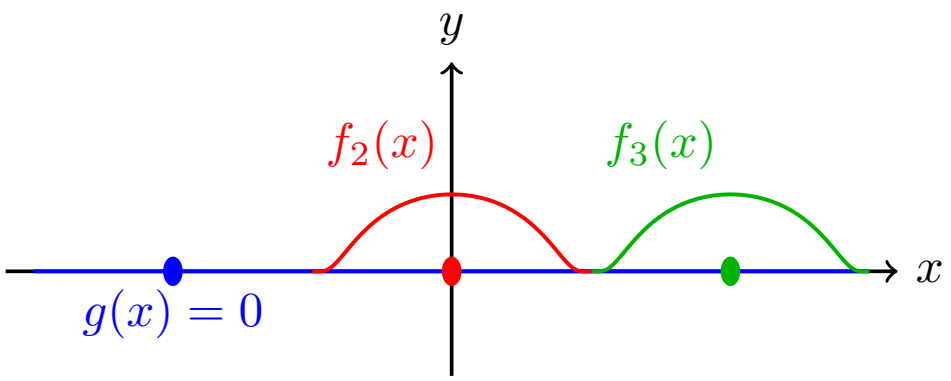



Figure 4.2: A simplified and one-dimensional plot for the construction of the proof of 32. The line is divided in intervals of constant size and, for each of them from $n = 2$ to $N$ we build a function that is compactly supported only on the corresponding interval.

Replacing $f_0$, $g_0$ with $f, g$, that are at most $B\varepsilon_{\text{cover}}^{\nu}-$away, gives

$$|f(\tilde{z}_{d_0+1}) - g(\tilde{z}_{d_0+1})| < \varepsilon + B\varepsilon_{\text{cover}}^{\nu} = \varepsilon + o(\varepsilon).$$

In summary the previous passages show that under $\sqrt{d_0}B\varepsilon_{\text{cover}}^{\nu} \ll \varepsilon$ and $\sqrt{\sum_{i=1}^{d_0}(f(\tilde{z}_i) - g(\tilde{z}_i))^2} \leq \varepsilon$ one has $f(x_{d_0+1}) - g(x_{d_0+1}) < \varepsilon$, where $d_0 \approx d\log(1/\varepsilon)$. In other words, to guarantee that the the length of any $\varepsilon-$independent sequence is bounded by $d_0$, one has to choose

$$\varepsilon_{\text{cover}} \ll \left(\frac{B\sqrt{d\log(1/\varepsilon)}}{\varepsilon}\right)^{-1/\nu}.$$

As the cover size scales as $N = \varepsilon^{-d}$, this fixes the number of regions to $N = \left\lceil \frac{B\sqrt{d\log(1/\varepsilon)}}{\varepsilon} \right\rceil^{m/\nu}$. Taking into account all the terms, the maximum length of an independent sequence does not exceed

$$N \leq \widetilde{\mathcal{O}}(d\log(1/\varepsilon)) \left\lceil \frac{B\sqrt{d\log(1/\varepsilon)}}{\varepsilon} \right\rceil^{m/\nu} = \widetilde{\mathcal{O}}(B^{m/\nu}\varepsilon^{-m/\nu}).$$

By definition of Eluder dimension, we have proved that

$$\dim_{\text{E}}(\mathcal{F}, \varepsilon) \leq \widetilde{\mathcal{O}}(B^{m/\nu}\varepsilon^{-m/\nu}).$$

This result shows that the Eluder dimension for the class $\mathcal{F}$ of functions in $\text{W}^{\nu,\infty}$ with bounded norm is, at most, of the same magnitude as the covering number, up to constants

and logarithmic terms. The proof is not completely rigorous due to the use of the notation $o(\cdot)$ which, just conveys the idea. To make the reasoning formally correct, it would be necessary to introduce the concept of $(\varepsilon_{\text{left}}, \varepsilon_{\text{right}})$-Eluder dimension, for which we refer to [79], Lemma 2. This definition is used only to formalize the ideas we have presented here, and does not carry any real additional interpretation beyond the original Eluder dimension. Much more interesting, however, is to see whether the result we just found is optimal. Is it true that the upper bound obtained corresponds to the true Eluder dimension of the class $\mathcal{F}$, for each $\nu, m$? The next result provides a clear answer.

**Theorem 32.** *The Eluder dimension of the function class $\mathcal{F}$ is lower-bounded as follows, in terms of $\varepsilon$ and $B$.*

$$dim_E(\mathcal{F}, \varepsilon) \geq \Omega(B^{m/\nu}\varepsilon^{-m/\nu}).$$

This result is demonstrated by employing a construction introduced in Appendix F.1 of [75][4], of which we try to give a visual representation in fig. 4.2. Define *standard mollifier* [39] the function $[-1,1]^m \to \mathbb{R}$

$$\Psi(x) := \begin{cases} e^{\frac{1}{1-\|x\|_2^2}} & \|x\|_2^2 < 1 \\ 0 & \|x\|_2^2 \geq 1. \end{cases} \tag{4.21}$$

This function is well-known [39] for being infinitely times differentiable with compact support $B_1(0)$. At this point, define a squeezed and translated version of $\Psi$,

$$\rho < 1, x_0 \in [-1,1]^m : \qquad \Psi_\rho(x|x_0) := \Psi\left(\frac{x - x_0}{\rho}\right). \tag{4.22}$$

This new function is defined on $[x_0 - \rho, x_0 + \rho]$. By evaluating the function in $x_0$, it follows that $\|\Psi_\rho\|_{L^\infty} = 1$. By lemma 4 from [75], it also follows that for any $\rho > 0$ $\|\Psi_\rho\|_{\mathrm{W}^{\nu,\infty}} \leq \|\Psi\|_{\mathrm{W}^{\nu,\infty}}\rho^{-\nu}$.

This fact is the key to find a long sequence of $\varepsilon-$independent points for the class $\mathcal{F}$. Fix $\varepsilon > 0$ and let $\{z_n\}_{n=1}^N$ be an $\varepsilon_{\text{packing}}-$packing, i.e. a set of points that have pairwise distance no lower than $\varepsilon_{\text{packing}}$ (for $\varepsilon_{\text{packing}}$ to be fined later), of $[-1,1]^m$ in infinity norm. It is well known that we can find a packing of this type such that

$$N = \left\lfloor \frac{1}{\varepsilon_{\text{packing}}} \right\rfloor^m.$$

[4]This strategy is not entirely novel; lower bounds in nonparametric statistics typically follow similar principles [131]. We use this one in particular because it is the one that better fits our notation.

Let $\mathcal{Z}_n := \{z : \|z - z_n\|_\infty \leq \varepsilon_{\text{packing}}/2\}$. By definition, these sets are mutually disjoint. We can prove that $\{z_n\}_{n=1}^N$ form a sequence of independent points by using the following functions, for $n \geq 2$:

$$f_n(z) = \begin{cases} \varepsilon\Psi_{\varepsilon_{\text{packing}}}(z|z_n) & z \in \mathcal{Z}_n \\ 0 & z \notin \mathcal{Z}_n \end{cases} \qquad g(z) = 0.$$

These functions show that the sequence of points is independent. On one side, fixing $n \leq N-1$ we have

$$\sum_{i=1}^{n}(f_{n+1}(z_i) - g(z_i))^2 = 0 < \varepsilon^2.$$

On the other,

$$f_{n+1}(z_{n+1}) - g(z_{n+1}) = \varepsilon\Psi_{\varepsilon_{\text{packing}}}(z_{n+1}|z_{n+1}) = \varepsilon.$$

So, for the condition to be valid, we only need $f_n, g \in \mathcal{F}$ for any $2 \leq n \leq N$. While the condition on $g$ is trivial, the one on the $f_n-$s is satisfied for $\varepsilon_{\text{packing}}^{-\nu} = \frac{B}{\varepsilon\|\Psi\|_{\mathrm{W}^{\nu,\infty}}}$. Indeed,

$$\begin{aligned} \|f_n\|_{\mathrm{W}^{\nu,\infty}} &= \|\varepsilon\Psi_{\varepsilon_{\text{packing}}}(\cdot|z_n)\|_{\mathrm{W}^{\nu,\infty}} \\ &= \varepsilon\|\Psi_{\varepsilon_{\text{packing}}}(\cdot|z_n)\|_{\mathrm{W}^{\nu,\infty}} \leq \varepsilon\|\Psi\|_{\mathrm{W}^{\nu,\infty}}\varepsilon_{\text{packing}}^{-\nu} = B. \end{aligned}$$

Note that $\|\Psi\|_{\mathrm{W}^{\nu,\infty}}$ is an absolute constant that does not depend on $\varepsilon, B$. The total length of the sequence is then given by

$$N = \left\lfloor \frac{1}{\varepsilon_{\text{packing}}} \right\rfloor^m = \left\lfloor \left( \frac{B}{\varepsilon\|\Psi\|_{\mathrm{W}^{\nu,\infty}}} \right)^{1/\nu} \right\rfloor^m = \Omega\left(B^{m/\nu}\varepsilon^{-m/\nu}\right).$$

Having found an $\varepsilon-$independent sequence of length $N$ proves the lower bound

$$\dim_{\mathrm{E}}(\mathcal{F}, \varepsilon) \geq \Omega(B^{m/\nu}\varepsilon^{-m/\nu}).$$

Our results, theorem 32 and theorem 31, match up to logarithmic terms. Interestingly, both coincide with the log-covering number: for $\mathrm{W}^{\nu,\infty}(\mathcal{Z})$ the two terms are of the same magnitude. This phenomenon also happens for subsets of finite norm in a $d$-dimensional vector space, where both take the value of $d$, but is not true in general. It is worth mentioning that our results on the Eluder dimension solve an open problem related to the performance of Thompson Sampling (the classical algorithm going back to [125]) on continuous spaces. Since the topic is orthogonal to our discussion on Smooth MDPs, we relegate it to the appendix C.

**Regret bounds for GFA algorithms on Smooth MDPs** The former bounds on the Eluder dimension allow to evaluate the regret bounds of general function approximation algorithms for classes of smooth functions. Let us start from SPEDE, the method introduced in [105]. This algorithm works under assumption 17 of Gaussian Transition-Realizable MDP, namely that $\mathcal{S} = \mathbb{R}^{m_1}, \mathcal{A} = \mathbb{R}^{m_2}$, with $m = m_1 + m_2$ and for any $h \in [H]$ the transition takes the form

$$s' = f^\star(s, a) + \eta \qquad \eta \sim \mathcal{N}(0, \sigma^2 I),$$

for some $f^\star \in \mathcal{F}$, an hypothesis class that is known to the agent. As already established, when the noise is Gaussian, in order to have a transition function $p_h$ that is smooth of order $\nu$ one needs each of the $m_1$ components of $f^\star$ to be in $\mathrm{W}^{\nu,\infty}(\mathcal{Z})$. This entails that the agent has to choose $\mathcal{F} = \{f \in \mathrm{W}^{\nu,\infty}(\mathcal{Z}) : \|f\|_{\mathrm{W}^{\nu,\infty}} \leq B\}$, the function class studied in the previous paragraph. Under this choice, the regret bound of the algorithm writes as

$$\mathrm{RegretBound}(\mathrm{SPEDE}|\mathscr{M}_{\mathrm{GTR}}) \leq \widetilde{\mathcal{O}}\left(H^{3/2}\sqrt{\dim_{\mathrm{E}}((HK)^{-1/2}, \mathcal{F}) \log \mathcal{N}_\infty((HK)^{-1/2}, \mathcal{F})K}\right).$$

Replacing $\dim_{\mathrm{E}}$ and $\log \mathcal{N}_\infty$ with the bounds we provided for $\mathcal{F}$ gives

$$\mathrm{RegretBound}(\mathrm{SPEDE}|\mathscr{M}_{\mathrm{GTR}}) \leq \widetilde{\mathcal{O}}\left(H^{3/2}(HK)^{\frac{m}{2\nu}}\sqrt{K}\right) = \widetilde{\mathcal{O}}\left(H^{\frac{m+3\nu}{2\nu}}K^{\frac{m+\nu}{2\nu}}\right). \quad (4.23)$$

This regret bound does not apply to every Strongly Smooth MDP with parameter $\nu$, as the noise on the next state is assumed to come from a multivariate Gaussian. Nonetheless, the GTR class with a smooth transition gives an idea of how a regret bound for Smooth MDPs may look like. Crucially, all the terms explode for $m \gg \nu$. In this regime, where the dimensionality of the problem is high compared to the degree of smoothness, the regret bound becomes vacuous. Indeed, for $\nu \leq m$ the exponent of $K$ is $\geq 1$. This means that the algorithms does not enjoy the no-regret property, and convergence to the performance of the optimal policy is not granted. On the other hand, in the opposite case, when the degree of smoothness prevails over the dimensionality, the regret is bounded nicely. In particular, for $\nu \gg m$, the regret bound is of order $\widetilde{\mathcal{O}}(\sqrt{H^3K})$, which is what one would obtain by ignoring the Eluder dimension and the log-covering number. This insight will recur in many of the subsequent results: as the degree of smoothness tends to infinity, the bound on regret behaves like that of a very small linear model, where the complexity terms related to the class of functions are logarithms that we can eliminate.

Let us now turn our attention to the most general algorithm introduced so far, namely GOLF [55]. In this case, the family of processes for which the theoretical guarantee

applies has a much more detailed definition. To run the algorithm, the agent needs two hypothesis spaces $\mathcal{F} = \mathcal{F}_1 \times \dots \mathcal{F}_h \cdots \times \mathcal{F}_H$ and $\mathcal{G} = \mathcal{G}_1 \times \dots \mathcal{G}_h \cdots \times \mathcal{G}_H$ of bounded functions $\mathcal{Z} \to \mathbb{R}$. These spaces must satisfy the following assumptions.

1. $\mathcal{T}_h(\mathcal{F}_{h+1}) \subseteq \mathcal{G}_h$
2. for any $\varepsilon > 0$, $\dim_{\mathrm{BE}}(\varepsilon, \mathcal{F}, \mathcal{P}) := \max_{h\in[H]} \dim_{\mathrm{DE}}(\varepsilon, (I-\mathcal{T}_h)\mathcal{F}_h, \mathcal{P}) < +\infty$, for $\mathcal{P}$ that is either the set of Dirac's deltas over $\mathcal{S} \times \mathcal{A}$ or the set of probability distributions over the state action couples that may result when executing a greedy policy with respect to some $f \in \mathcal{F}$.

The regret bound is going to depend on complexity measures related to these hypothesis spaces. Thus, to guarantee a fair comparison, we must choose them as small as possible, while still covering the largest possible notion of Smooth MDPs. Surprisingly, we will show here that for a particular choice of $\mathcal{F}, \mathcal{G}$ these assumption cover the large class Weakly Smooth MDPs. Indeed, let $C := \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}$ to lighten the notation, and

- $\mathcal{F}_h := \{f \in \mathrm{W}^{\nu,\infty}(\mathcal{Z}) : \|f\|_{\mathrm{W}^{\nu,\infty}} \leq \frac{C^{H-h+1}-C}{C-1}\}$,
- $\mathcal{G}_h = \mathcal{F}_h$.

The first condition if satisfied: take any $f_{h+1} \in \mathcal{F}_{h+1}$. Then,

$$\begin{aligned}
\|\mathcal{T}_h f_{h+1}\|_{\mathrm{W}^{\nu,\infty}} &\leq \|\mathcal{T}_h\|^{\circ}_{\mathrm{W}^{\nu,\infty}\to\mathrm{W}^{\nu,\infty}}(1 + \|f_{h+1}\|_{\mathrm{W}^{\nu,\infty}}) \\
&\leq C(1 + \|f_{h+1}\|_{\mathrm{W}^{\nu,\infty}}) \\
&\leq C\left(1 + \frac{C^{H-h} - C}{C-1}\right) \\
&\leq \left(\frac{C^{H-h+1} - C}{C-1}\right) \implies \mathcal{T}_h f_{h+1} \in \mathcal{G}_h.
\end{aligned}$$

Any smaller set would violate the norm bound. Now, we have to upper-bound the Bellman Eluder dimension, which is the maximum over $h$ of the minimum of two terms. The first is $\dim_{\mathrm{DE}}(\varepsilon, (I - \mathcal{T}_h)\mathcal{F}_h, \mathcal{P}_\delta)$, the distributional Eluder dimension over all possible Dirac's delta on $\mathcal{Z}$. The second is $\dim_{\mathrm{DE}}(\varepsilon, (I-\mathcal{T}_h)\mathcal{F}_h, \mathcal{P}_{\mathrm{greedy}})$, the set of probability distributions over the state action couples that may result when executing a greedy policy with respect to some $f \in \mathcal{F}$. As the Weakly Smooth MDPs are invariant for translation of the state-action space (the norm of a smooth function does not change after a translation), every action can be optimal for any state in some Weakly Smooth MDPs. Thus, the second term is no lower than the first one, which corresponds to

$$\max_{h\in[H]} \dim_{\mathrm{DE}}(K^{-1/2}, (I - \mathcal{T}_h)\mathcal{F}_h, \mathcal{P}_\delta) = \max_{h\in[H]} \dim_{\mathrm{E}}(K^{-1/2}, (I - \mathcal{T}_h)(\mathcal{F}_h - \mathcal{F}_h)),$$

as already noted in the discussion after presenting GOLF. For our choice of $\mathcal{F}_h$, the set $(I - \mathcal{T}_h)(\mathcal{F}_h - \mathcal{F}_h))$ corresponds to

$$(I - \mathcal{T}_h)(\mathcal{F}_h - \mathcal{F}_h)) = \left\{ f \in \mathrm{W}^{\nu,\infty}(\mathcal{Z}) : \|f\|_{\mathrm{W}^{\nu,\infty}} \leq 2\frac{(1+C)(C^{H-h+1} - C)}{C-1} \right\},$$

as it follows from applying the triangular inequality over the norm $\|\cdot\|_{\mathrm{W}^{\nu,\infty}}$ plus the definition of $C$. The Bellmann-Eluder dimension thus writes as

$$\dim_{\mathrm{BE}}(\varepsilon, \mathcal{F}_h, \mathcal{P}) = \widetilde{\mathcal{O}}\left( C^{\frac{(H-h+3)m}{\nu}} \varepsilon^{-\frac{m}{\nu}} \right).$$

Ignoring the constants, the maximum over $h$ is of order $C^{\frac{Hm}{\nu}}\varepsilon^{-\frac{m}{\nu}}$. The regret upper bound requires then to estimate the log-covering number of $\mathcal{F}_h$, for which eq. (4.19) shows a similar rate to the Eluder dimension. At this point, the regret bound of GOLF results in the first regret guarantee for Weakly Smooth MDPs.

$$\begin{aligned}
\mathrm{RegretBound}(\mathrm{GOLF}|\mathscr{M}_{\mathrm{WS}}) &\leq \widetilde{\mathcal{O}}\left( H^2 \sqrt{d_{\mathrm{BE}} \log \mathcal{N}_\infty(K^{-1}, \mathcal{F} \cup \mathcal{G}) K} \right) \\
&\leq \widetilde{\mathcal{O}}\left( H^2 \sqrt{C^{\frac{Hm}{\nu}} K^{\frac{m}{2\nu}} C^{\frac{Hm}{\nu}} K^{\frac{m}{\nu}} K} \right) \\
&= \widetilde{\mathcal{O}}\left( C^{\frac{Hm}{\nu}} K^{\frac{3m+2\nu}{4\nu}} \right). \qquad (4.24)
\end{aligned}$$

This regret bound has multiple problems, and not all them are caused by GOLF. First, the dependence in $C = \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}$ is exponential. There is not much that we can do about it: as we have shown (recall fig. 4.1), Weakly Smooth MDPs contain $\mathcal{W}-$Lipschitz MDPs as a particular case. Since the regret is doomed to grow exponential in $H$ for $\mathcal{W}-$Lipschitz MDPs, the same result affects the Weakly Smooth class. Second, and perhaps more important, the regret bound grows linearly unless the dimensionality of the problem is much smaller than its smoothness. In fact, the exponent of $K$ is $\frac{3m+2\nu}{4\nu}$, which is less than one only for $3m < 2\nu$. In the opposite case, which this time may mean even pairs $(m, \nu)$ such that $\nu > m$, the regret bound is vacuous. Unlike the previous one, this problem is specifically inherent to GOLF, since there is no a priori result that says that under these assumptions it is impossible to get the no-regret property.

In fact, when trying to apply GOLF to the smaller class of Mildly Smooth MDPs, the regret scales only polynomially with $H$. Let $C := \mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}$ to lighten the notation, and

- $\mathcal{F}_h := \{f \in \mathrm{W}^{\nu,\infty}(\mathcal{Z}) : \|f\|_{\mathrm{W}^{\nu,\infty}} \leq C(H-h+1), \|f_h\|_{L^\infty} \leq H-h+1\}$,
- $\mathcal{G}_h = \mathcal{F}_h$.

We can prove that this choice satisfies Bellman closure. Take any $f_{h+1} \in \mathcal{F}_{h+1}$. Then, by

$$\begin{aligned}\|\mathcal{T}_h f_{h+1}\|_{\mathrm{W}^{\nu,\infty}} &\leq \|\mathcal{T}_h\|_{L^\infty \to \mathrm{W}^{\nu,\infty}}(1 + \|f_{h+1}\|_{L^\infty}) \\ &\leq \|\mathcal{T}_h\|_{L^\infty \to \mathrm{W}^{\nu,\infty}}(1 + H - (h+1) + 1) \\ &\leq C\,(H - h + 1)\,.\end{aligned}$$

Moreover, by assumption 6 it follows that

$$\begin{aligned}\|\mathcal{T}_h f_{h+1}\|_{L^\infty} &\leq \|r_h\|_{L^\infty} + \left\| \int_{\mathcal{S}} \max_{a \in \mathcal{A}} f_{h+1}(s', a') p_h(s'|\cdot) ds' \right\|_{L^\infty} \\ &\leq 1 + \|f_{h+1}\|_{L^\infty} \leq H - h + 1.\end{aligned}$$

The second inequality comes from the fact that $\max_{a \in \mathcal{A}} f_{h+1}(s', a')$ cannot exceed $\|f_{h+1}\|_{L^\infty}$ and from the fact that $p_h$ is a probability measure. Together, these results ensure that $\mathcal{T}_h f_{h+1} \in \mathcal{G}_h$. The Bellman-Eluder dimension of this smaller function class, whose maximal norm in the Sobolev-Hölder space is just $CH$, can be bounded as before. This provides the following regret bound for GOLF.

$$\begin{aligned}\mathrm{RegretBound}(\mathrm{GOLF}|\mathscr{M}_{\mathrm{MS}}) &\leq \widetilde{\mathcal{O}}\left(H^2 \sqrt{d_{\mathrm{BE}} \log \mathcal{N}_\infty(K^{-1}, \mathcal{F} \cup \mathcal{G}) K}\right) \\ &\leq \widetilde{\mathcal{O}}\left(H^2 \sqrt{(CH)^{\frac{m}{\nu}} K^{\frac{m}{2\nu}} (CH)^{\frac{m}{\nu}} K^{\frac{m}{\nu}} K}\right) \\ &= \widetilde{\mathcal{O}}\left(H^{\frac{2\nu+m}{\nu}} K^{\frac{3m+2\nu}{4\nu}}\right). \qquad (4.25)\end{aligned}$$

As anticipated, the dependence on $H$ is only polynomial (even if the exponent may be very big for $m \gg \nu$). Nonetheless, the regret order is the same of eq. (4.24), with a superlinear dependence in $K$ in some regimes.

## 4.5. Research Questions on Smooth MDPs

In the former section, we saw that, although some RL algorithms with general function approximation apply to classes of Smooth MDPs, their regret guarantees are not necessarily sublinear. The dramatic effect is that this prevents convergence to an $\varepsilon-$optimal policy with arbitrarily small $\varepsilon$. Indeed, to move from regret bound to sample complexity, the proposition 9 requires it to grow at most as $K^\alpha$, with $\alpha < 1$.

Next, we list some open questions that we will attempt to address in the next chapters. These questions seek to establish how far the theoretical guarantees of Smooth MDPs can be pushed.

a In the case of access to a generative model, is it possible to achieve a sample complexity guarantee that is non-vacuous for any $\varepsilon, m, \nu$?

b The optimal sample complexity for identifying a $\varepsilon-$optimal arm in case of a Smooth Bandit problem is $\Theta(\varepsilon^{-2-m/\nu})$. Can we match this bound for the sample complexity of Smooth MDPs with access to a generative model?

c In case of the online interaction, is it possible to achieve a regret bound that is always sub-linear in $K$?

d Can we achieve a regret guarantee of $\Theta(K^{\frac{m+\nu}{m+2\nu}})$, which matches the lower bound for Smooth Bandits?

The former questions are (almost) ranked in terms of increasing difficulty. As the online interaction is more challenging than the generative model setting, and by proposition 9 we can translate any sublinear regret bound into a sample complexity guarantee, a positive answer to $(c)$ implies a positive answer to $(a)$ and one to $(d)$ implies one to $(b)$. Moreover, the sample complexity rate $\widetilde{\mathcal{O}}(\varepsilon^{-2-\nu/m})$ is always finite, so $(b)$ is stronger than $(a)$. At the same time, a regret guarantee of $\widetilde{\mathcal{O}}(K^{\frac{\nu+m}{2\nu+m}})$ is always sub-linear, so $(d)$ is stronger than $(c)$. The only exception in this ranking is given by $(c)$, whose positive answer does not necessarily entails one to $(b)$ (and vice versa). We will start in the next chapter from questions $(a)$ and $(b)$, which are related to access to a generative model.

# 5 | Smooth MDPs: Sample Complexity

In this chapter, we study the problem of learning an $\varepsilon$-optimal policy in Smooth Markov Decision Processes in the presence of a generative model. This setting, which has been introduced rigorously at the beginning of the thesis, allows the learner to query the environment at arbitrary state–action–stage tuples and obtain independent samples of rewards and transitions. While stronger than online interaction, access to a generative model is standard in the theoretical analysis of reinforcement learning, as it makes it possible to disentangle the intrinsic statistical difficulty of the problem from issues related to exploration and temporal dependence.

Our main goal is to show that, under smoothness assumptions, access to a generative model enables rate-optimal sample complexity in $\varepsilon$ for continuous-state and continuous-action MDPs. More precisely, we focus on the most general and challenging setting within the Smooth MDP framework, namely the class of weakly smooth MDPs, where smoothness is imposed directly on the Bellman optimality operator rather than on the reward and transition functions separately. This class strictly contains Lipschitz MDPs and, at the same time, encompasses most parametric and kernel-based models studied in the literature. As such, it represents a natural benchmark for understanding the limits of sample-efficient reinforcement learning in continuous spaces.

To make the analysis tractable and to design an algorithm achieving optimal rates, we introduce a mild technical modification to the definition of Smooth MDPs considered in definition 22. Specifically, we assume periodic boundary conditions on the functions involved in the Bellman updates. This assumption was not present in the original definition of smooth MDPs and is introduced here for methodological reasons, as it allows us to leverage tools from harmonic analysis, such as Fourier representations and projection operators defined via convolution, in a clean and controlled way, avoiding boundary effects that would otherwise obscure the main ideas.

Crucially, this additional assumption does not alter the nature of the learning problem

in a substantial way. The periodicity condition only concerns the behavior of functions at the boundary of the state–action domain and does not impose any extra regularity in its interior. As we will discuss throughout the chapter and show explicitly at the end, the intrinsic difficulty of the problem remains essentially unchanged: the resulting sample complexity bounds match known lower bounds in simpler settings and exhibit the same dependence on the accuracy parameter $\varepsilon$, the dimension of the state–action space, and the smoothness order $\nu$ as in the non-periodic formulation. In this sense, the periodicity assumption should be viewed purely as a technical device that simplifies the analysis without strengthening the underlying model. The results of this chapter are presented in [80]. Some of the proofs, which we present here in a more elegant version, are based on results from [75].

## 5.1. Fourier Representation

The algorithm we present in this chapter is based on the idea of approximating the state-action value function at any time-step $h$ with a trigonometric polynomial. We use the notation from appendix B.3. In this chapter we will always write:

- $\mathcal{S} = [-1, 1]^{m_1}, \mathcal{A} = [-1, 1]^{m_2}$
- $\mathcal{Z} = \mathcal{S} \times \mathcal{A} = [-1, 1]^m$
- $\mathrm{T}_N(\mathcal{Z})$ (or just $\mathrm{T}_N$) the vector space of trigonometric polynomials on $\mathcal{Z}$ whose degree does not exceed $N$
- $\varphi_N : \mathcal{Z} \to \mathbb{R}^d$ the Fourier feature map of degree $N$. Here, as follows from (B.10), the length of the feature map is $d := \binom{2N+m}{m} \leq (2N+1)^m$.
- $f * g =$ circular convolution, i.e. convolution where $g$ is extended periodically when its argument lies outside of the domain

At a high level, our algorithm will be based on the well-known Least Square Value Iteration (LSVI) algorithm. In a nutshell, LSVI repeats the following routine for all $h = H \dots 1$. To estimate $Q^*_{h+1}(s, a)$ as $\widehat{Q}_{h+1}(s, a)$. To generate transition and reward data relative to step $h$, i.e. quadruples $s_h, a_h, s_{h+1}, y_h$. To fit the next step $Q^*_h(s, a)$ using as target $\widehat{\mathcal{T}}_h \widehat{Q}_{h+1}(s, a)$, where $\widehat{\mathcal{T}}_h$ is an approximation of the operator that comes from the samples. In order to let this strategy work, it is necessary to choose a vector space of approximation function for $\widehat{Q}_{h+1}(s, a)$, which must be large enough to have good approximation properties, but small enough so that the agent does not overfit to the noisy data. The choice of this vector space depends on the assumptions we have about the target function. As we have

seen, the smoothness of the function $Q^*_{h+1}$ is guaranteed by all the assumptions of Smooth MDPs. However, using LSVI, the target will not be calculated from the exact $Q-$function at the next step, but from its approximation $\widehat{Q}_{h+1}$, which is not necessarily smooth.

Therefore, the function approximator must be capable of preserving smoothness during approximation. Taylor features (B.4), which are perhaps the most natural way to approximate a smooth function, do not work here, as their local nature leads to approximations that suffer from discontinuities at the boundaries of each interval. In contrast, Fourier features are infinitely differentiable and enjoy a similar approximation property as smooth functions, as the next theorem shows.

**Theorem 33.** *[Theorem 4, part (ii) in [113]] There is a constant $J_T > 0$, only depending on $m$, such that for any $f \in W_p^{\nu,\infty}(\mathcal{Z})$*

$$\min_{g \in T_N} \|f - g\|_{L^\infty} \leq \frac{J_T}{N^\nu} \|f\|_{W^{\nu,\infty}}.$$

The previous theorem states that the vector space $\mathrm{T}_N$, consisting of trigonometric polynomials of degree no greater than $N$, can approximate any function in $\mathrm{W}_{\mathrm{p}}^{\nu,\infty}(\mathcal{Z})$ at a rate of order $N^{-\nu}$. $\mathrm{W}_{\mathrm{p}}^{\nu,\infty}(\mathcal{Z})$ denotes the space $\mathrm{W}^{\nu,\infty}$ that we introduced in the previous chapter, with the only caveat of imposing periodic boundary conditions (for details, see appendix B.3). This condition is necessary to be able to use trigonometric polynomials, which by their nature can only produce periodic functions. We thus need to consider a slight modification of the smooth MDP definition. In this chapter, as in the paper [80], we will work with these variations of the original assumptions.

**Definition 23** (Periodic *Mildly* Smooth MDPs)**.** *The class $\mathscr{M}_{p\text{-}WS}$ of Periodic Mildly Smooth MDPs is the set Mildly Smooth MDPs (definition 21) where the condition on the operator norm is replaced with*

$$\|\mathcal{T}_h\|^\circ_{L^\infty \to W_p^{\nu,\infty}} \leq C_{\mathcal{T}}^{MS}. \tag{5.1}$$

**Definition 24** (Periodic *Weakly* Smooth MDPs)**.** *The class $\mathscr{M}_{p\text{-}MS}$ of Periodic Weakly Smooth MDPs is the set of Weakly Smooth MDPs (definition 22) where the condition on the operator norm is replaced with*

$$\|\mathcal{T}_h\|^\circ_{W_p^{\nu,\infty} \to W_p^{\nu,\infty}} \leq C_{\mathcal{T}}^{WS}. \tag{5.2}$$

Algorithm 5.1 Least-Squares Value Iteration (LSVI)

**Require:** Horizon $H$, feature map $\varphi : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^d$, datasets $\{\mathcal{D}_h\}_{h=1}^H = \{\{(s_t, a_t)\}_{t=1}^{n_h}\}_{h=1}^H$
**Ensure:** Policy $\widehat{\pi}$

Initialize $\widehat{V}_{H+1}(s) \leftarrow 0$ for all $s \in \mathcal{S}$
**for** $h \leftarrow H, \dots, 1$ **do**
  **for** $z_h^t = (s_h^t, a_h^t) \in \mathcal{D}_h$ **do**
    Query simulator $s_{h+1}^t \sim p_h(\cdot|z_h^t)$ and $y_h$ of mean $r_h(s_h^t, a_h^t)$
    $q_h^t \leftarrow y_h + \widehat{V}_{h+1}(s_{h+1}^t)$
  **end for**
  Solve least squares

$$\widehat{\theta}_h = \arg\min_{\theta\in\mathbb{R}^{\tilde{N}}} \sum_{t=1}^{n_h} \left(\varphi_N(s_h^t, a_h^t)^\top \theta - q_h^t\right)^2$$

  $\widehat{Q}_h(\cdot,\cdot) \leftarrow \varphi_N(\cdot,\cdot)^\top \widehat{\theta}_h$
  $\widehat{V}_h(\cdot) \leftarrow \max_{a\in\mathcal{A}} \widehat{Q}_h(\cdot, a)$
**end for**
**Return** $\{\widehat{\pi}_h(\cdot)\}_{h=1}^H \leftarrow \{\arg\max_{a\in\mathcal{A}} \widehat{Q}_h(\cdot, a)\}_{h=1}^H$

These new assumptions retain the exact same meaning as their original counterparts. The difference does not lie in the complexity of the problem, since in this chapter we will demonstrate that the same lower bounds hold under the previous assumptions, but in the intuition related to the edges of the state space. Intuitively speaking, a periodic Smooth MDP is simply a smooth MDP where the so-called Pac-Man effect occurs at the edges of the state space: exiting one edge causes the agent to teleport to the other. Another way to look at this assumption is to interpret the state space $\mathcal{S} = [-1, 1]^{m_1}$ as a parameterization of an MDP where the true state moves on a spherical surface, so that no edges exist and equally smoothness is required at every point. A discussion on the generality of these periodic boundary conditions will be delayed to section 5.4.2.

### 5.1.1. Issue with Standard LSVI

Now that theorem 33 guarantees the possibility of approximating smooth functions, including the $Q-$function with trigonometric polynomials of the form $\varphi_N(z)$, the most natural thing to do would be to apply the LSVI algorithm [92], shown in algorithm 5.1, to compute the function $\widehat{Q}_h$ at each time stage. This time, we assume we have already decided the datasets $\mathcal{D}_h$ of state-action pairs to query the simulator at each stage.

The algorithm proceeds backwards in time from the terminal step $H$ to step 1. At each stage $h$, it constructs targets of the form $r_h(s, a) + \widehat{V}_{h+1}(s')$, where $\widehat{V}_{h+1}$ is the state-

value function which was computed at the previous stage. Using a dataset of transitions collected at stage $h$, LSVI estimates $\widehat{\theta}_h$ by solving a least-squares regression problem that fits these targets in the feature space induced by $\varphi_h$.

LSVI works very well when the true $Q_h^*$ function lies exactly in the span of the feature map $\varphi$, as Theorem 3.3 from [4] establishes. However, in our scenario, this assumption is not strictly satisfied. Indeed, we have shown that i) in a Smooth MDP the $Q_h^*$ function is smooth and ii) a trigonometric polynomial can *approximate* every smooth function up to an error that decays with its degree. As the vector space of smooth functions is infinite-dimensional, the only way to have zero approximation error would be to take a degree $N = +\infty$, which is impossible.

Due to the presence of this approximation term, the feature map is said to suffer from a *misspecification*. Similar scenarios have been previously studied in the literature: recall theorem 20 and the following discussion. As anticipated, this misspecification significantly impacts the type of guarantees we are able to achieve. From this point on, let us define $\xi(f;\varphi)$ the misspecification that is incurred when trying to project $f$ over the span of $\varphi$. In other mathematical terms,

$$\xi(f;\varphi) = \inf_{\theta\in\mathbb{R}^d} \|f(\cdot) - \varphi(\cdot)^\top\theta\|_{L^\infty}.$$

Even in the simpler bandit setting (when $H = 1$) it is well known [69] that learning a $\varepsilon-$optimal policy, even with an arbitrary number of samples, is only guaranteed when $\varepsilon > \sqrt{d}\xi(f;\varphi)$. The fact that the misspecification gets multiplied by $\sqrt{d}$ in the sample complexity turns out not to be improvable in the general case [69]. Still, this has catastrophic effects on our setting.

Recall that the length $d$ of the $m-$variate Fourier feature $\varphi_N$ map is, roughly $d \approx (2N+1)^m$. At the same time, being the $Q_h^* \in \mathrm{W}_{\mathrm{p}}^{\nu,\infty}(\mathcal{Z})$, theorem 33 ensures that, in worst case $\xi(Q_h^*;\varphi_N) = \Theta(N^{-\nu})$. in order to decrease $\xi(Q_h^*;\varphi_N)$ we have to increase $N$, but this leads to an increase in $d$, a trade-off that may be unsolvable. In fact

$$m > 2\nu \implies \sqrt{d}\xi(Q_h^*;\varphi_N) = \Theta(N^{d/2-\nu}) \geq \Omega(1).$$

In this case, in the product $\sqrt{d}\xi(Q_h^*;\varphi_N)$ cannot become arbitrarily small, so that learning an $\varepsilon-$optimal policy using proof strategies that deal with misspecification for general feature maps is impossible. For this reason, [80] introduced a new technique called *Projection by convolution*, which gives name to the paper.

### 5.1.2. Projection by Convolution

The issue with the standard analysis of LSVI, motivates us to change our perspective. Instead of trying to learn $Q_h^*$, we focus on a projected version $\widetilde{Q}_h$ on the space of trigonometric polynomials.

We call *generalized projection operator* a function $\mathrm{P}_N : \mathrm{W}_{\mathrm{p}}^{\nu,\infty} \to \mathrm{T}_N$ that is linear and corresponds to the identity when restricted to $\mathrm{T}_M$, for some $M \leq N$. For every stage $h = 1, \dots H$, we define

$$\widetilde{Q}_h(s,a) = \mathrm{P}_N[\mathcal{T}_h \widetilde{Q}_{h+1}](s,a),$$

with $\widetilde{Q}_{H+1} = Q_{H+1}^* = 0$. Using this approach, the function $\widetilde{Q}_{h+1}$ is always well-specified. Indeed, starting from $\widetilde{Q}_{h+1}(s,a)$, that is a trigonometric polynomial (and therefore smooth), $\mathcal{T}_h \widetilde{Q}_{h+1}$ keeps being smooth and $\widetilde{Q}_h = \mathrm{P}_N[\mathcal{T}_h \widetilde{Q}_{h+1}]$ is a trigonometric polynomial. Hopefully, as $\xi(\widetilde{Q}_h; \varphi_N) = 0$, the classical results for the LSVI may help us in approximating the former function. Two problems now need to be addressed

1. Due to the nature of the problem, we only have access to samples of the form $s_h, a_h, s_{h+1} \sim p_h(\cdot|s_h, a_h)$ from the generative model. These samples are good for $\mathcal{T}_h$, not necessarily for $\mathrm{P}_N \cdot \mathcal{T}_h$. How can we approximate the projection operator $\mathrm{P}_N$ in this setting?

2. If the previous step succeeds, the algorithm is going to output $\widehat{Q}_h$, an unbiased estimator for $\widetilde{Q}_h$. For fixed $N$, this function is different from the optimal $Q-$function. How can we make $\widetilde{Q}_h$ converge to $Q_h^*$ for the number of data that grows to infinity?

Fortunately, both open points can be solved in a satisfactory way due to the particular properties of our Fourier feature map.

#### The Power of Convolution

In order to explain our approach, we start from a simplified learning problem. Let us suppose to have a bandit access to a black-box function $f : \mathcal{Z} \to \mathbb{R}$, so that it is possible to pick values $z_t \in \mathcal{Z}$ and receive an unbiased sample $z_t = f(z_t) + \eta_t$, where $\eta_t$ is a zero-mean noise independent from past samples. One simple trick shows that we can use randomization to obtain samples from functions that are different from $f$, still using the same generative model. Indeed, let $\omega$ be a noise variable with probability density function $\mathscr{P}$. If, after choosing $z_t$, we perturb it with $\omega_t$ (sampled from density function $\mathscr{P}$, independently of the past), the resulting sample has a mean satisfying

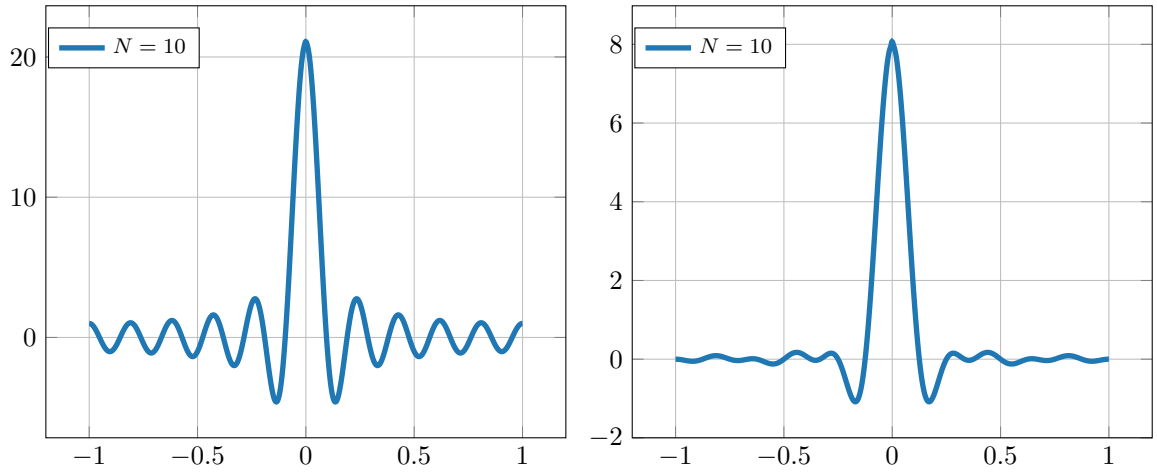



Figure 5.1: A comparison between the Dirichlet kernel $\mathscr{D}_N$ (left) and de la Vallée-Poussin $\mathscr{V}_N$ one (right) in one dimension. The second kernel is characterized by oscillations that are much smaller, allowing to bound its $L^1-$norm with a constant $\Lambda$ independent on $N$.

$$\mathbb{E}[y_t] = \mathbb{E}[f(z_t + \omega_t)] = [f * \mathscr{P}](z_t).$$

This process allows us to obtain unbiased samples from any function $\widetilde{f}$, as long as it can be expressed as a convolution between $f$ and another function $\mathscr{P}$. Thus, coming back to our problem, let $f \in \mathrm{W}_{\mathrm{p}}^{\nu,\infty}$. Can we leverage this method to obtain unbiased samples from $\mathrm{P}_N[f]$, for any choice of the function $f$, always using the same $\mathscr{P}$? The answer depends on the choice of the projection operator $\mathrm{P}_N$. When interested in the usual Fourier series projection, which minimizes the $L^2-$distance from the original function, Fourier analysis ensures that

$$\mathrm{P}_N[f](z) = \mathscr{D}_N * f(z),$$

where $\mathscr{D}_N$ is the $N-$th degree Dirichlet Kernel (see appendix B.3 for the explicit for of this function for $m = 1$). The main property of this kernel lies in the fact that, as the previous equation tells, for *any* function $f$, the convolution product $f * \mathscr{D}_N$ corresponds exactly to the truncation to the order $N$ of the Fourier series associated to $f$.

Thus, when projecting with respect to the $L^2$ norm, things go very smoothly. But, we are interested in a trigonometric polynomial that approximates $f$ uniformly: we need a different kernel corresponding to a projection that is more suited for our case. This can be done by a slight modification of the Dirichlet kernel, a function which takes the name of de la Vallée Poussin kernel [32, 33, 94], that we denote as $\mathscr{V}_N$ (see appendix B.3). The main feature that makes this kernel superior, for our purposes, to the Dirichlet kernel, is

that *its $L^1$ norm can be bounded regardless of $N$*. In fact, we have

$$\|\mathscr{V}_N\|_{L^1} = \int_{\mathcal{Z}} |\mathscr{V}_N(z)| \mathrm{d}z =: \Lambda, \tag{5.3}$$

where $\Lambda$ only depends on $\mathcal{Z}$ (precisely, on $m$), not on $N$. Moreover, when $N$ is an even number, the operator corresponding to this kernel is a projection for $M = N/2$, i.e. if $f \in \mathrm{T}_{N/2}$, then $\mathscr{V}_N * f = f$. These two properties allow to prove that the projection corresponding to $\mathscr{V}_N$ enjoys, in case the function to approximate is smooth, the same approximation properties of theorem 33, up to a constant factor.

**Theorem 34.** *[Theorem 2, [80]] Let $f \in W_p^{\nu,\infty}(\mathcal{Z})$. Then, $\mathscr{V}_N * f \in T_N$ and the following inequality holds*

$$\|f - \mathscr{V}_N * f\|_{L^\infty} \leq \frac{2^\nu(\Lambda+1)J_T}{N^\nu}\|f\|_{W^{\nu,\infty}},$$

*where $J_T$ is the constant of theorem 33.*

This result follows from eq. (B.9) in the appendix, which states that for every $f \in L^\infty(\mathcal{Z})$,

$$\|f - \mathscr{V}_N * f\|_{L^\infty} \leq (\Lambda+1) \inf_{v \in \mathrm{T}_{N/2}} \|f - v\|_{L^\infty}.$$

When $f \in \mathrm{W}_{\mathrm{p}}^{\nu,\infty}(\mathcal{Z})$, which means that $f \in L^\infty(\mathcal{Z})$ a fortiori, and that theorem 33 applies, ensuring that

$$\inf_{v \in \mathrm{T}_{N/2}} \|f - v\|_{L^\infty} \leq \frac{\mathrm{J_T}}{(N/2)^\nu}\|f\|_{\mathrm{W}^{\nu,\infty}} = \frac{2^\nu \mathrm{J_T}}{N^\nu}\|f\|_{\mathrm{W}^{\nu,\infty}}.$$

These results answer both question of the previous subsection. The kernel $\mathscr{V}_N$ offers a good approximation, and the properties of convolution allows to build a target $\mathscr{V}_N * f$ by injecting noise on the argument on the function. The only remaining problem is that the $\mathscr{V}_N$ kernel is not always positive (see fig. 5.1), and therefore is not proportional to the density function of a random variable. In the next section, we will see how to solve this problem and implement the algorithm.

## 5.2. Algorithm

Algorithm 5.2 (PLSVI) takes the same inputs as the standard LSVI algorithm. Once again, we are assuming that the datasets of points to query are given in advance: to perform the analysis, we will need to specify them. Differently from the base method, algorithm 5.2 does not directly query the state-action pairs in $\mathcal{D}_h$ to the generative model.

Algorithm 5.2 Perturbed Least-Squares Value Iteration (PLSVI)

**Require:** Horizon $H$, Degree $N$ of the Fourier feature map, datasets $\{\mathcal{D}_h\}_{h=1}^H = \{\{(s_t, a_t)\}_{t=1}^{n_h}\}_{h=1}^H$
**Ensure:** Policy $\widehat{\pi}$

Initialize $\widehat{V}_{H+1}(s) \leftarrow 0$ for all $s \in \mathcal{S}$
**for** $h \leftarrow H, \dots, 1$ **do**
  **for** $z_h^t = (s_h^t, a_h^t) \in \mathcal{D}_h$ **do**
    Sample $\omega^+ \sim \mathscr{V}_N^+$
    Sample $\omega^- \sim \mathscr{V}_N^-$
    Query simulator at $z_h^t + \omega^+$ receiving next state/reward $s_{h+1}^{t+}, y_h^{t+}$
    Query simulator at $z_h^t + \omega^-$ receiving next state/reward $s_{h+1}^{t-}, y_h^{t-}$
    $q_h^t \leftarrow \left[\beta^+ y_h^{t+} - \beta^- y_h^{t-}\right] + \left[\beta^+ \widehat{V}_{h+1}(s_{h+1}^{t+}) - \beta^- \widehat{V}_{h+1}(s_{h+1}^{t-})\right]$
  **end for**
  Solve least squares

$$\widehat{\theta}_h = \operatorname*{arg\,min}_{\theta \in \mathbb{R}^{\tilde{N}}} \sum_{t=1}^{n_h} \left(\varphi_N(s_h^t, a_h^t)^\top \theta - q_h^t\right)^2$$

  $\widehat{Q}_h(\cdot, \cdot) \leftarrow \varphi_N(\cdot, \cdot)^\top \widehat{\theta}_h$
  $\widehat{V}_h(\cdot) \leftarrow \max_{a \in \mathcal{A}} \widehat{Q}_h(\cdot, a)$
**end for**
**Return** $\{\widehat{\pi}_h(\cdot)\}_{h=1}^H \leftarrow \{\arg\max_{a \in \mathcal{A}} \widehat{Q}_h(\cdot, a)\}_{h=1}^H$

At the core of this new method there is the projection by convolution technique that we described in the former subsection. That technique would require to perturb the samples with a noise coming from the function $\mathscr{V}_N$. However, we cannot take random samples from something that is not a density function, so we have to decompose $\mathscr{V}_N$ in its positive and negative parts. We define

$$\beta^+ = \int_{\mathcal{Z}} \max\{0, \mathscr{V}_N(z)\} \mathrm{d}z, \quad \beta^- = -\int_{\mathcal{Z}} \min\{0, \mathscr{V}_N(z)\} \mathrm{d}z. \tag{5.4}$$

and

$$\mathscr{V}_N^+(z) := \frac{\max\{0, \mathscr{V}_N(z)\}}{\beta^+}, \quad \mathscr{V}_N^-(z) := -\frac{\min\{0, \mathscr{V}_N(z)\}}{\beta^-}, \tag{5.5}$$

By definition, the identity

$$\mathscr{V}_N(z) = \beta^+ \mathscr{V}_N^+(z) - \beta^- \mathscr{V}_N^-(z)$$

holds, with $\mathscr{V}_N^+, \mathscr{V}_N^-$ being nonnegative functions with integral one and $\beta^+ + \beta^- = \Lambda$. PLSVI is based on sampling from a noise whose density corresponds to these functions, as it is done in the steps highlighted in in green (for $\mathscr{V}_N^+$) and red (for $\mathscr{V}_N^-$). In these

passages, instead of just querying the generative model for the elements of the buffers $\mathcal{D}_h$, in lines 6 and 7 we perturb them with a noise sampled from the positive and negative parts $\sim \mathscr{V}_N^+, \sim \mathscr{V}_N^-$ of the Vallée de la Poussin Kernel. The results are aggregated in line 8, giving $q_h^t$, which can be proved to be an unbiased estimator of the $\widetilde{Q}_h$ function we want to estimate. Finally, the least-squares step is performed in a standard way. The only thing missing for this algorithm to work is the choice of datasets $\mathcal{D}_h$.

### 5.2.1. Choosing Datasets with Optimal Design

The construction of the datasets $\mathcal{D}_h$ is the missing crucial part of Algorithm 5.2. This is done employing an idea from [69], i.e. by applying the concept of optimal design for the least squares estimator. On an high level, the optimal design for a least-square fitting problem is the distribution of query samples that minimizes the variance of the estimator. While [69] builds on the consideration that this approach induces a $\sqrt{d}$ expansion of the misspecification, our projection technique technique bypasses this issue.

Given a set $\mathcal{V} = \{v_t\}_t \subseteq \mathbb{R}^d$, not necessarily finite, a quasi-optimal design is defined as a distribution $\rho$ of finite support over $\mathcal{V}$ such that, said

$$G := \sum_{t=1}^{|\text{supp}(\rho)|} \rho(v_t) v_t v_t^\top,$$

one has $\|v\|^2_{G^{-1}} \leq 2d$, for every $v \in \mathcal{V}$[1]. Theorem 4.4 from [69] ensures that, whatever $\mathcal{V}$, it is possible to build an optimal design whose support do not exceed $4d \log\log(d)$ points. Ideally we want to sample $\mathcal{D}_h$ from the distribution $\rho$, at all time stages. This is made more difficult from the fact that the size of $\mathcal{D}_h$ must be an integer (there exist no "half-sample"), while $\rho$ may in principle put a weight that is irrational on some point of its support. For this reason, given a reference size $n$ for the dataset, we name

$$\mathcal{D}_h := \{z \times \lceil n\rho(z) \rceil : z \in \text{supp}(\rho)\}.$$

This means repeating every $z$ in the support of $\rho$ for a number of times given by $\lceil n\rho(z) \rceil$. On one side, every element $z \in \text{supp}(\rho)$ appears at least $n\rho(z)$ times in the dataset. On

[1] recall that for a positive definite matrix $A$ the norm $\|v\|^2_A$ stands for $v^\top A v$

the other, the overall size is bounded as follows

$$\begin{aligned}|\mathcal{D}_h| &= \sum_{z\in\text{supp}(\rho)} \lceil n\rho(z)\rceil \\ &\leq |\text{supp}(\rho)| + \sum_{z\in\text{supp}(\rho)} n_{\text{tot}}\rho(z) \\ &= |\text{supp}(\rho)| + n \leq 4d\log\log(d) + n.\end{aligned}$$

Therefore, the overall size of the dataset does not significantly exceed $n$, if $d \ll n$. The reason to choose the dataset in this way is that the following proposition then holds.

**Proposition 35.** *Let $\mathcal{D}$ be built as before for $\mathcal{V} \subseteq \mathbb{R}^d$ and, for each $v_t \in \mathcal{D}$, $y_t = v_t^\top\theta + \eta_t$, for some unknown $\theta \in \mathbb{R}^d$, and a random noise $\{\eta_t\}_{t=1}^{|\mathcal{D}|}$ that is independent and $\sigma-$subgaussianconditionally on $\mathcal{D}$. Define the LS estimator as*

$$\widehat{\theta} := G_n^{-1}\sum_{t=1}^{|\mathcal{D}|} v_t y_t \qquad G_n := \sum_{t=1}^{|\mathcal{D}|} v_t v_t^\top.$$

*Then, for every $\delta > 0$, fixing an arbitrary subset $\mathcal{V}' \subset \mathcal{V}$ with $|\mathcal{V}'| = k$,*

$$\mathbb{P}\left(\sup_{v\in\mathcal{V}'} |v^\top\theta - v^\top\widehat{\theta}| > \sqrt{\frac{2d\log(2k/\delta)}{n}}\sigma\right) \leq \delta.$$

The proof is standard, and we leave it to the appendix. For some $n$, choosing $\mathcal{D}_h$ according to the procedure explained in this section (with respect to $\mathcal{V} := \{\varphi_N(z), z \in \mathcal{Z}\}$) is the missing piece of PLSVI. The optimal design guarantees that the features are explored in the optimal way, the projection trick eliminates the problem of misspecification, and the standard LSVI structure allows for the sample complexity guarantees. All these facts, that we have anticipated in a heuristic way, are the backbone of the sample complexity guarantee for algorithm 5.2, which we are proving this rigorously in the next section.

## 5.3. Sample Complexity Analysis

We prove a sample complexity guarantee under both the Weakly Smooth (definition 24) and the Mildly Smooth (definition 23) assumptions. We start from the Weakly Smooth case, as it was the one studied in [80], while the other one is novel to this thesis. The proof strategy presented in [80] is particularly long and cumbersome; here we give one that is more elegant thanks to some results of [75].

### 5.3.1. Upper Bound for Weakly Smooth MDPs

All the propositions of this section are going to assume that the underlying MDP is Periodic Weakly Smooth as in definition 24. Also, we are assuming that algorithm 5.2 is run with datasets designed as in section 5.2.1 for some values $n_h$.

The main theorem will result from several smaller propositions. We divide the propositions that are necessary to prove the sample complexity guarantees into some subsections.

#### Bounding the bias (error of the surrogate $Q-$functions)

As PLSVI builds on the idea of approximating the surrogate function $\widetilde{Q}_h$, the first step of the sample complexity guarantee is to upper bound the discrepancy between the former and the optimal $Q-$function. The sequence $\{\widetilde{Q}_h\}_{h=1}^{H+1}$ is defined as follows

$$\forall h = 1, \dots H, \quad \widetilde{Q}_h := \mathscr{V}_N * \mathcal{T}_h \widetilde{Q}_{h+1} \qquad \widetilde{Q}_{H+1} := 0.$$

The next result proves its smoothness.

**Proposition 36.** *For any $h = 1, \dots H$*

$$\|\widetilde{Q}_h\|_{\mathrm{W}^{\nu,\infty}} \leq \frac{(\Lambda C_{\mathcal{T}}^{WS})^{H-h+1} - \Lambda C_{\mathcal{T}}^{WS}}{\Lambda C_{\mathcal{T}}^{WS} - 1}$$

*Proof.* The demonstration proceeds by backward induction. The base case $h = H + 1$ is trivially true, and we need to show the inductive step. It is well-known (see [39] chapter 3, properties of convolution) that for a kernel $\mathscr{K}$ and a function $f \in \mathrm{W}^{\nu,p}$ for any $p$, we have $\|\mathscr{K} * f\|_{\mathrm{W}^{\nu,p}} \leq \|\mathscr{K}\|_{L^1} \|f\|_{\mathrm{W}^{\nu,p}}$. Thus, using the one-norm of the kernel $\mathscr{V}_N$:

$$\|\widetilde{Q}_h\|_{\mathrm{W}^{\nu,\infty}} \leq \Lambda \|\mathcal{T}_h \widetilde{Q}_{h+1}\|_{\mathrm{W}^{\nu,\infty}} \leq \Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} (1 + \|\widetilde{Q}_{h+1}\|_{\mathrm{W}^{\nu,\infty}}).$$

Iterating this inequality for every $h$ gives

$$\|\widetilde{Q}_h\|_{\mathrm{W}^{\nu,\infty}} \leq \frac{(\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}})^{H-h+1} - \Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}}{\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} - 1},$$

which completes the proof. □

After proving the smoothness of $\widetilde{Q}_h$, we need to show that this function is close to the true optimal state-action value function.

**Proposition 37.** *For any $h = 1, \dots H$*

$$\|\widetilde{Q}_h - Q_h^*\|_{L^\infty} \leq 2^\nu H \Lambda J_T \frac{(\Lambda C_{\mathcal{T}}^{WS})^{H-h+1} - \Lambda C_{\mathcal{T}}^{WS}}{N^\nu (\Lambda C_{\mathcal{T}}^{WS} - 1)},$$

*Where $J_T$ is the constant defined in theorem 33.*

*Proof.* Again, we proceed by backward induction. The base case $h = H + 1$ is trivial, as both functions are null, and we need to show the inductive step. Let $1 \leq h \leq H$,

$$\begin{aligned}
\|\widetilde{Q}_h - Q_h^*\|_{L^\infty} &= \|\mathscr{V}_N * \mathcal{T}_h \widetilde{Q}_{h+1} - \mathcal{T}_h Q_{h+1}^*\|_{L^\infty} \\
&\leq \|\mathscr{V}_N * \mathcal{T}_h \widetilde{Q}_{h+1} - \mathcal{T}_h \widetilde{Q}_{h+1}\|_{L^\infty} + \|\mathcal{T}_h \widetilde{Q}_{h+1} - \mathcal{T}_h Q_{h+1}^*\|_{L^\infty} \\
&\leq \frac{2^\nu \Lambda \mathrm{J_T}}{N^\nu} \frac{(\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}})^{H-h+1} - \Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}}{\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} - 1} + \|\widetilde{Q}_{h+1} - Q_{h+1}^*\|_{L^\infty} \\
&\leq \frac{2^\nu \Lambda \mathrm{J_T}}{N^\nu} \sum_{\tau=h}^{H} \frac{(\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}})^{H-\tau+1} - \Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}}{\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} - 1} \\
&= 2^\nu H \Lambda \mathrm{J_T} \frac{(\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}})^{H-h+1} - \Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}}{N^\nu (\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} - 1)}.
\end{aligned}$$

The first passages hold by definition of $\widetilde{Q}_h$ and by triangle inequality. The third passage comes from theorem 34 replacing the Sobolev-Hölder norm of $\mathcal{T}_h Q_{h+1}$ with its upper bound given by proposition 36. The next step is done by just bounding the sum with $H$ times the maximum term. □

This proposition establishes that the discrepancy between $\widetilde{Q}_h$ and $Q_h^*$ is bounded by a term of order $\widetilde{\mathcal{O}}(N^{-\nu})$. Such a bound on the bias is, somehow, the easy part, as the real challenge is to prove that our estimator enjoys optimal variance despite the intricate convolution part.

## Results about variance

In this section, we prove by induction some results concerning the approximation rate of $\widehat{Q}_h$ to $\widetilde{Q}_h$ depending on the number of samples. For all this section, focus on stage $h$ with $1 \leq h \leq H$ and let

$$\varepsilon_{h+1} := \|Q_{h+1}^* - \widehat{Q}_{h+1}\|_{L^\infty} \qquad \epsilon_{h+1} := \|\widetilde{Q}_{h+1} - \widehat{Q}_{h+1}\|_{L^\infty}.$$

As the data generation procedure is independent in all the stages, we can treat $\widehat{Q}_{h+1}$ as a fixed deterministic function.

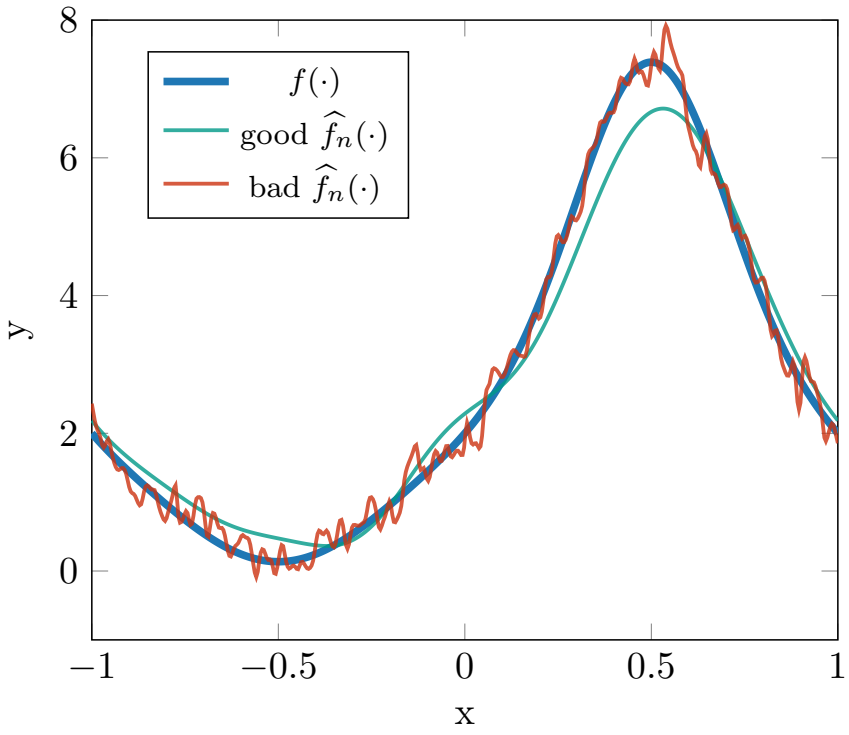



Figure 5.2: One of the central parts of this proof is to show that our function approximation strategy produces only smooth functions. As this figure shows, it is, in principle possible (red curve) to approximate a function with some $\widehat{f}_n$ that is very rough. While this approximation may seem good, our algorithm needs something like the the green curve: not only uniform approximation, but also smoothness properties that allow to enforce definition 24.

**Proposition 38.** *Fix $\delta > 0$ and $\mathcal{Z}' \subset \mathcal{Z}$ with $|\mathcal{Z}'| = k$. Then, with probability at least $1 - \delta$,*

$$\sup_{z \in \mathcal{Z}'} \left| \mathscr{V}_N * \mathcal{T}_h \widehat{Q}_{h+1}(z) - \widehat{Q}_h(z) \right| \leq \sqrt{\frac{2d \log(2k/\delta)}{n_h}} (H + 1 + \varepsilon_{h+1}).$$

The proof is long and technical, so we have left it in the appendix. The meaning of proposition 38 can be interpreted this way. Given any finite subset of points in $\mathcal{Z}$, we can obtain a uniform bound on the discrepancy between $\widehat{Q}_h$ and its target, which tends to zero as $\sqrt{d/n_h}$. The subtlety is that the result does not hold for all $\mathcal{Z}$: we cannot, in fact, make a union bound for infinitely many points. Nonetheless, to have the uniform bound we desire, it would be necessary to have the same result over the entire domain. The next proposition is the key to removing this finite-point constraint.

**Proposition 39.** *Let $f := \widehat{Q}_h - \mathscr{V}_N * \mathcal{T}_h \widehat{Q}_{h+1}$, and $\mathcal{Z}_\zeta$ be a $\zeta-$cover of $\mathcal{Z}$ in infinity norm.*

*Then, for* $\zeta \leq (8\pi N)^{-1}$,

$$\|f\|_{L^\infty} \leq 2 \sup_{z \in \mathcal{Z}_\zeta} |f(z)|.$$

This result may appear surprising, as it allows to bound uniformly a function starting from its bound on a finite set. In fact, the proof does not even use the specific form of $f$, but just the fact that it is a trigonometric polynomial of degree not larger than $N$. In fact, Theorem 12 from [75] ensures that, for every function of this type,

$$L(f) = \| \|\nabla f\|_\infty \|_{L^\infty} \leq 4\pi N \|f\|_{L^\infty}.$$

If the Lipschitz constant of the function is bounded in this way, by definition of $\zeta-$cover, it follows that

$$\|f\|_{L^\infty} \leq L(f)\zeta + \sup_{z \in \mathcal{Z}_\zeta} |f(z)| \leq 4\pi N \zeta \|f\|_{L^\infty} + \sup_{z \in \mathcal{Z}_\zeta} |f(z)|.$$

Using $\zeta \leq (8\pi N)^{-1}$ and rearranging, leads to the thesis of proposition 39. The former result allows to choose $\mathcal{Z}'$ as a $(8\pi N)^{-1}-$cover of $\mathcal{Z}$ in proposition 38. This leads to $k = (16\pi N)^m$ and, therefore, with probability at least $1-\delta$, to

$$\left\| \mathcal{V}_N * \mathcal{T}_h \widehat{Q}_{h+1} - \widehat{Q}_h \right\|_{L^\infty} \leq \sqrt{\frac{8dm \log(32\pi N/\delta)}{n_h}} (H + 1 + \varepsilon_{h+1}). \tag{5.6}$$

The last equation ensures that $\widehat{Q}_h$ is close to satisfying $\widehat{Q}_h = \mathcal{V}_N * \mathcal{T}_h \widehat{Q}_{h+1}$; the same relation holding exactly for $\widetilde{Q}_h$. This is enough to ensure that the two functions are close, as the following proposition establishes.

**Proposition 40.** *With probability at least* $1-\delta$,

$$\underbrace{\|\widetilde{Q}_h - \widehat{Q}_h\|_{L^\infty}}_{=\epsilon_h} \leq \Lambda \varepsilon_{h+1} + \sqrt{\frac{8dm \log(32\pi N/\delta)}{n_h}} (H + 1 + \varepsilon_{h+1}).$$

*Proof.* Simple algebraic manipulations show that

$$\begin{aligned}
\|\widetilde{Q}_h - \widehat{Q}_h\|_{L^\infty} &\leq \|\widetilde{Q}_h - \mathcal{V}_N * \mathcal{T}_h \widehat{Q}_{h+1}\|_{L^\infty} + \|\mathcal{V}_N * \mathcal{T}_h \widehat{Q}_{h+1} - \widehat{Q}_h\|_{L^\infty} \\
&= \|\mathcal{V}_N * \mathcal{T}_h \widetilde{Q}_{h+1} - \mathcal{V}_N * \mathcal{T}_h \widehat{Q}_{h+1}\|_{L^\infty} + \|\mathcal{V}_N * \mathcal{T}_h \widehat{Q}_{h+1} - \widehat{Q}_h\|_{L^\infty} \\
&= \Lambda \|\widetilde{Q}_{h+1} - \widehat{Q}_{h+1}\|_{L^\infty} + \|\mathcal{V}_N * \mathcal{T}_h \widehat{Q}_{h+1} - \widehat{Q}_h\|_{L^\infty} \\
&= \Lambda \epsilon_{h+1} + \|\mathcal{V}_N * \mathcal{T}_h \widehat{Q}_{h+1} - \widehat{Q}_h\|_{L^\infty}.
\end{aligned}$$

The thesis follows replacing (5.6). □

Proposition 40 shows a recursive relation that links $\epsilon_h$ to $\varepsilon_{h+1}$ and $\epsilon_{h+1}$. We can merge this result to what we proved in the previous section.

1. Proposition 37 establishes that

$$\varepsilon_h \leq \epsilon_h + \|\widetilde{Q}_h - Q_h^*\|_{L^\infty} \leq \epsilon_h + 2^\nu H \Lambda \mathrm{J_T} \frac{(\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}})^{H-h+1} - \Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}}{N^\nu (\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} - 1)}.$$

2. Proposition 40 shows that, with probability at least $1-\delta$,

$$\epsilon_h \leq \Lambda \varepsilon_{h+1} + \sqrt{\frac{8dm \log(32\pi N/\delta)}{n_h}}(H + 1 + \varepsilon_{h+1}).$$

Together, they prove that, with probability at least $1-\delta$,

$$\varepsilon_h \leq \Lambda \varepsilon_{h+1} + \sqrt{\frac{8dm \log(32\pi N/\delta)}{n_h}}(H + 1 + \varepsilon_{h+1}) + 2^\nu H \Lambda \mathrm{J_T} \frac{(\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}})^{H-h+1} - \Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}}{N^\nu (\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} - 1)}.$$

We can see the previous equation as a recurrence relation of the form $\varepsilon_h \leq a_h \varepsilon_{h+1} + b_h$, for $a_h = \Lambda + \sqrt{\frac{8dm \log(32\pi N/\delta)}{n_h}}$ and $b_h = \sqrt{\frac{8dm \log(32\pi N/\delta)}{n_h}}(H+1) + 2^\nu H \Lambda \mathrm{J_T} \frac{(\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}})^{H-h+1} - \Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}}{N^\nu (\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} - 1)}$. In this way, it follows that

$$\varepsilon_h \leq \sum_{i=h}^{H} \prod_{j=h}^{i-1} a_j b_i \leq \frac{a_\star^H - 1}{a - 1} b_\star \qquad b_\star = \max_{h=1,\dots H} b_h, \; a_\star = \max_{h=1,\dots H} a_h. \tag{5.7}$$

Fixing proper values for $N, n_h$ in this formula allows us to conclude.

## Final result: Fixing $N$ and $n$

In this section, we prove the main theorem of this chapter.

**Theorem 41.** *Assume access to a generative model. Let $\mathscr{M}$ the class of Periodic Weakly Smooth MDPs 24 satisfying 6 and 7. Then, the PLSVI algorithm 5.2 for $N = \Theta(\varepsilon^{-1/\nu})$ and simulation queries $\mathcal{D}_h$ built according to section 5.2.1 enjoys*

$$\boxed{\mathit{SampleComplexity}(PLSVI|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(C^H \varepsilon^{-2-m/\nu}\right)}.$$

*for some constant $C$.*

*Proof.* In order to guarantee that algorithm 5.2 outputs an $\varepsilon-$optimal policy, it is sufficient to ensure that for all $h$, $\|\widehat{Q}_h - Q_h^*\|_{L^\infty} \leq \varepsilon$. Assume eq. (5.7) holds at all time stages $h$. To ensure that the previous event has probability at least $1-\delta$, we need to change $\log(1/\delta)$ to $\log(H/\delta)$ (union bound). Let fix $\Psi := \frac{(\Lambda+1)^H - 1}{\Lambda}$ and let

$$N = \left( \frac{2^\nu \Psi \Lambda H^2 \mathrm{J_T} (\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}})^{H-h+1} - \Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}}{\varepsilon(\Lambda \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}} - 1)} \right)^{1/\nu}.$$

With this choice, $b_\star$ in eq. (5.7) does not exceed $\sqrt{\frac{8dm\log(32\pi NH/\delta)}{n_h}}(H+1) + \frac{\varepsilon}{2\Psi H}$. Taking, for all $h$,

$$n_h = \frac{32(H+1)^4 \Psi^2 dm \log(32\pi NH/\delta)}{\varepsilon^2},$$

we have $b_\star \leq \varepsilon/(\Psi H)$ and $a_\star \leq \Lambda + 1$. In this way,

$$\varepsilon_h \leq \frac{a_\star^H - 1}{a-1} b_\star \leq \frac{(\Lambda+1)^H - 1}{\Lambda} \frac{\varepsilon}{\Psi H} = \frac{\varepsilon}{H}.$$

This proves that the policy outputted by the algorithm is $\varepsilon-$optimal. Now, we have to count the total number of samples that the algorithm has queried to the generative model. First, note that $\sum_{h=1}^H n_h$ does not correspond to the exact total number. In fact, for every $z_h^t$, algorithm 5.2 has to query the simulator two times (lines 4 and 5). Moreover, the total size of the dataset $\mathcal{D}_h$ corresponds to $n_h + 4d\log\log(d)$, by section 5.2.1. So, the total number of samples is

$$n = \sum_{h=1}^H 2(n_h + 4d\log\log(d)) = \frac{64(H+1)^5 \Psi^2 dm \log(32\pi NH/\delta)}{\varepsilon^2} + 8Hd\log\log(d).$$

At this point, we know that $d = \Theta(N^m)$, and $N = \Theta(\varepsilon^{-1/\nu})$. Thus, the second part becomes a lower order term, and we can write

$$n \leq \widetilde{\mathcal{O}}\left(C^H \varepsilon^{-2-m/\nu}\right),$$

for a constant $C$. □

The result we just proved is order-optimal: it matches the lower bound 4.3 for the special case of smooth bandits. This theorem establishes that, in case of generative models, the same convergence rate can be achieved as for much simpler problems. Unfortunately,

the statistical complexity guarantee we demonstrated scales exponentially with the time horizon $H$. This phenomenon cannot be avoided; it stems from the nature of the Weakly Smooth MDP class. In the next section, we will therefore focus on a simpler case, the Mildly Smooth one, to demonstrate that this time the PLSVI algorithm manages to obtain a guarantee that scales polynomially with the horizon.

### 5.3.2. Upper Bound for Mildly Smooth MDPs

All the propositions of this section are going to assume that the underlying MDP is Periodic Mildly Smooth definition 23. Also, we are assuming that algorithm 5.2 is run with datasets designed as in section 5.2.1 for some values $n_h$.

Compared to the previous case, proving the sample complexity is easier, since the 23 assumption, unlike definition 24, does not require that the target function of the Bellman optimality operator be smooth, but only that it be bounded.

**Theorem 42.** *Assume access to a Generative model. Let $\mathscr{M}$ the class of Periodic Mildly Smooth MDPs 23 satisfying 6 and 7. Then, the PLSVI algorithm 5.2 for $N = \Theta(\varepsilon^{-1/\nu})$ and simulation queries $\mathcal{D}_h$ built according to section 5.2.1 enjoys*

$$\boxed{\mathit{SampleComplexity}(PLSVI|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(H^{7+4m/\nu}\varepsilon^{-2-m/\nu}\right)}.$$

*Proof.* We start proving by induction that, said $\varepsilon_h := \|Q^*_{h+1} - \widehat{Q}_{h+1}\|_{L^\infty}$, $\varepsilon_h \leq \frac{\varepsilon}{H}$ at any stage. We can still apply eq. (5.6) which ensure that, w.p. at least $1-\delta/H$ (we take $\delta/H$ to ensure that all the steps are valid with a global probability of $1-\delta$),

$$\left\|\mathscr{V}_N * \mathcal{T}_h\widehat{Q}_{h+1} - \widehat{Q}_h\right\|_{L^\infty} \leq \sqrt{\frac{8dm\log(32\pi NH/\delta)}{n_h}}(H+1+\varepsilon_{h+1}). \tag{5.8}$$

Here, the two proof strategies bifurcate. Indeed, as $\widehat{Q}_{h+1}$ is $\varepsilon_{h+1}-$away from $\widehat{Q}^*_h$, its $L^\infty$ norm must be bounded by $H-h-\varepsilon_{h+1}$. This entails, thanks to definition 23 that

$$\|\mathcal{T}_h\widehat{Q}_{h+1}\|_{\mathrm{W}^{\nu,\infty}} \leq \mathrm{C}^{\mathrm{MS}}_{\mathcal{T}}(H+1-h+\varepsilon_{h+1}).$$

At this point, theorem 34 applies, which ensures that

$$\|\mathscr{V}_N * \mathcal{T}_h\widehat{Q}_{h+1} - \mathcal{T}_h\widehat{Q}_{h+1}\|_{\mathrm{W}^{\nu,\infty}} \leq \frac{2^\nu(\Lambda+1)\mathrm{J}_{\mathrm{T}}\mathrm{C}^{\mathrm{MS}}_{\mathcal{T}}(H+1-h+\varepsilon_{h+1})}{N^\nu}.$$

Combining it with eq. (5.8) gives

$$\left\|\mathcal{T}_h\widehat{Q}_{h+1}-\widehat{Q}_h\right\|_{L^\infty}\leq \sqrt{\frac{8dm\log(32\pi NH/\delta)}{n_h}}(H+1+\varepsilon_{h+1})+\frac{2^\nu(\Lambda+1)\mathrm{J_T}\mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}(H+1-h+\varepsilon_{h+1})}{N^\nu}.$$

The former equation allows to relate $\varepsilon_h$ to the previous value $\varepsilon_{h+1}$.

$$\begin{aligned}\varepsilon_h&=\|Q_h^*-\widehat{Q}_h\|_{L^\infty}\\&\leq\|\mathcal{T}_hQ_{h+1}^*-\mathcal{T}_h\widehat{Q}_{h+1}\|_{L^\infty}+\|\mathcal{T}_h\widehat{Q}_{h+1}-\widehat{Q}_h\|_{L^\infty}\\&\leq\|Q_{h+1}^*-\widehat{Q}_{h+1}\|_{L^\infty}+\|\mathcal{T}_h\widehat{Q}_{h+1}-\widehat{Q}_h\|_{L^\infty}\\&\leq\varepsilon_{h+1}+\|\mathcal{T}_h\widehat{Q}_{h+1}-\widehat{Q}_h\|_{L^\infty}.\end{aligned}$$

By induction, $\varepsilon_{h+1}\leq\varepsilon/H\leq 1$, so this shows that

$$\begin{aligned}\varepsilon_h&\leq\sum_{i=h}^{H}\sqrt{\frac{8dm\log(32\pi N/\delta)}{n_i}}(H+2)+\frac{2^\nu(\Lambda+1)\mathrm{J_T}\mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}(H+2-i)}{N^\nu}\\&\leq\sqrt{\frac{8dm\log(32\pi N/\delta)}{n_i}}(H+2)^2+\frac{2^\nu(\Lambda+1)\mathrm{J_T}\mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}(H+1)^2}{N^\nu}.\end{aligned}$$

Replacing

$$N=2\left(\frac{(\Lambda+1)\mathrm{J_T}\mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}(H+1)^4}{\varepsilon}\right)^{1/\nu},$$

the second term is bounded by $\frac{\varepsilon}{2(H+1)}$. Taking

$$n_h=\frac{48dm\log(32\pi N/\delta)(H+2)^6}{\varepsilon^2},$$

the first term is also bounded by $\frac{\varepsilon}{2(H+1)}$. Thus, $\varepsilon_h\leq\frac{\varepsilon}{H}$, which completes the inductive step.

With previous passages, we have proved that, with probability at least $1-\delta$, $\|Q_h^*-\widehat{Q}_h\|_{L^\infty}\leq\frac{\varepsilon}{H}$, which is enough to ensure that the optimal policy outputted by algorithm 5.2, which is greedy w.r.t. $\widehat{Q}$, is $\varepsilon$-optimal. Now, we have to compute the total number of queries to the generative model that were necessary to prove this result. The same argument shown in theorem 41 bounds the number of samples with

$$n=\sum_{h=1}^{H}2(n_h+4d\log\log(d))=\widetilde{\mathcal{O}}\left(\frac{dH^7}{\varepsilon^2}\right).$$

Taking into account that $d = \Theta(N^m)$, our choice of $N$ corresponds to $d = \Theta\left(\left(\frac{H^4}{\varepsilon}\right)^{m/\nu}\right)$. This completes the proof. □

We have shown that for the Mildly Smooth MDP, it is possible to achieve the optimal complexity order $\varepsilon^{-2-m/\nu}$ without an exponential dependence on the time horizon $H$. Furthermore, the algorithm for solving the two problems is always algorithm 5.2. A relatively weak point of this algorithm is that choosing $N$ requires knowledge of $\nu$, the smoothness parameter of the MDP class. Without this, the choice of $N$ may be incorrect. In the next section, we will try to understand what happens when our algorithm receives a value for $\nu$ that is not correct.

### 5.3.3. Misspecification of the Smoothness Parameter

As proved in theorems 41 and 42, PLSVI is able to achieve order-optimal sample complexity for every Weakly/Mildly Smooth MDP. Still, to achieve this result, an exact knowledge of the smoothness parameter $\nu$ is required, as it is not possible to compute the optimal number of degree $N$ of the feature map without it. In many practical scenarios, the knowledge of the exact $\nu$ is rather unrealistic, so we study what happens in case the algorithm is run for a parameter $\nu_{\text{miss}}$ that is different from the correct value $\nu$. As both our main theorems fix the degree to be $N = \Theta(\varepsilon^{-1/\nu})$, this corresponds to studying what happens when running the algorithm for $N = \Theta(\varepsilon^{-1/\nu_{\text{miss}}})$. The two cases, namely overestimation ($\nu_{\text{miss}} > \nu$) and underestimation ($\nu_{\text{miss}} < \nu$) behave in two different ways.

Taking $\nu_{\text{miss}} > \nu$ leads to the impossibility of guaranteeing the learning of an $\varepsilon-$optimal policy: no sample complexity guarantee can be proved. As we prove in theorem 34, which is the backbone of all our results of this chapter, the error uniform error for approximating a smooth function decays as $N^{-\nu}$. Thus, taking $N = \Theta(\varepsilon^{-\frac{1}{\nu_{\text{miss}}}})$ leads to

$$\forall \varepsilon < 1 \qquad \|\widetilde{Q}_h - Q_h^*\|_\infty = \Theta\left(\varepsilon^{\frac{\nu}{\nu_{\text{miss}}}}\right) > \Omega(\varepsilon).$$

Clearly, this kind of approximation does not allow us to learn a $\varepsilon-$optimal policy, even ruling out the variance of the estimator.

In the opposite case, when $\nu_{\text{miss}} < \nu$, we can just exploit the fact that $\mathrm{W}_{\mathrm{p}}^{\nu,\infty}(\mathcal{Z}) \subset \mathrm{W}_{\mathrm{p}}^{\nu_{\text{miss}},\infty}(\mathcal{Z})$ and apply the bound we would obtain if the MDP were only smooth of order $\nu_{\text{miss}}$. This yields

$$n = \widetilde{\mathcal{O}}\left(\varepsilon^{-2-\frac{m}{\nu_{\text{miss}}}}\right),$$

| Class | Sample complexity order | Polynomial in $H$? |
|---|---|---|
| Strongly Smooth | $\varepsilon^{-2-\frac{m}{\nu}}$ | ✓ |
| Mildly Smooth | $\varepsilon^{-2-\frac{m}{\nu}}$ | ✓ |
| Weakly Smooth | $\varepsilon^{-2-\frac{m}{\nu}}$ | ✗ |

Table 5.1: In this table, we have reported the main results presented in this chapter. All the guarantees are relative to the PLSVI algorithm.

which is obviously worse than the one for $\nu$, but still finite.

**Is it possible to adaptively choose the smoothness parameter during learning?** The aforementioned results motivate the search for an algorithm capable of learning the MDP's smoothness parameter during the experiment.

A similar challenge in the bandit setting was addressed by [74] in Section 4 ("Adaptation to Unknown Smoothness"). This problem is particularly subtle, as the global nature of the smoothness parameter precludes the existence of a simple estimator. Consequently, a viable approach would be to build upon the CORRAL framework developed by [98]. While this methodology could potentially be adapted to our scenario, achieving adaptation to the unknown smoothness parameter without sacrificing the tightness of the sample complexity bound remains a major open challenge.

## 5.4. Final Considerations on Periodic Smooth MDPs and PLSVI

In this chapter we have answered the main questions regarding statistical complexity guarantees for smooth MDPs when having access to a generative model. The proposed algorithm is the same for all MDP classes considered, and its performance guarantee scales as $\varepsilon^{-2-m/\nu}$ in all cases. This result corresponds to the lower bound for Smooth Bandits instances, which we described at the beginning of the previous chapter; therefore, it cannot be improved. For Strongly and Mildly Smooth instances, we showed that the upper bound scales polynomially with respect to $H$. For the Weakly Smooth class, this result is not possible, as a lower bound of order $\exp(H)$ exists.

The main results we have demonstrated are reported in table 5.1. Comparing them with the data in table 4.1 shows that the sample complexity results for the two classes TV-Lipschitz and $\mathcal{W}$-Lipschitz are extended by our results for Mildly/Weakly Smooth MDPs.

As we demonstrated in the previous chapter, the two Lipschitz classes are included in the two Smooth classes for $\nu = 1$. For this choice of smoothness parameter, the theorems in this chapter reduce to the known state-of-the-art results for Lipschitz classes.

Since Kernelized MDPs are a subset of our Strongly Smooth MDPs, which, in turn, are a special case of the Mildly Smooth MDPs studied in this chapter, our sample complexity results also extend to the kernelized setting, achieving the optimal order of complexity. This result is particularly noteworthy because state-of-the-art regret bounds for kernelized MDPs, including those for GOLF (which applies to all the families considered here), were not only suboptimal but failed to provide any provable sample complexity, as they violated the sublinear regret hypothesis. It should be noted, however, that while those works assumed an online interaction between the agent and the environment, this chapter focuses on the simpler generative model setting.

To conclude this chapter, we will address some peripheral yet compelling aspects related to the topics discussed so far. First, we will analyze the PLSVI algorithm from a computational complexity perspective. Next, we will examine the fundamental periodicity assumption to determine whether the constraints introduced in this chapter lead to significant departures from the standard Smooth MDP setting. Finally, we will explore an intriguing connection with non-parametric statistics, illustrating how our reinforcement learning problem essentially reduces to a problem of uniform estimation.

### 5.4.1. Computational Complexity

The computational complexity of our algorithm can be divided into the one for the preliminary operations to build the dataset $\mathcal{D}_h$ and the one for executing the actual routine in algorithm 5.2.

Algorithmically, in order to compute the optimal design, it is necessary to cover the space $\mathcal{V} = \{\varphi_N(z) : z \in \mathcal{Z}\}$. This cover is performed by fixing $\varepsilon'$ and constructing an $\varepsilon'-$grid of the state-action space, which amounts to $k = (2/\varepsilon')^m$ points (note that, even if $\mathcal{V}$ is a subset of $\mathbb{R}^d$, it is only an $m-$dimensional variety, as the map $\varphi_N : \mathcal{Z} \to \mathbb{R}^d$ is continuous). As the matrix norm is continuous, this introduces an error that can be made arbitrarily small with $\varepsilon'$. Once this is done, the optimal design can be computed in just $kd^2$ steps (see [69] just before theorem 4.4). The learning part is not particularly heavy: sampling from the positive and negative parts of the $\mathscr{V}_N$ may be done in time $\mathcal{O}(d)$ with MCMC methods, which work well even for a dimension $m$ of the order of tens (which is a relatively big number in the context of nonparametric RL). The main computational bottleneck of the algorithm stays in the last step, where it requires to solve a linear regression problem.

This is well known to take $nd^2 + d^3$ steps, where $n$ is the number of samples. Therefore, the total computational complexity is roughly of order

$$\mathcal{O}\left(\underbrace{kd^2}_{\text{optimal design}} + \underbrace{nd}_{\text{sampling from kernel}} + \underbrace{nd^2 + d^3}_{\text{solving linear regression}}\right) = \mathcal{O}((k+n)d^2).$$

This bound can be refined replacing $k, d, n$, that are all built by the algorithm, with the native variables $\varepsilon, H, m, \nu$. For what concerns $k$, we can assume to take $\varepsilon' = \varepsilon$, so that its magnitude is of the order of $\varepsilon^{-m}$. As we have seen the proof of theorem 42, for Mildly Smooth MDPs the optimal choice of $N$ is of the order of

$$N = \Theta\left(\frac{H^4}{\varepsilon}\right)^{1/\nu},$$

which means that

$$d = \Theta\left(\frac{H^{4m/\nu}}{\varepsilon^{m/\nu}}\right) \qquad n = \widetilde{\Theta}\left(\frac{H^{7+4m/\nu}}{\varepsilon^{2+m/\nu}}\right).$$

With this choice of the parameters, the total computational complexity of the algorithm writes as

$$\text{Computational Comp.(PLSVI)} = \widetilde{\mathcal{O}}\left(\max\left\{\frac{H^{8m/\nu}}{\varepsilon^{m+2m/\nu}}, \frac{H^{7+12m/\nu}}{\varepsilon^{2+3m/\nu}}\right\}\right).$$

The term $m/\nu$, that appears as the exponent of all variables, suggests that the former bound is only good for a dimension $m$ that is not big compared to $\nu$. For a smoothness parameter that is much higher than $m$, the complexity reduces to $\max\{\varepsilon^{-m}, H^7\varepsilon^{-2}\}$. Unfortunately, the exponent of the time horizon is still relatively big, but this is a known problem that we inherit from LSVI [4].

### 5.4.2. Are Periodic Smooth MDPs Easier than Smooth MDPs?

At first sight, the periodicity assumption introduced in this chapter may appear to substantially restrict the class of admissible environments. Indeed, requiring periodic boundary conditions enforces a strong form of regularity at the edges of the state–action space, which is not part of the original definition of Smooth MDPs. One might therefore suspect that periodic Smooth MDPs constitute an easier subclass, for which sharper guarantees can be obtained only thanks to the artificial removal of boundary effects. In this section, we argue that this intuition is misleading: periodic Smooth MDPs are essentially equivalent to standard Smooth MDPs from a statistical complexity perspective.

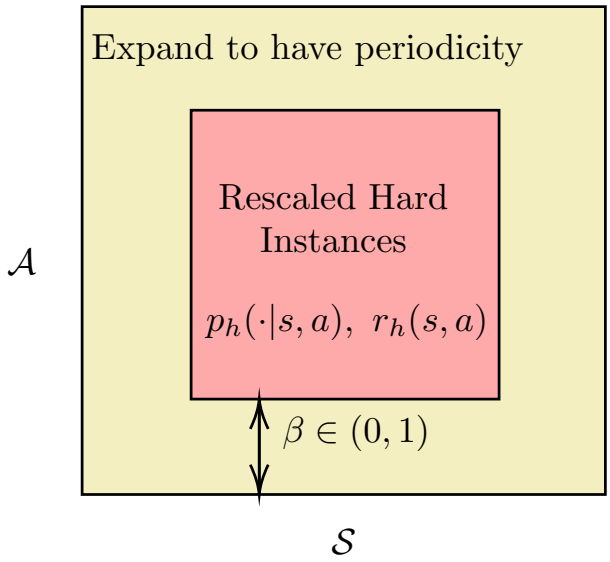



Figure 5.3: The red square this figure represents the state-action space of a Smooth MDP. The intuition is that, given an "hard instance" as a Smooth nonperiodic MDP, we can rescale its state-action space from $[-1,1]^m$ to $[-1+\beta, 1-\beta]^m$ and extend both $r_h$ and $p_h$ in the inner space between the two squares in a way that preserves smoothness while achieving periodicity at the boundaries of $[-1,1]^m$.

The key observation is that, for smooth function classes, boundary behavior can be modified without altering the intrinsic difficulty of approximation or estimation in the interior of the domain. This phenomenon is well understood in approximation theory and Sobolev analysis, where extension results play a central role. In particular, classical extension theorems ensure that functions defined on compact domains with sufficient smoothness can be extended to larger ones while preserving smoothness.

More concretely, let $f \in \mathrm{W}^{\infty,\nu}(\mathcal{Z}')$ for some $\mathcal{Z}' \Subset \mathcal{Z}$ (meaning that the closure of $\mathcal{Z}'$ does not touch the boundary of $\mathcal{Z}$). The Calderón-Zygmund-Stein theorem (see theorem 5 in [121]), ensures that there exists an extension operator

$$\mathrm{E} : \mathrm{W}^{\infty,\nu}(\mathcal{Z}') \to \mathrm{W}^{\infty,\nu}(\mathcal{Z})$$

such that, for any $f$,

$$\mathrm{E}(f) = f \text{ on } \mathcal{Z}', \qquad \forall |\alpha| \le \nu; D^\alpha f = 0 \text{ on } \partial\mathcal{Z}.$$

for a constant $C$ depending only on $\mathcal{Z}', m, \nu$. Thanks to the condition on the boundary, such an extension always produces periodic and smooth of the same order. Importantly, this construction preserves $f$ inside $\mathcal{Z}'$. This observation allow the following argument

(for a visual intuition, see fig. 5.3). Given a "hard" non-periodic Smooth MDP instance, one can rescale the state–action space to a slightly smaller hypercube $\mathcal{Z}' = [-1+\beta, 1-\beta]$, for some $\beta > 0$, and extend rewards and transition kernels smoothly to the remaining boundary layer by means of the extension operator E.

From a minimax standpoint, this implies that periodic Smooth MDPs cannot be significantly easier than standard Smooth MDPs. Any lower bound construction for the non-periodic case can be embedded into the periodic setting by the extension argument above, yielding the same dependence on the accuracy parameter $\varepsilon$, the dimension $m$, and the smoothness order $\nu$. This is consistent with the fact that the sample-complexity upper bounds proved in this chapter match known lower bounds for smooth bandits and continuous control problems, where no periodicity assumption is made.

**Extending the current algorithm to general Smooth MDPs** As the previous analysis suggest that the lower bound for Smooth MDPs coincides with the one for Periodic Smooth MDPs, it is legitimate to ask if the algorithms presented in this chapter may be extended to the general case. While the Fourier features used are intrinsically periodic, this is possible, under proper assumptions, via an extension argument. Specifically, for what concerns smooth regression, Appendix G in [75] shows that it is possible to extend order-optimal sample complexity upper bounds to the non-periodic case *when queries are allowed to be outside of the domain.* As the approach proposed in this chapter decomposes the algorithm in a sequence of $H$ regression problems, this result could easily be adapted to our scenario. Nonetheless, we prefer not to focus on this generalization, as assuming in our context that the simulator allows to query a pair $(s, a) \notin \mathcal{S} \times \mathcal{A}$ is difficult to motivate in the light of real applications.

### 5.4.3. Comparison with Non-parametric Statistics

In section 4.1.3 of the previous chapter, we compared the results for continuous-armed bandits with those from non-parametric statistics. The conclusion, as previously suggested by [143], was that optimizing a smooth function from noisy data is, in the worst-case scenario, as difficult as approximating it uniformly.

The results presented in this chapter offer further insight. While the bandit setting requires optimizing a smooth function and non-parametric statistics focuses on uniform approximation, our RL setting with a generative model requires performing both tasks simultaneously. Specifically, one must first obtain a uniform estimate $\widehat{Q}_{h+1}$ of $Q^*_{h+1}$, then compute $\widehat{V}_{h+1}$ by maximizing $\widehat{Q}_{h+1}$ over the action space, and finally use this new function

to construct the target for the subsequent step. In this chapter, we demonstrated through algorithm 5.2 that the overall statistical complexity remains $\widetilde{\mathcal{O}}(\varepsilon^{-2-m/\nu})$, consistent with the individual cases mentioned above. The fact that the problem spans $H$ stages impacts the complexity only by a constant factor and does not alter the method's rate of convergence.

# 6 | Smooth MDPs: Regret Bounds

Our results in the generative model case show that it is possible to learn a $\varepsilon-$optimal policy across all Smooth MDP classes in a statistically efficient way (although in the Weakly Smooth case, the guarantees scale exponentially in $H$). Moreover, the sample complexity result we obtained turns out to be order-optimal even in the much simpler bandit scenario.

In this chapter, we will focus on a more challenging and realistic setting: online interaction. The theoretical goal in this case, proving a bound on regret, is more ambitious than bounding sample complexity. Furthermore, perhaps more importantly, having to interact with the environment episodically online creates another set of difficulties that we have not encountered so far, even if we were only trying to obtain sample complexity results. The fact that the samples collected in each episode $k$, of the form $(s_h^k, a_h^k, s_{h+1}^k)$, depend on the policy $\pi^k$, which is chosen by the agent on the basis of the data collected up to that point, creates an intricate statistical dependence between the samples, which weakens all the concentration inequalities that would be necessary to demonstrate the sample complexity bounds.

The existing literature on theoretical Reinforcement Learning is replete with problem classes where proving an optimal regret bound in the online interaction setting is significantly more challenging than establishing sample complexity bounds for the generative model case. A prime example is that of Tabular MDPs: while this class has been studied since the turn of the century and sample complexity results were established over a decade ago (see section 3.3.2), an algorithm with a provably optimal regret bound (see theorem 14) was only developed in 2024. Furthermore, there are instances where obtaining results for online interaction is intrinsically more difficult than in the generative model scenario, such as the case described in proposition 22. Where do Smooth MDPs fit into this landscape? In this chapter, we aim to address this question.

## 6.1. Regret Bound for Weakly Smooth MDPs

In the last chapter we used trigonometric polynomials to approximate the state-action value function at any time-step $h$. The reason for this choice was dictated by the need to use the de la Valleè Poussin kernel $\mathscr{V}_N$ to circumvent the problem of misspecification amplification. Unfortunately, this strategy requires having an environment simulator available, and does not apply to our online interaction scenario.

For this reason, the results in this chapter will employ polynomial-based feature maps, which offer similar approximation results without requiring periodic boundary conditions. We use a notation similar to the previous chapter for $\mathcal{S}, \mathcal{A}, \mathcal{Z}$. This time, we write $\mathrm{P}_N(\mathcal{Z})$ (or just $\mathrm{P}_N$) for the vector space of polynomials on $\mathcal{Z}$ whose degree does not exceed $N$, and $\varphi_N : \mathcal{Z} \to \mathbb{R}^d$ the Legendre polynomial feature map of degree $N$ (normalized to one) on $\mathcal{Z}$. In one dimension, this map is given by

$$\varphi_N(z) = \left[\frac{1}{2}, \frac{3z}{2}, \frac{5z^2-1}{4}, \dots, \frac{2N+3}{2} P_{N+1}(z)\right],$$

where $P_N$ is the $N-$th standard Legendre polynomial. The reason why we do not use the feature map given by the standard Legendre is polynomials their normalization condition:

$$\int_{-1}^{1} P_N^2(x)\mathrm{d}x = \frac{2}{2N+1}.$$

Instead, in our definition the features are normalized so that for each of its components $\varphi_N^i(x)$ one has $\|\varphi_N^i\|_{L^2} = 1$, like the Fourier features. In the one dimensional case, the length of $\varphi_N$ is just $N+1$. For general dimensions $m$ this number increases to $d := \binom{N+m}{m} \leq (N+1)^m$.

As we have seen, the key to the success of the Fourier feature map in Smooth MDPs was theorem 33, which ensured optimal approximations for smooth functions using this basis. In the context of standard polynomials, a very similar theorem holds.

**Theorem 43.** *[Theorem 8, in [79]] There is a constant $J_P > 0$, only depending on $m$, such that for any $f \in W^{\nu,\infty}(\mathcal{Z})$*

$$\min_{\substack{g \in P_N \\ \|g\|_{W^{\nu,\infty}} \leq 3J_P \|f\|_{W^{\nu,\infty}}}} \|f - g\|_{L^\infty} \leq \frac{J_P}{N^\nu} \|f\|_{W^{\nu,\infty}}.$$

This theorem frees us from needing periodicity conditions at the boundary. That requirement was entirely inherent to the Fourier feature map; without it, all subsequent results

in this chapter naturally extend to smooth, non-periodic MDPs. What is fundamentally different from its analogous result theorem 33 is the presence of the condition

$$\|g\|_{\mathrm{W}^{\nu,\infty}} \leq 3\mathrm{J}_{\mathrm{P}}\|f\|_{\mathrm{W}^{\nu,\infty}}.$$

In this case, the theorem ensures that the approximator function $g$ is roughly as smooth as $f$. While $g$ is always infinitely differentiable, in theorem 33 its norm could, in principle, be much bigger than the one of $f$. In the last chapter, we solved this problem employing the properties of convolution. This time, the approximation theorem rules out this scenario automatically. The proof of theorem 43 is based on theorem 4 by [17], plus a convolution argument to show the bound on $\|g\|_{\mathrm{W}^{\nu,\infty}}$.

The first algorithm proposed in this chapter leverages established Reinforcement Learning techniques for Linear MDPs and environments characterized by low inherent Bellman error. While the value function of a Smooth MDP will never be exactly linear in $\varphi_N$, the approximation theorem suggests that the results from the literature for linear models with misspecification, in particular the classes 9 and 10 classes, could be suitable.

### 6.1.1. Reducing to MDPs with Low Inherent Bellman error

In this section we try to adapt the results of theorem 20 by [151] to our setting, and to apply its ELEANOR algorithm to Weakly Smooth MDPs. As from definition 10, an MDP with low inherent Bellman error must satisfy what follows. Said

$$Q_h[\theta](z) := \varphi_N(z)^\top \theta,$$

there is a sequence $\{\mathcal{B}_h\}_{h=1}^H$ of subsets of $\mathbb{R}^d$ such that

$$\forall z \in \mathcal{Z} \qquad \sup_{\theta \in \mathcal{B}_{h+1}} \inf_{\theta' \in \mathcal{B}_h} |Q_h[\theta'](z) - \mathcal{T}_h Q_{h+1}[\theta](z)| \leq \mathcal{I},$$

for some misspecification $\mathcal{I} > 0$. In order to apply the regret bound for this class, theorem 20, it is necessary to satisfy the conditions in (3.7). In particular, the choice of $\mathcal{B}_h$ is crucial to show any theoretical guarantees.

**Choosing the sets $\mathcal{B}_h$.** We construct sets $\mathcal{B}_h$ with the following form, for some $B(h) \geq 0$ to be determined.

$$\mathcal{B}_h := \left\{\theta \in \mathbb{R}^d : \|\varphi_N(\cdot)^\top \theta\|_{\mathrm{W}^{\nu,\infty}} \leq B(h) \cap \|\varphi_N(\cdot)^\top \theta\|_{L^\infty} \leq 2H\right\}, \tag{6.1}$$

If $\theta \in \mathcal{B}_{h+1}$ then, by definition of Weakly Smooth MDP,

$$\|\mathcal{T}_h Q_{h+1}[\theta]\|_{\mathrm{W}^{\nu,\infty}} \leq \mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}(B(h+1)+1).$$

Applying theorem 43, this ensures that there is a polynomial $g \in \mathrm{P}_N$ such that

$$\|\mathcal{T}_h Q_{h+1}[\theta] - g\|_{L^\infty} \leq \frac{\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}}(B(h+1)+1)}{N^\nu}, \qquad \|g\|_{\mathrm{W}^{\nu,\infty}} \leq 3\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}}(B(h+1)+1).$$

Therefore, choosing $B(h) = 3\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}}(B(h+1)+1)$, by definition there is $\theta' \in \mathcal{B}_h$ that satisfies $g(z) = \varphi_N(z)^\top \theta' = Q_h[\theta'](z)$. Thus, for this choice

$$\|\mathcal{T}_h Q_{h+1}[\theta] - Q_h[\theta']\|_{L^\infty} \leq \frac{\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}}(B(h+1)+1)}{N^\nu}.$$

The previous equation allows us to express $B(h)$ in a recursive way as $B(h) = 3\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}}(B(h+1)+1)$. Recalling that $B(H+1) = 0$, the solution to this recursion corresponds to

$$B(h) = \sum_{\tau=h}^{H} (3\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}})^{H-\tau} = \frac{(3\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}})^{H-h+1} - 3\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}}}{3\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}} - 1}. \tag{6.2}$$

By choosing $\mathcal{B}_h$ in this way, the misspecification error is bounded as $\mathcal{I} \leq \widetilde{\mathcal{O}}\left(\frac{(\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}})^H}{N^\nu}\right)$. Now, to prove the regret bound it is necessary to verify if this choice satisfies assumptions (3.7). The first condition requires that $\forall z \in \mathcal{Z}, h \in [H], \|\varphi_N(s,a)\|_2 \leq 1$. Let us try to estimate this number:

$$\begin{aligned}
\|\varphi_N(z)\|_2 &= \left\| \left\{ \prod_{j=1}^{m} \frac{2i_j+3}{2} P_{i_j+1}(z) : \sum_{j=1}^{m} i_j \leq N \right\} \right\|_2 \\
&= \sqrt{\sum_{\substack{i_1,\dots i_m=1 \\ i_1+\cdots+i_m \leq N}} \left( \prod_{j=1}^{m} \frac{2i_j+3}{2} \right)^2} \\
&\leq \sqrt{\sum_{\substack{i_1,\dots i_m=1 \\ i_1+\cdots+i_m \leq N}} \left( \frac{2N+3m}{2m} \right)^{2m}} \\
&\leq \sqrt{d \left( \frac{2N+3m}{2m} \right)^{2m}} = \Theta(d^{3/2}).
\end{aligned}$$

The first passage comes by making the cartesian product of all the vectors of uni-variate Legendre polynomials. Then, one notes that the optimal way to split $i_1 + \cdots + i_m \leq N$ to maximize the product $\prod_{j=1}^{m} \frac{2i_j+3}{2}$ is to take all the terms equal to $N/m$. The last step

is due to the fact that $d = \mathcal{O}(N^m)$. This result is far from what we needed: theorem 20 cannot be applied in this scenario.

**Refining the regret bound for MDPs with Low-Inherent Bellman error.** While we cannot apply the result of [151] as it is, it is possible to get rid of the problematic assumption by a small refinement of the statement. Theorem 20 assumed a bound $L_\phi$ for the 2−norm of the feature map, with $L_\phi \leq 1$. In our scenario, $L_\phi$ grows relatively fast with $d$, which compromises the current analysis and regret bound. The naive solution of dividing $\varphi_N$ by $d^{3/2}$ to move this coefficient to the parameter $\theta$ does not work, as the norm of this parameter needs to be bounded by $\sqrt{d}$. The only thing that is left to do is to analyze the proofs of [151] to understand where this assumption is employed, hoping to obtain a regret bound that is more suited to our problem.

In the proof of Theorem 1 by [151], $L_\phi$ is used to bound a martingale difference a trough Hoeffding's inequality. The argument is what follows (in our notation):

> Since $\overline{V}^k_{h+1}(s) = \max_{a\in\mathcal{A}} \varphi_N(s,a)^\top \theta_{h+1}$ for some $\theta_{h+1} \in \mathcal{B}_{h+1}$, then $\overline{V}^k_{h+1}(s) \leq L_\phi \mathcal{R}_{h+1}$ by Cauchy-Schwartz inequality.

To tighten the argument, instead of using the Cauchy-Schwartz inequality, we can consider that

$$|\overline{V}^k_{h+1}(s)| = \sup_{s,a\in\mathcal{Z}} \sup_{\theta\in\mathcal{B}_h} |\varphi_N(s,a)^\top \theta| = \sup_{\theta\in\mathcal{B}_h} \|\varphi_N(\cdot)^\top \theta\|_{L^\infty} \leq 2H. \tag{6.3}$$

This passage cannot be carried out in the original paper by [151], as they assumed the random variable $y^k_h + \max_a Q_h[\theta](s^k_{h+1}, a)$ (minus its mean) to be sub-Gaussian without explicitly requiring the function to be bounded.

Except for this point, in the proofs of [151] $L_\phi$, only appears inside logarithms. After this changes, their main theorem reads as follows.

**Theorem 44.** *Let $\mathscr{M}$ the class of MDPs with low inherent Bellman error, assuming online interaction and assumptions 6 and 7. Let $L_\phi := \sup_{s,a,h} \|\varphi_h(s,a)\|_2$, $\mathcal{R}_h := diam(\mathcal{B}_h)$ and assume that $\sup_{h,\theta\in\mathcal{B}_h} \|Q_h[\theta]\|_\infty \leq 2H$. Then, the* Eleanor *algorithm enjoys*

$$RegretBound(\textsc{Eleanor}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left( H\left( H\sqrt{d} + \sum_{h=1}^{H} \mathcal{R}_h \right) \sqrt{d\log(L_\phi)K} + H^2\sqrt{d\mathcal{I}K} \right).$$

If, as assumed in the original formulation of MDPs with Low Inherent Bellman Error, $\mathcal{R}_h \leq \sqrt{d}$, and $L_\phi$ is polynomial in $d$, the bound reduces to the usual $H^2 d\sqrt{K} + H^2\sqrt{d\mathcal{I}K}$ rate.

In the next paragraph, we are going to show that this result allows to get a non-trivial regret bound for Weakly Smooth MDPs.

**Applying theorem 44 to the Weakly Smooth class** First of all, we see that the condition $\sup_{h,\theta\in\mathcal{B}_h} \|Q_h[\theta]\|_\infty \leq 2H$ is guaranteed by equation (6.1), regardless of (6.2). Next, it is possible to find an upper bound for $\mathcal{R}_h$. By definition, the Legendre feature map $\varphi_N$ that we have built is orthonormal on $\mathcal{Z}$. Thus, by Parseval's theorem

$$\begin{aligned}\sup_{\theta\in\mathcal{B}_h} \|\theta\|_2^2 &= \sup_{\theta\in\mathcal{B}_h} \|\varphi_N(\cdot)^\top\theta\|_{L^2} \\ &\leq \sup_{\theta\in\mathcal{B}_h} \sqrt{\mathrm{Vol}(\mathcal{Z})}\|\varphi_N(\cdot)^\top\theta\|_{L^\infty} \\ &\leq 2^{m/2} \sup_{\theta\in\mathcal{B}_h} \|\varphi_N(\cdot)^\top\theta\|_{L^\infty} \leq 2^{m/2+1}H,\end{aligned}$$

Where the last step is due to the fact that, by definition, $\mathcal{B}_h$ only contains parameters corresponding to linear function that are bounded by $2H$. It is then necessary to ensure that $\log(L_\phi)$ is adsorbed in the $\widetilde{\mathcal{O}}$ notation. We have already proved that $L_\phi = \mathcal{O}(d^{3/2})$, so its contribution gets lost in the $\widetilde{\mathcal{O}}$. All the assumptions of theorem 44 are now satisfied. Its thesis ensures that

$$\mathrm{RegretBound}(\textsc{Eleanor}|\mathscr{M}_{\mathrm{WS}}) \leq \widetilde{\mathcal{O}}\left(H^2 d\sqrt{K} + H^2\sqrt{d\mathcal{I}K}\right). \tag{6.4}$$

This result matches the exact regret bound proved in [151], up to logarithmic terms.

### 6.1.2. Regret Upper Bound

Given a choice for $N$, eq. (6.4) allows to derive an upper bound for the algorithm Eleanor when applied to Weakly Smooth MDPs with the Legendre feature map $\varphi_N$ and the choice of $\mathcal{B}_h$ done in equations 6.1 and 6.2. In fact, from the discussion made so far, we know that the misspecification does not exceed

$$\mathcal{I} \leq \widetilde{\mathcal{O}}\left(\frac{(\mathrm{J_P C}_{\mathcal{T}}^{\mathrm{WS}})^H}{N^\nu}\right).$$

When fixing $N = \Theta(K^\alpha)$, it follows that $d = \Theta(K^{m\alpha})$ and $\mathcal{I} = \mathcal{O}(\exp(H)N^{-\alpha\nu})$. Imposing that all the leading terms of eq. (6.4) have the same magnitude in $K$, the parameter $\alpha$ must satisfy

$$\frac{1}{2} + m\alpha = \frac{m\alpha}{2} - \nu\alpha + 1 \implies \alpha = \frac{1}{m+2\nu}.$$

Replacing in the regret bound, the result is

$$R_K \leq \widetilde{\mathcal{O}}\left(\exp(H) K^{\frac{3m/2+\nu}{m+2\nu}}\right).$$

With the former passages, the main theorem of [79] is proved:

**Theorem 45** (Theorem 1 from [79]). *Assume online interaction and assumptions 6 and 7. For the feature map $\varphi_N$ (with $N = \Theta(K^{\frac{1}{m+2\nu}})$) and $\mathcal{B}_h$ given by 6.1, the* ELEANOR *algorithm enjoys the following regret bound on Weakly Smooth MDPs.*

$$\boxed{\textit{RegretBound}(\textsc{Eleanor}|\mathscr{M}_{WS}) \leq \widetilde{\mathcal{O}}\left(C^H K^{\frac{3m/2+\nu}{m+2\nu}}\right)},$$

*for some constant $C$.*

As expected given the lower bound, the regret scales exponentially with respect to the time horizon. However, the most striking aspect is the exponent of $K$, which is $\frac{3m/2+\nu}{m+2\nu}$. This value is strictly greater than the lower bound of $\frac{m+\nu}{m+2\nu}$. Such a growth rate brings a problematic feature: the regret bound it is not always sublinear. On the contrary, the no-regret property holds only if $2\nu > m$. Above the critical threshold of $2\nu = m$, the result of Theorem theorem 45 becomes vacuous, as the regret exceeds $K$.

This is not the first time such a phenomenon has been observed for Smooth MDP classes. On the contrary, the regret bound of the GOLF algorithm, which we established in eq. (4.24), scales as $K^{\frac{3m+2\nu}{4\nu}}$. As noted in the discussion of that result, this bound is acceptable only if $3m < 2\nu$, a significantly more stringent condition than the one required by theorem 45. In this sense, our latest theorem extends the region where Weakly Smooth MDPs admit a non-vacuous regret bound to include the parameter space $\{(m, \nu) : 2\nu/3 \leq m < 2\nu\}$, which was previously uncovered. In fig. 6.1, we plot the regions of parameters where both algorithm fail to ensure the no-regret guarantee.

Like in the former chapter, after proving the regret guarantee for the Weakly Smooth case, the most general class that we consider, we turn our attention on the Strongly and Mildly Smooth processes. Is there a way to improve the result of theorem 45 under these stronger assumptions? The answer is yes. Indeed, by a simple change in the definition of $\mathcal{B}_h$ (see eq. (6.1)) is enough to avoid the $\exp(H)$ blow-up: all that is needed is to use the norm bound for Mildly Smooth MDPs, instead of the one of the Weakly Smooth class. Still, in the following section we show a different approach, that aims to achieve a regret bound that is better than the one of ELEANOR, also under other aspects.

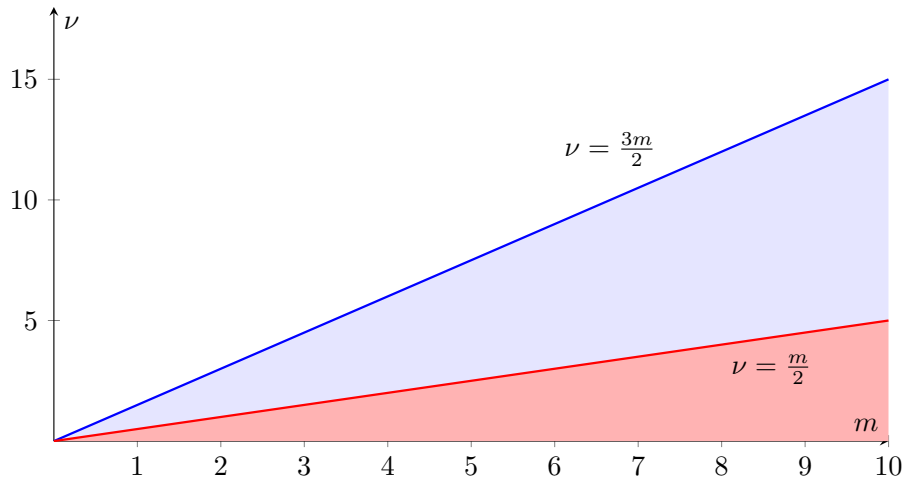



Figure 6.1: Regions of parameters $(m, \nu)$ where GOLF and ELEANOR **fail** to enjoy the no-regret property in a Weakly Smooth MDP. Red: region where the regret of both algorithms is superlinear, Blue: region where only the regret GOLF is superlinear.

## 6.2. Improving the Regret Upper Bound for Mildly Smooth MDPs

In the former section, we demonstrated how modifying the analysis of the ELEANOR algorithm leads to a regret bound for Smooth MDPs. In this one, we aim for a more radical approach: modifying the core definition of MDPs with low inherent Bellman error (definition 10) to establish an even more general abstract class. Once this framework is established, the regret bound for Mildly Smooth MDPs will follow directly by appropriately setting the class parameters.

### 6.2.1. Locally Linearizable MDPs

MDPs with low inherent Bellman error were conceived to model scenarios where, given a feature map $\phi_h : \mathcal{Z} \to \mathbb{R}^d$, each optimal Q-function $Q_h^*(z)$ can be approximated by $\phi_h(z)^\top \theta_h$ for some $\theta_h \in \mathbb{R}^d$. This is complemented by a "Bellman closure" assumption, which holds up to a constant $\mathcal{I} > 0$. From our perspective, the "weak point" of this formulation is that the $\theta_h$ value *has to be the same for all the domain* $\mathcal{Z}$. One can still enlarge $\phi_h$ with more feature to distinguish different regions of $\mathcal{Z}$, adding for example indicator functions. Unfortunately, enlarging the feature map in this way, has tremendous impact on the regret, which scales as $\sqrt{d\mathcal{I}K}$. In fact, if one increases $d$ too much, the regret bound scales linearly in $K$.

To overcome this limitation, we introduce a novel concept of *locality* to enable the feature map to be associated with different parameters $\theta$ in different regions of the state-action space. While definition 10 concerned two element, an MDP and a feature map, we build our class on three elements. The former two, plus a partition of $\mathcal{Z}$ into regions. Precisely, given a Markov Decision Process $M$, define the following two entities:

1. $\phi_h : \mathcal{Z} \to \mathbb{R}^d$: a feature map which is allowed to depend on the current stage.
2. $\mathcal{U}_h$: a sequence of partitions of the state-action space $\mathcal{Z}$ in $N_h$ regions, so that $\mathcal{U}_h := \{\mathcal{Z}_{h,n}\}_{n=1}^{N_h}$. We call $\rho_h : \mathcal{Z} \to [N_h]$ the map linking every element $z \in \mathcal{Z}$ to the index of its set in the partition $\mathcal{U}_h$.

Calling $\boldsymbol{\theta}_h = \{\theta_{h,n}\}_{n=1}^{N_h}$ a list of vectors in $\mathbb{R}^d$, one for each of the regions $\mathcal{Z}_{h,n}$, employ, as candidates for the state-action value function, the following set

$$\mathcal{Q}_h := \left\{ Q_h[\boldsymbol{\theta}_h](\cdot) = \phi_h(\cdot)^\top \theta_{h,\rho_h(\cdot)}, \quad \text{where} \quad \boldsymbol{\theta}_h = \{\theta_{h,n}\}_{n=1}^{N_h},\ \theta_{h,n} \in \mathcal{B}_{h,n} \right\}. \tag{6.5}$$

Coherently, define $\mathcal{V}_h := \{V(s) = \sup_{a\in\mathcal{A}} Q(s,a) : \ Q \in \mathcal{Q}_h\}$. For fixed $z$, we have $Q_h[\boldsymbol{\theta}_h](z) = \phi_h(z)^\top \theta_{h,\rho_h(z)}$, so that the feature map is allowed to depend directly on $z$, while the linear parameter only depends on $\rho_h(z)$, the function indicating in which of the $N_h$ regions we are. $\mathcal{B}_{h,n}$ are arbitrary compact sets which contain the candidate values for the linear parameters $\theta_{h,n}$. Two relevant quantities for this model are the $2-$norm of the feature map and the diameter of the sets $\mathcal{B}_{h,n}$,

$$L_\phi := \sup_{h\in[H], z\in\mathcal{Z}} \|\phi_h(z)\|_2 \qquad \mathcal{R}_{h,n} := \operatorname{diam}(\mathcal{B}_{h,n}).$$

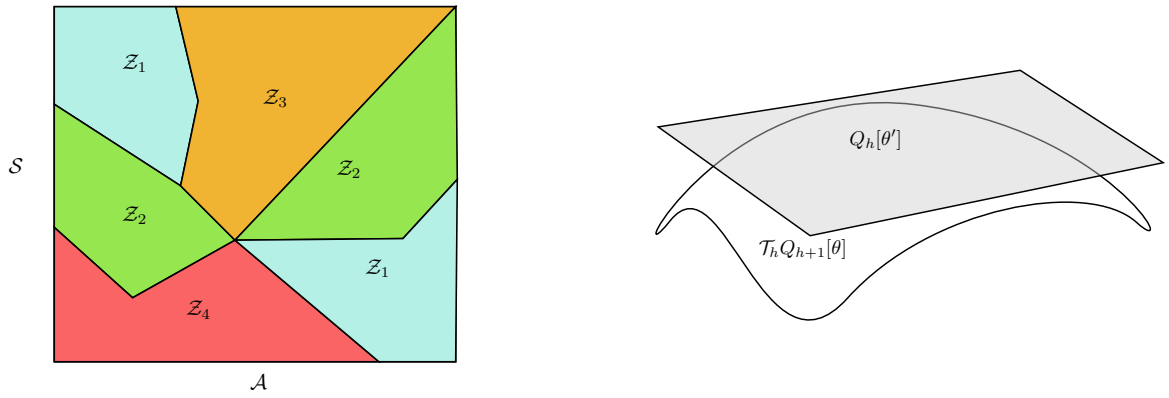
The low inherent Bellman error property can be redefined in this setting as follows.

**Definition 25.** *Given a family of compact sets $\mathcal{B}_{h,n} \subset \mathbb{R}^d$ depending on $h \in [H], n \in [N_h]$, and their Cartesian product $\mathcal{B}_h = \bigtimes_{n=1}^{N_h} \mathcal{B}_{h,n}$, define:*

$$\mathcal{I}(\phi,\mathcal{U}) := \max_{h\in[H]} \sup_{\boldsymbol{\theta}_{h+1}\in\mathcal{B}_{h+1}} \inf_{\boldsymbol{\theta}_h\in\mathcal{B}_h} \|Q_h[\boldsymbol{\theta}_h](\cdot) - \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}](\cdot)\|_{L^\infty}. \tag{6.6}$$

For $N_h = 1$, this definition exactly reduces to definition 10, as expected. As in that case, the term $\mathcal{I}$ plays the role of an approximation error. By assuming a bound on $\mathcal{I}(\phi,\mathcal{U})$, it is possible to give a formal definition of the class of MDPs we study in this section.

**Definition 26** (definition 2 in [78]). *A Locally Linearizable MDP is a triple $(\mathcal{M}, \{\phi_h\}_{h=1}^H, \{\mathcal{U}_h\}_{h=1}^H)$ where $\mathcal{M}$ is an MDP, $\phi_h : \mathcal{Z} \to \mathbb{R}^d$ is a feature map and $\mathcal{U}_h$ a sequence of*




(a) (b)

Figure 6.2: The space $\mathcal{Z}$ is partitioned into several regions (a), which do not need to be convex nor connected. On each of these regions, as shown in (b), the result of the Bellman optimality operator can be well approximated by a $Q$ function that is linear in the feature map, with a parameter $\theta$ that may depend on the region itself.

*partitions of the state-action space $\mathcal{Z}$ in $N_h$ regions such that the corresponding inherent Bellman error (definition 25) satisfies $\mathcal{I}(\phi, \mathcal{U}) \leq \mathcal{I}$.*

As for the class of MDPs with Low Inherent Bellmann error, $\phi$ (and $\mathcal{U}$ in our case) must be known to the agent, while $\mathcal{I}$ is not needed. The novel aspect of this class is that, as a result of definition 25, linearity is independently enforced in separate regions of the state-action space. We illustrate this concept in Figure 6.2.

While it is trivial to see that every MDP with low-inherent Bellman error is Locally Linearizable, the vice-versa also holds, if we multiply each element of the original feature map by the indicator function of each region. However, given the regret bound for ELEANOR, this would make the regret bound scale with $\sum_{h=1}^{H} \sqrt{N_h d \mathcal{I} K}$, which we want to avoid.

## CINDERELLA: an Algorithm for Locally Linearizable MDPs

To prove a better regret bound, we propose algorithm 6.1, which is specific to locally linearizable MDPs (algorithm 1 in [78]). Call $z_h^k := (s_h^k, a_h^k)$ and $\phi_h^k = \phi_h(z_h^k)$. At every episode $k \in [K]$, algorithm 6.1 defines an optimistic estimation of the $Q$-function for every step $h \in [H]$. Then, it chooses actions in order to maximize this function while fixing $s_1^k$ (line 7). To compute $\overline{Q}_h$, CINDERELLA relies on solving an optimization problem, which follows an idea similar to the one of [151]. The goal is to optimize over three sets of variables: $\widehat{\theta}_{h,n}, \overline{\xi}_{h,n}, \overline{\theta}_{h,n}$, the first one representing ridge regression of the linear parameter for region $n$ at step $h$, the second representing the uncertainty relative to this estimation,

Algorithm 6.1 CINDERELLA (Constrained INDEpendent REgressions with Local Linear Approximations)

**Require:** Failure probability $\delta$, Regularization $\lambda$, Region mapping $\rho_h$, Feature mapping $\phi_h$
Initialize $\Lambda^1_{h,n} := \lambda I$ for every $h \in [H], n \in [N_h]$
**for** $k = 1, 2, \dots K$ **do**
  Receive initial state $s^k_1$
  Solve optimization program in (6.7) obtaining $\overline{\theta}^k_{h,n}$ for every $h \in [H]$ and $n \in [N_h]$
  $Q_h[\overline{\boldsymbol{\theta}}^k_h](z) \leftarrow \phi_h(z)^\top \overline{\theta}^k_{h,\rho_h(z)} \quad \forall h \in [H]$
  **for** $h = 1, 2, \dots H-1$ **do**
    Choose action $a^k_h \in \arg\max_{a \in \mathcal{A}} Q_h[\overline{\boldsymbol{\theta}}^k_h](s^k_h, a)$
    Receive reward $r^k_h$ and next state $s^k_{h+1}$
  **end for**
**end for**

and the third one an "optimistic" estimate.

Under this view, the objective is to maximize the surrogate $V$-function in the first state, and the constraints are designed so that all variables match their intuitive interpretation. Formally, the optimization problem is defined as:

Global Optimization of Region Parameters: (6.7)

$$\max_{\widehat{\theta}_{h,n}, \overline{\xi}_{h,n}, \overline{\theta}_{h,n}} \max_{a \in \mathcal{A}} \phi_1(s^k_1, a)^\top \overline{\theta}_{1,\rho_1(s^k_1,a)}$$

$$\text{s.t.} \quad \widehat{\theta}_{h,n} = {\Lambda^k_{h,n}}^{-1} \sum_{\tau=1}^{k-1} \mathbf{1}\{\rho_h(z^\tau_h) = n\} \phi^\tau_h \left( r^\tau_h + \max_{a \in \mathcal{A}} \phi_{h+1}(s^\tau_{h+1}, a)^\top \overline{\theta}_{h+1,\rho(s^\tau_{h+1},a)} \right)$$

$$\overline{\theta}_{h,n} = \widehat{\theta}_{h,n} + \overline{\xi}_{h,n}$$

$$\|\overline{\xi}_{h,n}\|_{\Lambda^k_{h,n}} \leq \sqrt{\alpha^k_{h,n}}.$$

Where $\Lambda^k_{h,n} := \sum_{\tau=1}^k \mathbf{1}\{\rho_h(z^\tau_h) = n\} \phi^\tau_h {\phi^\tau_h}^\top + \lambda I$, is the design matrix of $\lambda-$regularized ridge regression, and $\alpha^k_{h,n}$ is a constant determining the exploration rate which will be fixed in the analysis (see the appendix E.3). The first constraint enforces that $\widehat{\theta}_{h,n}$ is estimated with ridge regression having as target the (optimistic) value function estimated for step $h+1$, and the second constrain $\overline{\theta}_{h,n} = \widehat{\theta}_{h,n} + \overline{\xi}_{h,n}$ ensures that the optimistic estimate in every region is given by its mean estimate plus the uncertainty, while the third one bounds the magnitude of $\overline{\xi}_{h,n}$ so that this uncertainty shrinks the more data we collect.

Although the structure of the algorithm is directly inherited from ELEANOR, CINDERELLA distinguishes itself by dividing the samples across the various regions of the state-action space of the MDP. This allows parameters associated with different regions to be learned

independently. Despite introducing another layer of technical difficulty in proving the regret bound, this procedure is relatively natural given the characteristics of our class of problems.

As the derivation of the regret bound for this algorithm is lengthy, the complete proof is relegated to the appendix E.3. Below, we outline the most significant steps and insights.

The first important step of the proof is to define an "approximately optimal" solution for the MDP. Define $\boldsymbol{\theta}_h^\star$ in the following recursive way

$$\theta_{h,n}^\star := \arg\min_{\theta_n \in \mathcal{B}_{h,n}} \max_{z \in \mathcal{Z}_{h,n}} |\phi_h(z)^\top \theta_n - \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}^\star](z)|.$$

So that $\boldsymbol{\theta}_h^\star$ is constructed by stacking these $\theta_{h,n}^\star$ terms. This parameters are able to well approximate the $Q_h^*$ function.

**Theorem 46.** *The approximately optimal $Q-$function satisfies*

$$\forall h = 1, \dots, H \qquad \|Q[\boldsymbol{\theta}_h^\star](\cdot) - Q_h^\star(\cdot)\|_{L^\infty} \leq (H-h)\mathcal{I}.$$

This result allows to ensure that in our parameter space a function exists that can approximate the $Q_h^*$ function sufficiently well. At that point, most of the proof focuses on showing that such a parameter $\{\boldsymbol{\theta}_h^\star\}_{h=1}^H$ always belong to the feasible region of eq. (6.7). This is done in theorem 57. The choice of $\sqrt{\alpha_{h,n}^k}$ that makes it possible is then split in different parts and bounded as, roughly,

$$\alpha_{h,n}^k = \mathcal{O}\left(p_{h,n}^k \mathcal{I}^2 + (H+\sigma)^2 d + (H+\sigma)^2 \log(\delta^{-1}\mathcal{N}(1/\sqrt{k}), \mathcal{V}_h)) + \lambda^{-1}\mathcal{R}_{h,n}\right),$$

where

$$p_{h,n}^k := \sum_{\tau=1}^{k-1} \mathbf{1}\{z_h^\tau \in \mathcal{Z}_{h,n}\}.$$

Apart from optimism, the other ingredient to prove the regret bound is proposition 59, which ensures that with probability at least $1-\delta$ the $Q-$function computed with the optimization algorithm at each step $k$ satisfies,

$$\forall z \in \mathcal{Z} \qquad \left|\overline{Q}_h^k(z) - \mathcal{T}_h[\overline{Q}_{h+1}^k](z)\right| \leq \mathcal{I} + 2\sqrt{\alpha_{h,\rho_h(z)}^k}\|\phi_h(z)\|_{{\Lambda_{h,\rho_h(z)}^k}^{-1}}.$$

At this point, the fact that both the approximately optimal $Q[\boldsymbol{\theta}_h^\star]$ and the function $\overline{Q}_h^k$, computed by our algorithm, have low Bellman error, allows to prove the final regret

bound. The regret is proved to be bounded with

$$R_K \leq KH\mathcal{I} + \sum_{k=1}^{K}\sum_{h=1}^{H} 2\sqrt{\alpha^k_{h,\rho_h(z^k_h)}}\|\phi_h(z^k_h)\|_{{\Lambda^k_{h,\rho_h(z^k_h)}}^{-1}} + \zeta^k_h,$$

where $\zeta^k_h$ is a negligible martingale term. The final regret bound (reported in the appendix as corollary 61) follows from applying the pigeonhole argument (on the regions $\mathcal{Z}_{h,n}$) to $\sqrt{\alpha^k_{h,\rho_h(z^k_h)}}\|\phi_h(z^k_h)\|_{{\Lambda^k_{h,\rho_h(z^k_h)}}^{-1}}$, and writes as follows.

**Theorem 47.** *Let $\mathscr{M}$ the class of MDPs satisfying definition 26. Under online interaction, assumptions 6 and 7 and $\sup_{h,\boldsymbol{\theta}_h\in\mathcal{B}_h}\|Q_h[\boldsymbol{\theta}_h]\|_{L^\infty} \leq 2H$,*

$$\textit{RegretBound}(\textsc{Cinderella}|\mathscr{M}) \leq \widetilde{\mathcal{O}}\left(\sum_{h=1}^{H} HN_h(\sqrt{d}+\mathcal{R}_h)\sqrt{d\log(L_\phi)K} + H\sqrt{d\mathcal{I}K}\right),$$

*where $\mathcal{R}_h := \max_{n\in[N_h]}\mathcal{R}_{h,n}$.*

Theorem 47 provides a bound on regret that is much tighter than the one which would result from Eleanor. In the previous result, ignoring the terms that are not related to the misspecification, the regret scales as $Nd\sqrt{K} + \sqrt{d\mathcal{I}K}$. Therefore, it allows to avoid having $N$ in the second term, which contains the misspecification error.

## Applying Theorem 47 to Mildly Smooth MDPs

Having proved that our Cinderella algorithm enjoys an improved regret bound on Locally Linearizable MDPs, we need to see how to reduce Mildly Smooth MDPs to this class.

The key idea is to leverage the Taylor feature map introduced in appendix B.2. This map offers a significant advantage over Fourier features or Legendre polynomials: it employs distinct parameters for different regions of the space $\mathcal{Z}$. We exploit this property to construct a partitioning of $\mathcal{Z}$ that minimizes the number of parameters required for each region. As in eq. (4.20), we start fixing $\varepsilon > 0$, then build a $\varepsilon-$net $\mathcal{Z}^\varepsilon$ in the infinity norm, whose size can be proved to be $|\mathcal{Z}^\varepsilon| =: N \leq (2/\varepsilon)^m$. Now, for every $z^n \in \mathcal{Z}^\varepsilon$, we define recursively $\mathcal{Z}_n$ to be the set of points which are near to $z^n$, formally:

$$\mathcal{Z}_1 := \{z \in \mathcal{Z} : \|z - z^1\|_\infty \leq \varepsilon\}, \qquad \mathcal{Z}_n := \{z \in \mathcal{Z} : \|z - z^n\|_\infty \leq \varepsilon\} \setminus \cup_{\ell=1}^{n-1}\mathcal{Z}_\ell. \tag{6.8}$$

By definition, every point of $\mathcal{Z}$ is matched with a point $z^n$ of the cover $\mathcal{Z}^\varepsilon$ and, importantly,

is assigned to exactly one subset $\mathcal{Z}_n$ of $\mathcal{Z}$. The feature map is simply defined as

$$\phi_h(z) = H \cdot \psi^{\nu_*}_{z_n}(z) \qquad n = \rho(z), \tag{6.9}$$

where $\rho$ is the function that assigns points to the corresponding region, and $\psi^{\nu_*}_y$ is the Taylor feature map of degree $\nu_*$, whose length corresponds to $d_{\nu_*} = \binom{\nu_*+m}{\nu_*} \leq \min\{m^{\nu_*}, \nu_*^m\}$. Under this choice, we can map Mildly Smooth MDPs to Locally Linearizable ones.

**Theorem 48.** *Let $\mathcal{M}$ be a Mildly Smooth MDP, and $\varepsilon \leq 1/(2C_{\mathcal{T}}H)^{1/\nu}$. Then, choosing $\mathcal{U}_h = \{\mathcal{Z}_n\}_n^{N_h}$ as in (6.8) and taking $\phi_h$ as in (6.9)), the triple $(\mathcal{M}, \mathcal{U}, \phi)$ is a Locally Linearizable MDP, where $\mathcal{B}_{h,n}$ may be chosen in a way that*

$$L_\phi = (1 + 2\sqrt{d}_{\nu_*})H, \qquad \mathcal{R}_{h,n} = 2\sqrt{d}_{\nu_*}C_{\mathcal{T}}, \qquad \mathcal{I} \leq 2C_{\mathcal{T}}^{MS}\varepsilon^{\nu},$$

*and* $\sup_{h,\boldsymbol{\theta}_h \in \mathcal{B}_h} \|Q_h[\boldsymbol{\theta}_h]\|_{L^\infty} \leq 2H$.

The proof of this result follows from the basic property of Taylor polynomials, namely that the approximation error decays as $\varepsilon^\nu$, for a detailed explanation of the other constants, see appendix E.3.5.

Using theorem 48 allows to apply the regret guarantee for CINDERELLA to the Mildly Smooth class.

**Theorem 49** (Theorem 4 in [78])**.** *Assume online interaction and assumptions 6 and 7. Choosing feature map and regions according to theorem 47 (with $\varepsilon = \Theta(K^{\frac{1}{2m+2\nu}})$),*

$$\boxed{\textit{RegretBound}(\textsc{Cinderella}|\mathscr{M}_{MS}) \leq \widetilde{\mathcal{O}}\left(H^{\frac{2\nu+m}{\nu+m}}K^{\frac{2m+\nu}{2m+2\nu}} + H^{\frac{5\nu+2m}{\nu}}\right)}.$$

*Proof.* Let $\mathcal{M} \in \mathscr{M}_{\text{MS}}$, and fix $\varepsilon > 0$. We apply theorem 48 to map $\mathcal{M}$, and call $R_K$ the regret of CINDERELLA, for parameters determined by theorem 48 for $\varepsilon$, ran on $\mathcal{M}$. Since the number of regions is constant $N_h = N$, theorem 47 yields

$$R_K \leq \widetilde{\mathcal{O}}\left(H^2 N d\sqrt{K} + H\sqrt{d}\mathcal{I}K\right). \tag{6.10}$$

The values in the regret are linked to $\varepsilon$ as follows:

$$N = \Theta(\varepsilon^{-m}) \qquad \mathcal{I} = \mathcal{O}(\varepsilon^{\nu}).$$

Letting $\varepsilon = H^\beta K^\alpha$, and equalizing the magnitude of the two terms, results in

$$2 - m\beta = 1 + \nu\beta \implies \beta = \frac{1}{\nu + m},$$

$$\frac{1}{2} - m\alpha = 1 + \nu\alpha \implies \alpha = -\frac{1}{2\nu + 2m}.$$

Instead, $d$ does not depend on $\varepsilon$: the length of the Taylor feature map only depends on $m, \nu_*$. Replacing in the previous results in the following regret bound.

- if $\varepsilon \leq 1/(2C_{\mathcal{T}}H)^{1/\nu}$, by (6.10)

$$R_K \leq \widetilde{\mathcal{O}}\left(H^{\frac{2\nu+m}{\nu+m}} K^{\frac{2m+\nu}{2m+2\nu}}\right).$$

- If $\varepsilon > 1/(2C_{\mathcal{T}}H)^{1/\nu}$, one has

$$\begin{aligned} K &= H^2 \varepsilon^{-2(\nu+m)} \\ &\leq H^2 \left((2C_{\mathcal{T}}H)^{1/\nu}\right)^{2(\nu+m)} \\ &\lesssim H^2 H^{\frac{2(\nu+m)}{\nu}} = H^{\frac{4\nu+2m}{\nu}}. \end{aligned}$$

  Also, the regret is bounded by $HK$, as it is always. Therefore,

$$R_K \leq HK \lesssim H^{\frac{5\nu+2m}{\nu}}.$$

Making the sum of the two cases ends the proof. □

While this section originally aimed to establish a regret bound that could overcome the exponential dependence on $H$ observed in theorem 45, the resulting analysis addresses an even more fundamental issue: the elimination of any superlinear term. The bound derived in theorem 49 scales sublinearly for any choice of the parameters $m$ and $\nu$. This effectively resolves the concerns raised in the previous section regarding parameter regions where the no-regret property was not guaranteed (see fig. 6.1). Remarkably, our findings suggest that the distinction between Weakly Smooth and Mildly Smooth classes extends beyond the transition from exponential to polynomial dependence on $H$. We have demonstrated that the growth rate of the regret itself improves significantly, moving from $K^{\frac{3m/2+\nu}{m+2\nu}}$ in the Weakly Smooth case to $K^{\frac{2m+\nu}{2m+2\nu}}$ in the Mildly Smooth setting.

### Why the Same Strategy Cannot Work for Weakly Smooth MDPs

The previous section shows an approach that, at the cost of generalizing the concept of MDPs with Low Inherent Bellmann error, manages to derive a bound on the regret that grows sublinearly with $K$, for every pair of parameters $\nu, m$ for Mildly Smooth MDPs. The same thing was not achieved in the Weakly Smooth case, where our regret bound grows superlinearly for $\nu < m/2$. Generalizing the strategy based on Locally Linearizable MDPs to the Weakly Smooth case would be of great interest, as it would imply convergence to the value of the optimal policy for the broadest class of processes we are studying in this thesis.

Unfortunately, this strategy cannot work, for the following reason. As theorem 48 shows, the strategy for mapping Mildly Smooth processes to Locally Linearizable MDPs is intrinsically tied to the use of a local feature map, such as the Taylor map. These types of maps, which use different and independent values of $\theta_n$ for each region $\mathcal{Z}_n$, are forced to produce a discontinuous approximator, which changes abruptly when the region changes. As long as one is working within the context of Mildly Smooth MDPs, this is not a problem: their assumption can be written as eq. (4.13):

$$\|\mathcal{T}_h\|^{\circ}_{L^\infty \to \mathrm{W}^{\nu,\infty}} \leq \mathrm{C}^{\mathrm{MS}}_{\mathcal{T}}.$$

Since the Bellman operator acts on the infinite norm of the starting function, it is sufficient for the feature map-based function approximator to produce bounded functions. In this context, using a smooth approximation is redundant. In contrast, the assumption for the Weakly Smooth case takes the following form:

$$\|\mathcal{T}_h\|^{\circ}_{\mathrm{W}^{\nu,\infty} \to \mathrm{W}^{\nu,\infty}} \leq \mathrm{C}^{\mathrm{WS}}_{\mathcal{T}}.$$

In this case, the operator assumption holds as long as the starting function has bounded norm $\|\cdot\|_{\mathrm{W}^{\nu,\infty}}$. For a local function approximator, where $\widehat{Q}_h$ is discontinuous, the previous equation provides no guarantee on the result of the Bellman optimality operator. As already intuited in the previous chapter (see for example fig. 5.2), this phenomenon forces us to use feature maps such as Fourier maps or Legendre polynomials, in order to ensure that $\widehat{Q}_h(z) = \varphi(z)^\top \widehat{\theta}_h$ is always smooth.

## 6.3. A Computationally Efficient Algorithm

In this section we move our attention away from the regret bounds to try to solve a different problem: developing an algorithm for Smooth MDPs that is computationally

Algorithm 6.2 Least-Squares Value Iteration with UCB (LSVI-UCB)

**Require:** Regularization $\lambda$, Optimism parameter $\beta$, Feature map $\phi$

1: **for** episode $k = 1, \dots, K$ **do**
2: Receive the initial state $s_1^k$
3: **for** step $h = H, \dots, 1$ **do**
4: $\Lambda_h \leftarrow \sum_{\tau=1}^{k-1} \phi(x_h^\tau, a_h^\tau)\phi(s_h^\tau, a_h^\tau)^\top + \lambda \cdot I$
5: $\theta_h \leftarrow \Lambda_h^{-1} \sum_{\tau=1}^{k-1} \phi(s_h^\tau, a_h^\tau)\left[r_h(s_h^\tau, a_h^\tau) + \max_a Q_{h+1}(s_{h+1}^\tau, a)\right]$
6: $Q_h(\cdot,\cdot) \leftarrow \min\left\{\theta_h^\top \phi(\cdot,\cdot) + \beta\sqrt{\phi(\cdot,\cdot)^\top \Lambda_h^{-1}\phi(\cdot,\cdot)},\ H\right\}$
7: **end for**
8: **for** step $h = 1, \dots, H$ **do**
9: Take action $a_h^k \leftarrow \arg\max_{a\in\mathcal{A}} Q_h(s_h^k, a)$ and observe $s_{h+1}^k$
10: **end for**
11: **end for**

efficient.

In fact, the intractability of ELEANOR is a well-known problem in the literature of RL (for this topic in detail, see [149]). Our CINDERELLA, which follows the same structure, inherits the same problem. For this reason, we will dedicate this section to finding an algorithm that has a complexity that scales polynomially across the most relevant variables.

The core intuition is straightforward: by restricting our focus to the Strongly Smooth class, we can employ the LSVI-UCB algorithm (see algorithm 6.2) with a specifically chosen feature map. Since theorem 19 provides a regret bound even in the presence of misspecification error, the problem reduces to selecting a feature map capable of approximating smooth functions at the optimal order. Following the approach in section 6.1, we utilize Legendre polynomials as features, denoting the normalized Legendre polynomial feature map of degree $N$ on $\mathcal{Z}$ as $\varphi_N$.

**Theorem 50** (Theorem 2 from [79]). *Assume online interaction and assumptions 6 and 7. For the Legendre polynomial feature map $\varphi_N$ (with $N = \Theta(K^{\frac{1}{m+2\nu}})$),*

$$\textit{RegretBound}(\textit{LSVI-UCB}|\mathscr{M}_{SS}) \leq \widetilde{\mathcal{O}}\left(H^2 K^{\frac{2m+\nu}{m+2\nu}}\right).$$

While the regret bound presented here may seem less exciting, given that its order is worse to that of theorem 49 despite a more restricted problem class, the primary merit of this approach lies in its computational efficiency, which we analyze in the next lines.

At each episode $k$ and stage $h$, line 4 involves computing and inverting a $d\times d$ design matrix $\Lambda_h$. By employing incremental updates via the Sherman-Morrison formula, this operation

is reduced to a complexity of $\mathcal{O}(d^2)$ per $(k,h)$ pair. However, the subsequent step in line 5 necessitates a summation over past episodes that cannot be performed incrementally, as the term $\max_a Q_{h+1}(s_{h+1}^\tau, a)$ depends on the current $Q_{h+1}$ function. Consequently, its complexity scales as $\mathcal{O}(Kd^2)$ per $(k,h)$ pair, since evaluating the $Q_h$ function, which involves a quadratic form, requires at least $\mathcal{O}(d^2)$ for each input. Finally, assuming that the maximization and truncation in line 6 are efficiently handled through an appropriate cover of the action space, the overall complexity remains

$$\text{Computational Comp.(LSVI-UCB)} = \mathcal{O}\left(\sum_{k=1}^{K}\sum_{h=1}^{H} d^2 K\right) = \mathcal{O}\left(HK^2d^2\right).$$

Taking $N = \Theta(K^{\frac{1}{m+2\nu}})$, that achieves the regret bound, results in $d = \Theta(N^m) = \Theta(K^{\frac{m}{m+2\nu}})$. Substituting this value, we get the following computational complexity for Strongly Smooth MDPs.

$$\text{Computational Comp.(LSVI-UCB}|\mathscr{M}_{\text{SS}}) = \mathcal{O}\left(HK^{\frac{4m+2\nu}{m+2\nu}}\right).$$

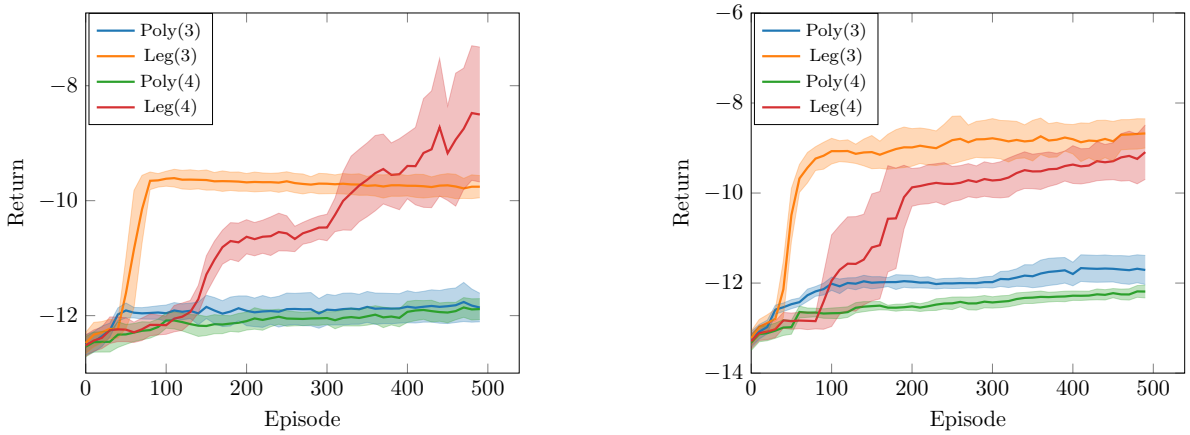
The computational complexity is therefore always polynomial, although it may grow up to $K^4$ in case $\nu \ll m$.

### 6.3.1. Numerical Simulations

The previous section shows that algorithm 6.2 enjoys both a regret bound and tractable computational complexity. In what follows, we are using it in a numerical simulation to show the usefulness of the Legendre basis with respect to standard polynomials.

As the vector space spanned by the Legendre polynomials is exactly the one spanned by the standard polynomials, it is natural to as what the advantages of this choice are. Even if it is not clear from the regret analysis, the reason behind this choice concerns the well-conditioning of the problem, as well as the orthogonality of Legendre polynomials. Indeed, when using an orthogonal feature map, by Parseval's theorem the $L^2$ norm of a function $f$ corresponds to the Euclidean norm of the vector of its coefficients. This ensures that, for the smooth functions we encounter in the previous theory, this coefficient vector does not grow too much.

In this subsection we show this feature in an empirically way: the use of orthogonal features has beneficial effects on learning performance. While the experimental evaluation is conducted on a simplified environment, such a practice is relatively standard for Reinforcement Learning algorithms characterized by theoretical guarantees.. We employ two




(a) (b)

Figure 6.3: Curves of the episodic return for the simulation in section 6.3.1 with 95% confidence intervals over five random seeds. The four algorithms represent LSVI-UCB for different choices of the feature map. The $x-$axis represent the current episode $k$, the $y-$axis represent the current return at the end of the episode.

modified versions of the LQR, in which the state, after the linear dynamic transition, is pushed towards the origin in a way that prevents it from escaping from a given compact set. Using the same formalism of the LQR, define the following transition function:

$$s_{h+1} = g(As_h + Ba_h + \eta) \qquad \eta \sim \mathcal{N}(0, I),\ g(x) := \frac{x}{1 + \|x\|_2}.$$

As the support of the Gaussian distribution is the full $\mathbb{R}^m$, after applying $g$, the possible set of new states is the ball of radius one. The reward keeps being defined as $-s_h^\top Q s_h - a_h^\top R a_h$ like in the standard LQR problem. We performed two experiments with different parameter values and with horizon $H = 20$. For both environments the dimension of the state space corresponds to 2, and the one of the action space to 1. Also, we have in both cases

$$B = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad Q = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \qquad R = [0.2].$$

What changes is the matrix $A$, which determines most of the dynamics of the system. This matrix, takes the following values

$$\text{Simulation (a): } A = \begin{bmatrix} 0.7 & 0.7 \\ -0.7 & 0.7 \end{bmatrix} \qquad \text{Simulation (b): } A = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}.$$

While polynomial, the algorithm took a very long time to run; in appendix D we report

the details about running time and reproducibility. Figure 6.3, shows the episodic return of four algorithms as a function of the number of learning episodes $k = 1, \dots K$. All the algorithm come from LSVI-UCB, with different choices of the feature map.

1. Poly(N) use as feature map the standard basis of multivariate polynomials (e.g., $\{1, x, y, x^2, y^2, xy, \dots\}$ up to degree $N$. This feature map is not orthogonal on the domain.
2. Leg(N) use as feature map the Legrendre polynomials up to degree $N$, what we used to call $\varphi_N$. This feature map is orthogonal on the domain.

Empirical results demonstrate that baselines incorporating Legendre polynomials significantly outperform their standard polynomial counterparts in terms of cumulative episodic return. While the latter (shown in green and blue) fail to exhibit meaningful learning over $K = 500$ episodes, Leg(3) (orange) achieves rapid policy optimization, subsequently stabilizing at a near-constant return. Conversely, Leg(4) exhibits a slower convergence rate than Leg(3); however, it achieves a superior return in the first environment and comparable performance in the second. These observations align with theoretical expectations: increasing the feature map dimensionality and polynomial degree inherently slows the learning process but enhances approximation accuracy, ultimately leading to higher asymptotic performance.

## 6.4. Summary of the Regret Bounds for Smooth MDPs

To conclude this chapter, it is useful to compare the various regret upper bounds developed for Smooth MDPs. This comparison highlights where we have improved upon existing literature, demonstrated the no-regret property and closed the gap between upper and lower bounds.

The following section details the regret bounds established in this work, organized according to their original derivation. In doing so, we will limit ourselves to considering the order of regret, which contains the dependence on $K$ (which also contains the dependence on $m, \nu$ in its exponent). The first result is eq. (4.24), proving

$$\text{RegretBound}(\text{GOLF}|\mathscr{M}_{\text{WS}}) \leq \widetilde{\mathcal{O}}\left(K^{\frac{3m+2\nu}{4\nu}}\right).$$

The previous result serves primarily as a benchmark: it shows us how the most general RL algorithm with general function approximation behaves on the large class of Weakly Smooth MDPs. Turning to the results demonstrated in this chapter, theorem 45 yields

the following result.

$$\text{RegretBound}(\textsc{Eleanor}|\mathscr{M}_{\text{WS}}) \leq \widetilde{\mathcal{O}}\left(K^{\frac{3m/2+\nu}{m+2\nu}}\right).$$

In this case, even though the algorithm used, Eleanor, was already known, it was necessary to modify its analysis to provide satisfactory theoretical guarantees with the Legendre polynomial map. As we have seen, the previous bound on regret can be further improved by considering the Mildly Smooth family. This produces theorem 49.

$$\text{RegretBound}(\textsc{Cinderella}|\mathscr{M}_{\text{MS}}) \leq \widetilde{\mathcal{O}}\left(K^{\frac{2m+\nu}{2m+2\nu}}\right).$$

This last result is based on a new algorithm, Cinderella, developed together with the Locally Linearizable MDPs class specifically for its applications to continuous RL. The last result on regret comes from theorem 50,

$$\text{RegretBound}(\text{LSVI-UCB}|\mathscr{M}_{\text{SS}}) \leq \widetilde{\mathcal{O}}\left(K^{\frac{2m+\nu}{m+2\nu}}\right).$$

This theorem does not yield a better growth rate than the previous ones, even though the considered MDP family $\mathscr{M}_{\text{SS}}$ is the smallest. However, it works for a computationally efficient algorithm.

To compare these results across different values of $\nu$, we provide a visual representation in fig. 6.4. In order to allow plotting is a single figure, all the curves are shown for $m = 4$, that means having a state dimension $m_1$ and an action dimension $m_2$ that sum up to 4. Using different values of $m$ change the slope of all curves, without significantly changing the message.

As fig. 6.4 shows, there is an unambiguous ranking between the guarantees of the four algorithms that we are taking into consideration. GOLF achieves the worst rate, higher than one unless $\nu > 6$. As already noted, this algorithm requires a Smooth MDP where $\nu \gg m$ in order to achieve satisfactory guarantees. Nonetheless, this algorithm (and all the others) enjoys a regret order that converges to $1/2$ as the smoothness goes to $+\infty$. This rate, corresponding to $\sqrt{K}$ is the same of tabular MDPs for $|\mathcal{S}||\mathcal{A}| \ll K$. Thus, it cannot be improved in general.

LSVI-UCB, which we showed in the previous section to have a computationally efficient algorithm, enjoys a rate that is significantly better than the one of GOLF. In this case, the no-regret property holds for $\nu > 4$. Eleanor performs much better than the former two algorithms. As anticipated, its regret is sublinear if $\nu > m/2$, meaning $\nu > 2$ in this case. The red curve converges to $1/2$ relatively fast.

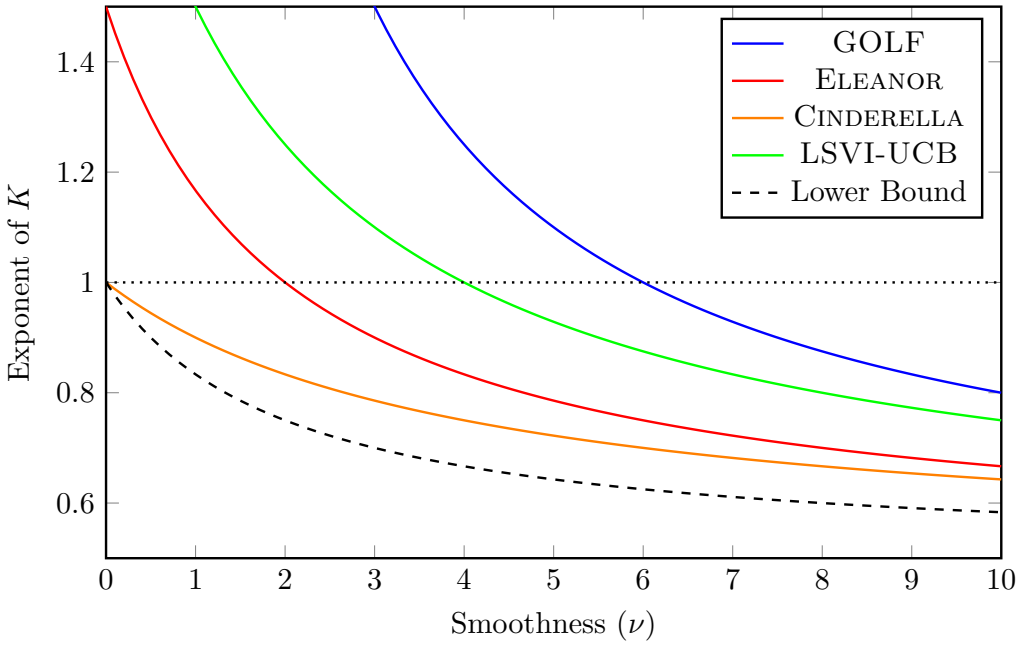



Figure 6.4: Exponent of $K$ in the regret bounds for the various algorithms. Problem dimension is fixed to $m = 4$ and the smoothness parameter $\nu$ is let vary from 0 to 10. Dotted line corresponds to $y = 1$, the requirement to have to no-regret property.

The best algorithm shown in the figure is CINDERELLA. While the difference between its curve, in orange, and that of ELEANOR is quite narrow for large $\nu$, the gap becomes critical for small values of $\nu$. As we can see, CINDERELLA is the only algorithm for which the no-regret property holds over the entire domain, every $\nu > 0$. The fact that this regret guarantee holds even in scenarios with very little smoothness might suggest that it achieves the lower bound. Surprisingly, the figure shows that this is not correct. Conversely, the dashed curve, which corresponds to the lower bound for bandits $\Omega(K^{\frac{m+\nu}{m+2\nu}})$, coincides with the upper bound of CINDERELLA only for $\nu = 0$. For higher values, it converges to $1/2$ more quickly. In fact, the rate $\mathcal{O}(K^{\frac{2m+\nu}{2m+2\nu}})$ from CINDERELLA shows as dependency on $m$ that is twice the one in the lower bound. Clearly, the lower $m$, the lower regret order, in both cases.

Perhaps, this relation is clearer when converting the former regret bounds in sample complexity guarantees, with proposition 9. Indeed,

$$\text{SampleComplexity}(\text{CINDERELLA}|\mathscr{M}_{\text{MS}}, \varepsilon, \delta) \leq \widetilde{\mathcal{O}}\left(\varepsilon^{-2-\frac{2m}{\nu}}\right),$$

which competes with the lower bound,

$$\text{MinMaxComplexity}(\mathscr{M}_{\text{MS}}, \varepsilon, \delta) = \Omega\left(\varepsilon^{-2-\frac{m}{\nu}}\right).$$

In the generative model case, this optimal rate was achieved even for Weakly Smooth MDPs theorem 42. What prevents optimal regret and sample complexity in the case of online interaction? The answer lies in a very specific point in the proof of regret. It is not the approximation properties, which are the same for Taylor polynomials, Legendre polynomials, and the Fourier basis: the problem arises for every the Locally Linearizable MDP. It is the fact that, in the bound on regret, a term appears relating to the log-covering number of the class of all possible state-value functions proposition 54. Without this term, the regret bound of CINDERELLA for Locally Linearizable MDPs would write as

$$R_K \leq \widetilde{\mathcal{O}}\left(d\sqrt{NK} + \sqrt{d\mathcal{I}K}\right).$$

Plugging the parameter of a smooth MDP (see theorem 48) in this *hypothetical* upper bound, the optimal regret bound could be achieved. The problem does not exist under the Generative model assumption, as that setting allowed to estimate the state-action value function with independent samples at any stage $h$, which in rules out this problem. In fact, the need to upper bound the covering number is intrinsically linked to the sequential nature of the interaction: as data comes in trajectories that depend on the current policy, a very intricate dependence between the current estimated value function and the one at the next stage is created. A similar issue used to exists in tabular MDPs 3.3.2: the first regret bounds used to scaled as $S\sqrt{AK}$, instead of the optimal value $\sqrt{SAK}$, for this reason. In that context, different solutions were adopted, which however do not seem to be transferable to the continuous case. Perhaps even more interestingly, the same problem of controlling the covering number of the space of possible state value functions also plagues the world of kernelized MDPs, as we saw in section 4.2.4. The difficulty of reducing the impact of the covering argument on the regret bounds of kernelized MDPs is so challenging that the open problem was officially presented at the prestigious COLT2024 conference (see [133]). Indeed, as proved in section 4.4.1, Kernelized MDPs can be seen as a subclass of our Strongly Smooth MDPs, which allows us to compare the regret results obtained in the two cases. Although neither our algorithms nor those in the literature are able to definitively solve the problem, closing the gap with the lower bound, from this analysis it can be deduced that our CINDERELLA algorithm manages to significantly improve the regret bounds of the Kernelized MDPs class. Calling $\mathscr{M}_{\mathscr{K}}$ the family of kernelized MDPs

Matérn kernel of parameter $\nu$, state of the algorithms for Kernelized MDPs enjoy

$$\text{RegretBound}(\text{KOVI}|\mathscr{M}_{\mathscr{K}}) \leq \widetilde{\mathcal{O}}\left(K^{\frac{3/2m+\nu}{m+2\nu}}\right),$$

the same rate that is achieved by ELEANOR. Instead, by the inclusion $\mathscr{M}_{\mathscr{K}} \subset \mathscr{M}_{\text{SS}} \subset \mathscr{M}_{\text{MS}}$, which hold for the same $\nu$ in all families, we deduce,

$$\text{RegretBound}(\text{CINDERELLA}|\mathscr{M}_{\mathscr{K}}) \leq \widetilde{\mathcal{O}}\left(K^{\frac{2m+\nu}{2m+2\nu}}\right).$$

Improving the regret for one part of the Kernelized MDP family is a contribution of independent interest, even if the presented results do not match the best known lower bound.

### 6.4.1. Smooth and Kernelized MDPs: Which is the Optimal Regret Bound

In this section, which closes the chapter, we will dwell a little more on the relationship between Kernelized MDPs, the comparison with our Smooth MDPs and the coverage problem that affects both.

As we have seen, despite the lower bound on regret of Kernelized MDPs being $\widetilde{\mathcal{O}}(\sqrt{\gamma_K K})$, known algorithms cannot do better than $\widetilde{\mathcal{O}}(\gamma_K\sqrt{K})$. If we substitute $\gamma_K = \mathcal{O}(K^{\frac{m}{2\nu+m}})$, which is the information gain bound for the Matérn kernel, the total regret bound gets a rate of $\widetilde{\mathcal{O}}\left(K^{\frac{3/2m+\nu}{m+2\nu}}\right)$, which is vacuous for $\nu \leq m/2$, as we discussed.

This gap has plagued Kernelized MDP authors for years. In particular, a major attempt to close the gap once and for all was made in 2023 by [134] ([135] for the ARXIV version, that has minor corrections), who proposed a regret bound of order $\widetilde{\mathcal{O}}\left(K^{\frac{m+\nu}{m+2\nu}}\right)$ which, if correct, would have closed the gap. The result giving the regret bound is Theorem 2 and, precisely, equation (21) in that paper. Without digging into the algorithm that they use, we can say that it is based on dividing $\mathcal{Z}$ into sub-regions, there called $\mathcal{Z}'$. As the authors recognise in their section 4.1, the hardest point of the analysis is to impose a concentration inequality for the Bellman update. To fit the current estimate $Q_h^k$ of the state-action value function, the algorithm use $r_h^k + P_h V_{h+1}^k$, which is a random function (as $V_{h+1}^k$ is estimated as well). Therefore, standard confidence bounds do not apply straightforwardly. The way [135], as most papers, deal with this issue, is to make a covering of the space $\mathcal{V}_{h+1}$ of the state-value functions at the next step, and then make a union bound over all the possible values for $V_{h+1}^t$.

Here the proof breaks. Instead of using the covering number of $\mathcal{V}_{h+1}$, they fix one region $\mathcal{Z}'$ and cover the space of functions restricted to $\mathcal{Z}'$, as it is said in the first eight lines of section 4.2. In this way, as said in the same part of the paper, the $\varepsilon-$covering number of the function class, for $\varepsilon = \mathcal{O}(\sqrt{\log(T)}/\sqrt{N_T(\mathcal{Z}')})$ (where $N_T(\mathcal{Z}')$ corresponds to the number of times region $\mathcal{Z}'$ is visited), results for be of order $\log(T)$. Despite the novel idea of splitting the domain in regions, this step reveals a gap in the argument. Line 7 of their algorithm, when the estimated $Q_h^t(\cdot)$ is computed, relies on equations (15-16), which do a Gaussian process regression with target $r_h(z') + V_{h+1}^t(s'_{h+1})$ for all $z' \in \mathcal{Z}'$ visited in the process. The issue here is that, while we know that $z' \in \mathcal{Z}'$, we cannot say anything about $s'_{h+1}$, which is sampled from $p_{h+1}(\cdot|z')$. The new state is not guaranteed to belong to any specific sub-region of $\mathcal{S}$. Therefore, it is not sufficient to do a cover restricted to the value functions $\mathcal{Z}' \to [0, 1]$.

The same authors admitted that this proof strategy is not able to get the desired regret bound (in fact, one of them contributed to the presentation of the open problem the sequent year [133]).

This situation further demonstrates how the necessity of employing a covering argument to account for the stochasticity of the estimated value function (referred to by some authors as the 'moving target problem') remains an arduous and pivotal challenge for the theoretical RL community. The adoption of novel frameworks and techniques to address this issue could potentially close the regret gaps across various classes of MDPs in a single stroke.

# 7 | Conclusions

In this final chapter of the thesis, the time has come to take stock of everything developed along this path. After introducing, in chapter 4, the set of assumptions of reinforcement learning on continuous spaces proposed in the literature, we devoted the second part of the chapter to defining a unifying family of assumptions, called Smooth MDPs (section 4.3). This opened a series of questions, listed in section 4.5, relating to theoretical learning guarantees in processes of this type. Thanks to the results we have provided in the last two chapters, we will try to answer each of them.

a In the generative model setting, is it possible to achieve a sample complexity guarantee that is non-vacuous for any $\varepsilon, m, \nu$? **Answer:** Yes, theorem 41 shows that, for the most general case of Weakly Smooth MDPs, the sample complexity is bounded with an order that is always well-defined. theorem 42 shows a similar result for the Mildly Smooth case, avoiding the exponential blow-up in $H$.

b The optimal sample complexity for identifying a $\varepsilon-$optimal arm in case of a Smooth Bandit problem is $\Theta(\varepsilon^{-2-m/\nu})$. Can we match this bound for the sample complexity of Smooth MDPs with access to a generative model? **Answer:** Yes, theorem 41 and theorem 42 show the optimal rate $\varepsilon^{-2-m/\nu}$ up to log-factors.

c In the online interaction setting, is it possible to achieve a regret bound that is always sublinear in $K$? **Answer:** Yes, for what concerns Mildly Smooth MDPs theorem 49 shows a regret bound that scales as $K^{\frac{2m+\nu}{2m+2\nu}}$, up to log-factors. For the most general case of Weakly Smooth processes, the best regret guarantees are given by theorem 45 which, unfortunately, is superlinear in some regimes.

d Can we achieve a regret guarantee of $\Theta(K^{\frac{m+\nu}{m+2\nu}})$, which matches the lower bound for Smooth Bandits? **Answer:** We have not been able to prove this regret rate in any setting, not even for Strongly Smooth MDPs. Proving this would close the historical gap that exists in Kernelized MDPs. This problem is intertwined with the infamous "moving target problem" (see section 6.4.1), a well-known obstacle for theoretical guarantees in online RL.

| Class | Algorithm | Regret Bound | S. C. | Tractable | Unknown reward |
|---|---|---|---|---|---|
| $\mathscr{M}_{\text{p-WS}}$ | PLSVI | N/A | $\exp(H)\varepsilon^{-2-m/\nu}$ | ✗ | ✓ |
| $\mathscr{M}_{\text{p-MS}}$ | PLSVI | N/A | $\text{poly}(H)\varepsilon^{-2-m/\nu}$ | ✓ | ✓ |
| $\mathscr{M}_{\text{WS}}$ | Eleanor | $\exp(H)K^{\frac{3m/2+\nu}{m+2\nu}}$ | N/A | ✗ | ✓ |
| $\mathscr{M}_{\text{MS}}$ | Cinderella | $\text{poly}(H)K^{\frac{2m+\nu}{2m+2\nu}}$ | $\text{poly}(H)\varepsilon^{-2-2m/\nu}$ | ✗ | ✓ |
| $\mathscr{M}_{\text{SS}}$ | LSVI-UCB | $\text{poly}(H)K^{\frac{2m+\nu}{m+2\nu}}$ | N/A | ✓ | ✓ |

Table 7.1: The table follows the conventions of table 4.1. Class names are as usual: $\mathscr{M}_{\text{p-WS}}$ = periodic Weakly Smooth, $\mathscr{M}_{\text{p-MS}}$ = periodic Mildly Smooth, $\mathscr{M}_{\text{WS}}$ = Weakly Smooth, $\mathscr{M}_{\text{MS}}$ = Mildly Smooth and $\mathscr{M}_{\text{SS}}$= Strongly Smooth.

To conclude this thesis, we add to the table 4.1 the main results we proved in this thesis. All our theorems are related to Smooth MDPs, even if our results are also applicable to other classes, as shown by the comparison with the literature we made in section 4.4. Each row of table 7.1 represents one of our main performance guarantees for some algorithm and some setting. The first two come from theorem 41 and theorem 42, the two main results under generative model. The next two are theorem 45 and theorem 49, which show the strongest guarantees for the online case. Finally, the last row comes from theorem 50, and provides the only efficient algorithm for the online case.

Most of the algorithms we presented cannot enjoy both a regret guarantee and a sample complexity guarantee, even if for different reasons. In fact, PLSVI only works under the assumption of having access to a generative model, while the regret is intrinsically linked to the online case. Eleanor and LSVI-UCB enjoy a regret bound that is vacuous in some regimes, therefore, we cannot use proposition 9 to turn their bounds into sample complexity guarantees for every Smooth MDPs. Nonetheless, sample complexity guarantees exist provided that $m, \nu$ are such that the regret bound is sublinear. Cinderella is the only algorithm with a regret bound that is always non-vacuous: for this reason its sample complexity guarantee exists for all $m, \nu$. Note that, letting $\nu \to +\infty$, all the regret bounds converge to $\sqrt{K}$. In fact, in this case the process is so smooth that our algorithms work for a feature map of length proportional to $\log(K)^m$. This phenomenon is closely related to the fact that the covering numbers of the spaces of infinitely differentiable functions grow in a polylogarithmic way.

In this table we have also taken into account the computational complexity, in the *Tractable* column. As before, with Tractable we mean polynomial dependence in $H, K$

(also $n, \varepsilon$ for algorithm that only provide sample complexity guarantees), but not in $m$, and the absence of a planning oracle (see section 4.2.3). Importantly, while ELEANOR and CINDERELLA are inefficient by design of the algorithm, PLSVI is inefficient for Weakly Smooth MDPs because the reduction requires to build a feature map of length $\exp(H)$.

This table concludes the summary of the results demonstrated in this thesis. In this work, we have unified notions of MDPs originating from very distant fields of RL theory under some common smoothness characteristics. The methods we have proposed to solve them modify and extend some seminal algorithms that come from MDPs with linear function approximation. Based on our results, several interesting questions remain, in addition to the one already mentioned regarding the remaining gap between upper and lower bounds in regret. In particular, CINDERELLA, the only presented algorithm that consistently provides sublinear regret bounds, does not work in the Weakly Smooth setting (at least with our reduction). Is it possible to prove a sublinear regret bound for the most general Smooth MDP class? Also, as CINDERELLA is not computationally efficient, is there a way to design an algorithm that is tractable while still enjoying the no-regret property for every Mildly Smooth MDP?

## 7.1. Next Research Directions

The mathematical framework of this thesis leverages fundamental principles of mathematical analysis to provide a unified treatment of RL assumptions in continuous spaces. Indeed, we have seen that Sobolev-type assumptions had already been used in the bandit context. Although the adopted solution effectively improves the statistical complexity bounds for smooth MDPs, avoiding the infamous $\varepsilon^{-m}$ scaling due to the curse of dimensionality, the requirement for differentiable functions $\nu$ is demanding in many scenarios.

A possible solution to this problem lies in a tool of modern mathematical analysis that, to the best of our knowledge, is largely unexplored in theoretical machine learning. This tool is called *Korobov spaces*, also known as Sobolev spaces of dominating mixed smoothness. First introduced in [118] and rigorously defined in [140], these ideas, which represented the flagship of Soviet mathematics, remained unknown in the West for over twenty years, until [110] republished the results in English. In their simplest formulation, Korobov spaces contain $m-$dimensional functions $f$ such that

$$\mathrm{K}^{1,p}(\mathcal{X}) := \left\{ f : \frac{\partial^m f}{\partial x_1, \dots \partial x_m} \in L^p(\mathcal{X}) \right\},$$

for some $p \in [1, \infty]$. The big difference with respect to the Sobolev spaces that we used

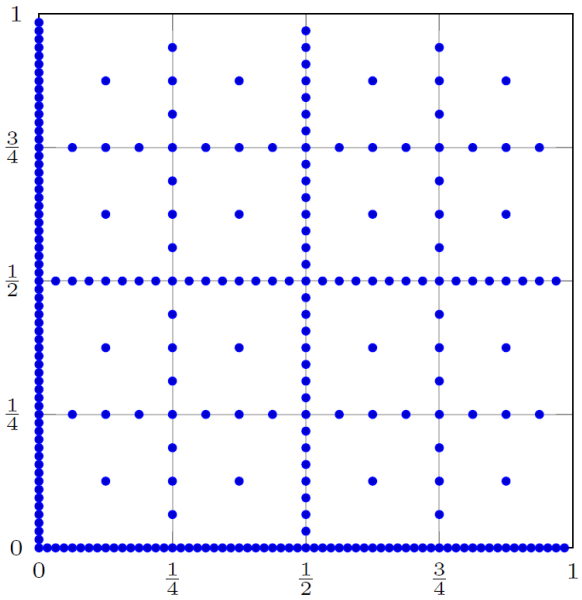



Figure 7.1: Smolyak grid for $m = 2$, a picture from Figure 8 of [38]

.

in this work lies in the fact that, while our formulation required having all the derivatives whose total degree is $\nu$ in $L^p$, in this case only one derivative is required, but with respect to each of the variables of the function. In fact for $\dim(\mathcal{X}) > 2$, one has neither

$$\mathrm{W}^{2,p}(\mathcal{X}) \subseteq \mathrm{K}^{1,p}(\mathcal{X}) \qquad \text{nor} \qquad \mathrm{K}^{1,p}(\mathcal{X}) \subseteq \mathrm{W}^{2,p}(\mathcal{X}).$$

In fact, $\mathrm{K}^{1,p}(\mathcal{X})$ requires $\frac{\partial^3 f}{\partial x_1 \partial x_2 \partial x_3} \in L^p(\mathcal{X})$, which $\mathrm{W}^{2,p}(\mathcal{X})$ does not, and the latter requires $\frac{\partial^2 f}{\partial x_1^2} \in L^p(\mathcal{X})$, which $\mathrm{K}^{1,p}(\mathcal{X})$ does not.

The reason why the mixed smoothness assumption is often considered more reasonable than the isotropic smoothness we used in this thesis is that a function $F \in \mathrm{K}^{1,p}([0,1]^m)$ can be written as

$$F(x_1, \dots x_m) := \int_0^{x_1} \int_0^{x_2} \cdots \int_0^{x_m} f(t_1, t_2, \dots t_m) \mathrm{d}t_1 \mathrm{d}t_2 \dots \mathrm{d}t_m \qquad f \in L^p([0,1]^m).$$

This kind of assumptions is relatively natural in many practical problems and, crucially, reduces to standard Lipschitzness for $p = +\infty$ and $m = 1$.

The most astonishing fact about these functions is that their approximation properties resembles the ones of the much smaller class $\mathrm{W}^{m,p}([0,1]^m)$, i.e. the Sobolev space for $\nu = m$. While we have seen (theorem 43 plus length of the feature map) that a proper vector space $\varepsilon-$approximates all functions in $\mathrm{W}^{m,\infty}([0,1]^m)$ with $d = \mathcal{O}(\varepsilon^{-1})$ terms, [24]

shows that the same can be achieved for $\mathrm{K}^{1,\infty}([0,1]^m)$ with

$$d = \Theta\left(\log(1/\varepsilon)^m \varepsilon^{-1}\right).$$

The approximation scheme bringing this rate uses the vector space of hat functions, and is based on evaluating the function on the Smolyak grid, fig. 7.1, a clever construction that allows to find the non-equispaced points that are most informative for a function of dominating mixed smoothness.

Integrating these types of assumptions into reinforcement learning (or even just bandit), as done with smooth MDPs, would allow for the theoretically valid solution of highly interesting continuous-state problems. However, the technical difficulty of this undertaking would be particularly high: despite the grid, Smolyak's approach is much more advanced than the simple discretization method we saw for Lischitz MDPs. Such a method might not necessarily allow for a variant that balances the exploration-exploitation dilemma; on the contrary, Smolyak's construction is extremely rigid and may require sampling the function into points that correspond to a very low reward, leading to poor regret guarantees.

## 7.2. Conclusion of the Conclusions

Unlike the more tractable Tabular MDPs, the study of MDPs with continuous spaces continues to face substantial theoretical hurdles. These gaps in our current understanding are not merely technical obstacles but represent deep, foundational questions that require a concerted research effort. We trust that the insights presented here will help guide future research toward addressing these open problems with both rigor and innovation.

# A | Notation Summary

Given the length of this work, we present the notation adopted in each of the chapters in a separate table.

| Symbol | Meaning |
|---|---|
| $a, \mathcal{A}, A$ | Action, space of actions, Cardinality of the space of actions. |
| $\mathscr{P}_a$ | Reward distribution associated to action $a$. |
| $K$ | Total number of rounds / episodes. |
| $k$ | Index of the current round. |
| $y_k$ | Stochastic reward at round $k$. |
| $r(a)$ | Reward function ( $r_h(s,a)$ for MDPs). |
| $\pi$ | Policy. |
| $J^\pi$ | Return of the policy $\pi$. |
| $R_K(\pi)$ | Regret of policy $\pi$ ($\overline{R}_K(\pi) = \mathbb{E}[R_K(\pi)]$). |
| $\mathcal{F}$ | Set of possible reward function for a class of bandit problems. |
| $s, \mathcal{S}, S$ | State, Space of states, Cardinality of the space of states. |
| $H$ | Time horizon of the episode. |
| $\Delta(\mathcal{X})$ | Probability distributions on $\mathcal{X}$. |
| $p_h$ | Transition function at step $h$. |
| $V_h^\pi$ | State value function, $V_h^* = \sup_\pi V_h^\pi$. |
| $Q_h^\pi$ | State-action value function, $Q_h^* = \sup_\pi Q_h^\pi$. |
| $\mathcal{T}_h$ | Bellman optimality operator. |
| $s_h^k, a_h^k, y_h^k$ | state/action/reward at stage $h$ of episode $k$. |

Table A.1: Table of the notation for chapter 2.

| Symbol | Meaning |
|---|---|
| $\mathcal{Z}$ | $\mathcal{S} \times \mathcal{A}$. |
| $\sigma$ | Subgaussianity constant. |
| $J^*$ | Return of the optimal policy. |
| $\mathscr{M}$ | Class of MDPs. |
| $\mathscr{A}$ | Algorithm. |
| $\mathscr{L}$ | Loss. |
| $\mathscr{H}$ | Hypothesis space. |
| $\widehat{p}_h^k$ | Estimated transition function. |
| $\widehat{r}_h^k$ | Estimated reward function. |
| $b_h^k$ | Bonus function. |
| $\mathcal{O}$ | We say $f(x) = \mathcal{O}(g(x))$ if $f(x) \leq Cg(x)$ for some $C > 0$. |
| $\widetilde{\mathcal{O}}$ | Same as $\mathcal{O}$ but ignoring log-factors. |
| $\Omega$ | We say $f(x) = \Omega(g(x))$ if $f(x) \geq cg(x)$ for some $c > 0$. |
| $\Theta$ | We say $f(x) = \Theta(g(x))$ if $f(x) = \mathcal{O}(g(x))$ and $f(x) = \Omega(g(x))$. |
| $\varphi_h$ | Feature map. |
| $d$ | Dimension of $\varphi_h$. |
| $Q_h[\theta](s, a)$ | $\varphi_h(s, a)^\top \theta$. |

Table A.2: Table of the notation for chapter 3.

| Symbol | Meaning |
|---|---|
| $m_1$ | Dimension of the state space. |
| $m_2$ | Dimension of the action space. |
| $m$ | $= m_1 + m_2$. |
| $L^p$ | Space of functions such that $\int_{\mathcal{X}} \lvert f(x)\rvert^p \mathrm{d}x < +\infty$. |
| $D^\alpha$ | Derivative w.r.t. the multiindex $\alpha$. |
| $\mathrm{W}^{\nu,p}$ | Sobolev space of functions that have $\nu$ derivatives in $L^p$. |
| $\lVert f\rVert_{\mathrm{W}^{\nu,p}}$ | Norm of $f$ in $\mathrm{W}^{\nu,p}$. |
| $L(f)$ | Lipschitz constant of $f$. |
| $\mathcal{N}$ | Gaussian distribution. |
| $\Sigma$ | Covariance matrix. |
| $\mathcal{W}(\mu,\omega)$ | Wasserstein distance between measures $\mu, \omega$. |
| $\mathrm{TV}(\mu,\omega)$ | Total variation distance between measures $\mu, \omega$. |
| $\mathcal{H}(\mathscr{K})$ | Reproducing kernel Hilbert space w.r.t kernel $\mathscr{K}$. |
| $\gamma_K$ | Maximal information gain. |
| $\mathcal{N}_\infty$ | Covering number (see definition 15). |
| $\mathcal{B}_r(\mathcal{H})$ | Ball with radius $r > 0$ in $\mathcal{H}$. |
| $\dim_{\mathrm{E}}$ | Eluder dimension (see definition 16). |
| $\dim_{\mathrm{BE}}, \dim_{\mathrm{DE}}$ | Distributional / Bellmann Eluder dimension. |
| $\mathscr{M}_{\mathrm{SS}}$ | Class of Strongly Smooth MDPs, with constants $\mathrm{C}_{\mathrm{r}}^{\mathrm{SS}}$, $\mathrm{C}_{\mathrm{p}}^{\mathrm{SS}}$. |
| $\lVert\mathcal{T}\rVert^{\circ}_{X\to Y}$ | Growth functional of $\mathcal{T}$ (4.12). |
| $\mathscr{M}_{\mathrm{MS}}$ | Class of Mildly Smooth MDPs, with constant $\mathrm{C}_{\mathcal{T}}^{\mathrm{MS}}$. |
| $\mathscr{M}_{\mathrm{WS}}$ | Class of Weakly Smooth MDPs, with constant $\mathrm{C}_{\mathcal{T}}^{\mathrm{WS}}$. |
| $\mathscr{K}_\nu$ | Matérn kernel parametrized with $\nu > 0$. |
| $\mathcal{C}^\infty$ | Functions that are differentiable infinity many times. |
| $T_y^{\nu_*}[f]$ | Taylor's polynomial of $f$, centered in $y$. |

Table A.3: Table of the notation for chapter 4.

| Symbol | Meaning |
|---|---|
| $\mathrm{T}_N$ | Vector space of trigonometric polynomials on $\mathcal{Z}$ whose degree does not exceed $N$. |
| $\phi_N^m$ | $m-$dimensional Fourier feature map of degree $N$. |
| $d$ | $= \binom{2N+m}{m} \leq (2N+1)^m$. |
| $f * g$ | Circular convolution. |
| $\mathrm{J}_\mathrm{T}$ | Jackson's constant (theorem 33). |
| $\mathrm{W}_p^{\nu,\infty}$ | Sobolev space $\mathrm{W}^{\nu,\infty}$ with periodic boundary conditions. |
| $\xi(f;\varphi)$ | Uniform distance between function $f$ and span$(\varphi)$. |
| $\mathscr{D}_N$ | Dirichlet kernel of degree $N$. |
| $\mathscr{V}_N$ | de la Vallée-Poussin kernel of degree $N$. |
| $\Lambda$ | Lebesgue constant. |
| $\beta^+, \beta^-, \mathscr{V}_N^+ \mathscr{V}_N^-$ | see (5.5)(5.4). |
| $\mathcal{D}_h$ | Dataset built with quasi-optimal design in section 5.2.1. |
| $\widetilde{Q}_h$ | Surrogate $Q-$function. |
| E | Extension operator. |

Table A.4: Table of the notation for chapter 5.

| Symbol | Meaning |
|---|---|
| $\mathrm{P}_N$ | Vector space of polynomials on $\mathcal{Z}$ whose degree does not exceed $N$. |
| $d$ | $= \binom{N+m}{m}$. |
| $\varphi_N$ | Legendre polynomials feature map of degree $N$. |
| $\mathrm{J}_\mathrm{T}$ | Jackson's constant for polynomials (theorem 43). |
| $\mathcal{I}$ | Inherent Bellman error. |
| $\mathcal{U}_h$ | Partition of the state-action space (Locally linearizable MDP). |
| $N_h$ | $= \lvert\mathcal{U}_h\rvert$ |
| $\mathcal{B}_{h,n}$ | Space of candidate parameters (Locally linearizable MDP). |
| $L_\phi$ | $= \sup_{h\in[H],z\in\mathcal{Z}} \lVert\phi_h(z)\rVert_2$ |
| $\mathcal{R}_{h,n}$ | $= \mathrm{diam}(\mathcal{B}_{h,n})$ |
| $\mathrm{K}^{1,p}$ | Korobov space |

Table A.5: Table of the notation for chapters 6 and 7.

# B | Mathematical Analysis

Let $\mathcal{X}$ be a measurable set and $\mu$ a positive measure over $\mathcal{X}$. We define, for any $f : \mathcal{X} \to \mathbb{R}$, and every $p \in [1, \infty)$

$$\|f\|_{L^p} := \left( \int_{\mathcal{X}} |f(x)|^p \mu(dx) \right)^{1/p}.$$

for $p = +\infty$, let us call

$$\|f\|_{L^\infty} := \sup_{x \in \mathcal{X}} |f(x)|.$$

It can be proved that the previous satisfy the assumptions to be called norm (formally, if $\mathcal{X}$ is infinite, in order to let $\|\cdot\|_{L^p}$ be a norm, we need to ignore null-measure sets, so that functions that are equal $\mu-$almost everywhere are considered to be the same, see [106]). We therefore call

$$L^p(\mathcal{X}, \mu) := \{f : \|f\|_{L^p} < +\infty\}.$$

We will omit $\mu$ in case we are dealing with the uniform measure. It can be proved that all these spaces are Banach spaces and, for $p = 2$, Hilbert spaces, where the scalar product writes as $\langle f, g \rangle = \int_{\mathcal{X}} f(x) g(x) \mu(dx)$.

## B.1. Sobolev Spaces

Intuitively, a Sobolev space is a space of functions whose derivatives belong to an $L^p$ space (see [3] for a more detailed introduction). First, we define the concept of weak derivative.

**Definition 27.** *We denote $|\boldsymbol{\alpha}| = \sum_{i=1}^m \alpha_i$ the magnitude of the multiindex.*

Given the magnitude of the multiindex, the weak derivative is defined in the following way.

**Definition 28.** *Let $\mathcal{X} \subseteq \mathbb{R}^m$ and $f : \mathcal{X} \to \mathbb{R}$. The weak derivative $D^{\boldsymbol{\alpha}}$ is defined as follows*

- *if* $f \in \mathcal{C}^{|\boldsymbol{\alpha}|}(\mathcal{X})$*, then*

$$D^{\boldsymbol{\alpha}} f(x) = \frac{d^{|\boldsymbol{\alpha}|} f}{dx_1^{\alpha_1} \dots dx_m^{\alpha_m}}(x).$$

- *otherwise, we note* $D^{\boldsymbol{\alpha}} f := g$*, where is* $g : \mathcal{X} \to \mathbb{R}$*, such that for all* $\varphi \in \mathcal{C}_c^\infty(\mathcal{X})$*,*

$$\int_{\mathcal{X}} g(x)\varphi(x)dx := (-1)^{|\boldsymbol{\alpha}|} \int_{\mathcal{X}} f(x) D^{\boldsymbol{\alpha}} \varphi(x) dx.$$

*(it can be proved that such* $g$ *is uniquely defined a.e.)*

This allows us to define the Sobolev spaces as follows

$$\mathrm{W}^{\nu,p}(\mathcal{X}) := \{f : \forall |\boldsymbol{\alpha}| \leq \nu,\ D^{\boldsymbol{\alpha}} f \in L^p(\mathcal{X})\}.$$

This space, when endowed with the following norm

$$\|f\|_{\mathrm{W}^{\nu,p}} := \max_{|\boldsymbol{\alpha}| \leq \nu} \|D^{\boldsymbol{\alpha}} f\|_{L^p}, \tag{B.1}$$

becomes a Banach space (also an Hilbert space for $p = 2$). The following property holds true whenever $\mathcal{X} \subset \mathbb{R}^d$ is bounded:

$$\mathrm{W}^{\nu',p'}(\mathcal{X}) \subseteq \mathrm{W}^{\nu,p}(\mathcal{X}).$$

for all $p' \geq p$ and $\nu' \geq \nu$. Therefore, these spaces get smaller the bigger $p$ and $\nu$. For our purposes, the case $p = +\infty$ is of paramount importance.

### B.1.1. The Sobolev-Hölder Spaces $\mathrm{W}^{\nu,\infty}$

As follows from the previous definitions, the space $\mathrm{W}^{\nu,\infty}(\mathcal{X})$ contains all the functions such that

$$D^{\boldsymbol{\alpha}} f \in L^\infty(\mathcal{X}).$$

This has in interesting interpretation. As it is well-known (see [65]) that a function is $L-$Lipschitz continuous if and only if all its partial derivatives are uniformly bounded by $L$, for any bounded set $\mathcal{X}$ we have

$$f \in \mathrm{W}^{\nu,\infty}(\mathcal{X}) \iff D^{\boldsymbol{\alpha}} f \text{ is Lipschitz continuous } \forall |\boldsymbol{\alpha}| = \nu - 1.$$

Under this lens, the space $\mathrm{W}^{\nu,\infty}(\mathcal{X})$ shall be seen as the space of functions whose derivatives of order $\nu - 1$ are Lipschitz continuous. At the same time, with a slight abuse of notation, this allows us to extend the previous definition beyond $\nu \in \mathbb{N}$ requiring Hölder continuity instead of Lipschitz continuity:

$$f \in \mathrm{W}^{\nu,\infty}(\mathcal{X}) := D^{\boldsymbol{\alpha}} f \text{ is } (\nu - \nu_*)\text{−Hölder continuous } \forall |\boldsymbol{\alpha}| = \nu_* := \lceil \nu - 1 \rceil. \tag{B.2}$$

In such cases, the Hölder inequality writes as

$$\forall \boldsymbol{\alpha} : \ |\boldsymbol{\alpha}| = \nu_*, \forall x, y \in \Omega, \qquad |D^{\boldsymbol{\alpha}} f(x) - D^{\boldsymbol{\alpha}} f(y)| \leq L_\nu(f) \|x - y\|_\infty^{\nu - \nu_*},$$

and the norm generalizes to

$$\|f\|_{\mathrm{W}^{\nu,\infty}} := \max \left\{ \max_{|\alpha| \leq \nu_*} \|D^{\boldsymbol{\alpha}} f\|_{L^\infty}, L_\nu(f) \right\} \tag{B.3}$$

which extends eq. (B.1) (but only in the case $p = \infty$). $\mathrm{W}^{\nu,\infty}$ builds both on the idea of bounding the derivatives of the function and on a Hölder inequality. For this reason, when talking about this spaces, which is the cornerstone of the Smooth MDP family, we name it "Sobolev-Hölder" spaces.

The smoothness properties of characterizing functions in a Sobolev space relate to the concepts of Taylor and Fourier series, that we are recalling in the next sections.

## B.2. Taylor Series

A fundamental property of the Sobolev space $\mathrm{W}^{\nu,\infty}$ that we are considering is the possibility of approximating the elements of this spaces with functions that are piecewise polynomial.

Let $\mathcal{X} \subset [-1, 1]^m$ and $f \in \mathrm{W}^{\nu,\infty}(\mathcal{X})$. The Taylor polynomial $T_y^{\nu_*}[f](\cdot)$ centered in $y \in \mathcal{X}$ of order $\nu_*$ is

$$T_y^{\nu_*}[f](x) = \sum_{\|\boldsymbol{\alpha}\|_1 \leq \nu_*} \frac{D^{\boldsymbol{\alpha}} f(y)}{\boldsymbol{\alpha}!} (x - y)^{\boldsymbol{\alpha}}.$$

If $\nu_* = 0$ the polynomial reduces to a constant, $T_y^{\nu_*}[f](x) = f(y)$. This means that for functions that are not smoother than the Lipschitz class, and the Taylor polynomial in one point reduces to the value of the function on that point. Conversely, as the smoothness

of the function increases, the complexity of the Taylor polynomial increases very rapidly. While in the case $m = 1$, each additional degree of smoothness corresponds to one more degree of the polynomial, the issue in multiple dimensions becomes considerably more complex, and it can be shown that the total number of terms appearing is

$$|\{\boldsymbol{\alpha} \in \mathbb{N}^m : \ \|\boldsymbol{\alpha}\|_1 \leq \nu_*\}| = \binom{\nu_* + m}{m} \leq \min\left\{(\nu_* + 1)^m, (m+1)^{\nu_*}\right\}.$$

This number grows faster than any polynomial in the minimum between $\nu_\star$ and $m$. While the rate in $m$ is an unlucky consequence of the curse of dimensionality, a big $\nu_*$ brings also strong approximation properties.

**Theorem 51.** *Let $f \in W^{\nu,\infty}(\mathcal{X})$. For fixed $y \in \mathcal{X}$, let $T_y^{\nu_*}[f](\cdot)$ be the Taylor polynomial centered in $y$. Then,*

$$\forall x \in \mathcal{X}, \ |f(x) - T_y^{\nu_*}[f](x)| \leq L_\nu(f)\|x - y\|_\infty^\nu.$$

The previous result ensures that, for a large $\nu$ and $x$ that is sufficiently close to $y$, the Taylor polynomial gives a very good approximation for the value of $f(x)$. When needing a uniform approximation over the domain, the idea is often to split $\mathcal{X}$ in several balls of diameter no more than $\varepsilon$, and to use the piecewise polynomial function given by the Taylor polynomials in the centers of each ball.

Note that eq. (B.3) guarantees that the $L_\nu(f)$ that appears in the former theorem is always upper bounded by $\|f\|_{\mathrm{W}^{\nu,\infty}}$. This allows to relate the approximation rate to the norm of $\|f\|_{\mathrm{W}^{\nu,\infty}}$ itself. Another thing that is worth mentioning is that the Taylor polynomial take the form of a scalar product over $\|\boldsymbol{\alpha}\|_1 \leq \nu_*$ of one vector with components $\frac{D^{\boldsymbol{\alpha}} f(y)}{\boldsymbol{\alpha}!}$ and another of components $(x-y)^{\boldsymbol{\alpha}}$. In fact we can rewrite it in the following form

$$T_y^{\nu_*}[f](x) = \sum_{\|\boldsymbol{\alpha}\|_1 \leq \nu_*} \frac{D^{\boldsymbol{\alpha}} f(y)}{\boldsymbol{\alpha}!}(x-y)^{\boldsymbol{\alpha}} =: \boldsymbol{w}^\top \psi_y^{\nu_*}(x). \tag{B.4}$$

The first term only depends on $f$ and its derivatives only evaluated in $y$. The second depends on $x$, but does not depend on $f$ at all. Thus, this form allows to approximate the full function as something that takes into account the function and its derivatives evaluated in a specific point and and a fixed, point dependent, feature map.

## B.3. Fourier Analysis

Another form of representation for a generic function is given by the Fourier series. Unlike the Taylor polynomials we presented earlier, the Fourier polynomial does not provide a local approximation, but a global one. Let $f \in L^2([-1,1])$. Given

$$a_n = \int_{-1}^{1} \cos(n\pi x) f(x)\ \mathrm{d}x \qquad b_n = \int_{-1}^{1} \sin(n\pi x) f(x)\ \mathrm{d}x,$$

we call Fourier series truncated to order $N$ the function

$$S_N[f](x) = \frac{a_0}{2} + \sum_{n=1}^{N} a_n \cos(n\pi x) + b_n \sin(n\pi x).$$

By convenience, we can stack the coefficients of this representation into a unique vector $\theta$ and write the Fourier series in the form $\phi_N(x)^\top \theta$, where $\theta \in \mathbb{R}^{2N+1}$ contains the previous coefficients $a_n, b_n$ and $\phi_N(\cdot)$ contains the trigonometric monomials. In this way, we can emphasize the linear dependence on the coefficients:

$$S_N[f](x) = \phi_N(x)^\top \theta. \tag{B.5}$$

There is another way, more compact, to express the Fourier series of a function $f$, that is trough the convolution with the Dirichlet kernel. The Dirichlet kernel (see [59]) is the function $\mathscr{D}_N : [-1,1] \to \mathbb{R}$ defined as follows,

$$\mathscr{D}_N(x) = \frac{\sin((N+1/2)\pi x)}{\sin(\pi x/2)},$$

and is known to satisfy

$$S_N[f](x) = \mathscr{D}_N * f(x),$$

for any $f \in L^2([-1,1])$. Here, $*$ denotes the circular convolution, that means

$$f * g(x) := \int_{-1}^{1} f(y) g(x-y)\ dy,$$

where $g$ is extended periodically when its argument lays outside of the domain $[-1,1]$: for $z > 1$, $g(z) := g(z-2)$ and for $z < -1$, $g(z) := g(z+2)$.

The meaning of the Fourier series is to project an arbitrary function $f$ on the vector space $\mathrm{T}_N$ of the trigonometric polynomials of degree $N$ at most. Precisely, the Fourier series

correspond to the $L^2$ projection, namely

$$S_N[f] = \arg\min_{v \in \mathrm{T}_N} \|f - v\|_{L^2}.$$

While this result is very beautiful in $L^2$, it does not generalize to more general $L^p$ spaces that are not Hilbert. In particular, for the space $L^\infty$, which is of paramount importance in this work, it can be proved that

$$\|S_N[f] - f\|_{L^\infty} \gtrsim \underbrace{\|\mathscr{D}_N\|_{L^1}}_{\sim \log(N)} \inf_{v \in \mathrm{T}_N} \|f - v\|_{L^\infty}.$$

This makes the Fourier projection substantially worse than the $L^\infty$ projection operator, as $N \to \infty$. One way to get rid of this problem is the with Valleè de la Poussing Kernel [32, 33], the function defined in this way:

$$\mathscr{V}_{2N}(x) = \frac{1}{N} \sum_{k=N}^{2N} \mathscr{D}_N(x).$$

This kernel is the sum of the Dirichlet kernels from $n = N$ to $n = 2N$, and is endowed with three powerful properties

1. If $f \in \mathrm{T}_N$, then $\mathscr{V}_{2N} * f = f$ (like $\mathscr{D}_N$)
2. For any $f \in L^2([-1, 1])$, $\mathscr{V}_{2N} * f(x) \in \mathrm{T}_{2N}$ (like $D_{2N}$)
3. $\|\mathscr{V}_{2N}\|_{L^1} \leq \frac{1}{3} + \frac{2\sqrt{3}}{\pi} \approx 1.43$ as proved from [85], which stands in sharp contrast with the result for the Dirichlet Kernel.

Calling $f^\star := \arg\min_{v \in \mathrm{T}_N} \|f - v\|_{L^\infty}$, the previous points entail what follows

$$\begin{aligned}
\|\mathscr{V}_{2N} * f - f\|_{L^\infty} &\leq \|\mathscr{V}_{2N} * f - \mathscr{V}_{2N} * f^\star\|_{L^\infty} + \|\mathscr{V}_N * f^\star - f\|_{L^\infty} && \text{(B.6)} \\
&\leq \|\mathscr{V}_{2N} * (f - f^\star)\|_{L^\infty} + \|f^\star - f\|_{L^\infty} && \text{(B.7)} \\
&\leq \|\mathscr{V}_{2N}\|_{L^1} \|f - f^\star\|_{L^\infty} + \|f^\star - f\|_{L^\infty} && \text{(B.8)} \\
&= \underbrace{(1 + \|\mathscr{V}_{2N}\|_{L^1})}_{\leq \frac{4}{3} + \frac{2\sqrt{3}}{\pi}} \inf_{v \in \mathrm{T}_N} \|f - v\|_{L^\infty}. && \text{(B.9)}
\end{aligned}$$

(Here, the second line is due to the fact that $\mathscr{V}_{2N} * f^\star = f^\star$ and the third from the fact that $\|g * f\|_{L^\infty} \leq \|g\|_{L^1} \|f\|_{L^\infty}$ always). The previous eq. (B.9) ensures that the $\mathscr{V}_{2N}$ kernel

is only worse than the $L^\infty$ projection $f^\star$ by a constant factor.

### B.3.1. Multi-dimensional Fourier Series

Most of the theory of Fourier series, defined in the previous section for functions on $[-1,1]$, generalizes *mutatis mutandis* to functions on $[-1,1]^m$. In fact, like we did in B.5, we can write the multi-dimensional Fourier series of a function $f \in L^2([-1,1]^m)$ as

$$S_N[f](x) = \phi_N^m(x)^\top \theta,$$

where $\phi_N^m(x)$ is the vector of all trigonometric monomials whose total degree does not exceed $N$ (f.e. the degree of the trigonometric monomial $\sin(2\pi x_1)\cos(3\pi x_2)$ is $2+3=5$). Therefore, while we needed $2N+1$ coefficients to have the Fourier Series of a function in one dimension, we now need much more. Precisely, the number corresponds to

$$|\{\boldsymbol{\alpha} \in \mathbb{Z}^m : \ \|\boldsymbol{\alpha}\|_1 \leq N\}| = \binom{2N+m}{m} \leq (2N+1)^m . \tag{B.10}$$

The De La Valleè Poussin Kernel $\mathscr{V}$ is defined as well, and its norm is bounded as $\|\mathscr{V}_N\|_{L^1} \leq 2^m c$ (see [94]).

These multidimensional series are often used to approximate functions in the corresponding Sobolev space with periodic boundary conditions. In fact, we call

$$\mathrm{W}_{\mathrm{p}}^{\nu,\infty}(\mathcal{Z}) := \{f \in \mathrm{W}^{\nu,\infty}(\mathcal{Z})\} \cap \{f \text{ periodic on } \mathcal{Z}\}.$$

The latter condition ensures that $f$ must not only be smooth in the interior of $\mathcal{Z}$, but also at the boundary; in the sense that for any multiindex $\boldsymbol{\alpha}$ with $|\boldsymbol{\alpha}| \leq \nu$

$$\lim_{\varepsilon \to 0} D^{\boldsymbol{\alpha}} f(z_1, \dots, -1+\varepsilon, \dots, z_m) = D^{\boldsymbol{\alpha}} f(z_1, \dots, 1, \dots, z_m).$$

Ensuring that the limit can "jump" from one side of the boundary to the other reveals necessary to ensure that the trigonometric polynomials, which always enjoy this property, have good approximation properties.

## B.4. Reproducing Kernel Hilbert Spaces

Reproducing Kernel Hilbert spaces [8, 62] are a fundamental tool for modern functional analysis and theoretical machine learning.

Let $\mathcal{X}$ be any non-empty set. A function $\mathscr{K} : \mathcal{X}^2 \to \mathbb{R}$ is a *kernel* if the following two properties hold (see Defnition 2.1 in [58]).

- $\mathscr{K}$ is symmetric: for every $x, x' \in \mathcal{X}$, $\mathscr{K}(x, x') = \mathscr{K}(x', x)$
- $\mathscr{K}$ is positive definite: for any $n$, any sequence $x_1, \dots x_n$ of points in $\mathcal{X}$ and real coefficients $c_1, \dots c_n$,

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_i c_j \mathscr{K}(x_i, x_j) \geq 0.$$

Several examples of kernel exist, and every function that takes the form $\mathscr{K}(x, x') = \phi(x)^\top \phi(x')$, for $\phi : \mathcal{X} \to \mathbb{R}^d$ is a valid kernel. For $\mathcal{X} = [-1, 1]^m$, the most popular kernel is perhaps the squared-exponential kernel

$$\mathscr{K}(x, x') = \sigma^2 \exp\left(-\frac{\|x - x'\|_2^2}{2\ell^2}\right),$$

which is characterized by parameters $\sigma, \ell$ that represent how spiky the function is. This kernel function cannot be represented as a scalar product $\phi(x)^\top \phi(x)$ for any finite-dimensional feature map. A generalization of this kernel function is given by the family of Matérn kernels [84], defined as

$$\mathscr{K}(r; \nu) = \sigma^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu} r}{\ell}\right)^\nu K_\nu \left(\frac{\sqrt{2\nu} r}{\ell}\right), \qquad r = \|x - x'\|_2 \tag{B.11}$$

Here, $\sigma, \ell$ have the same meaning of the previous formula, $\Gamma$ denotes the Gamma function and $K_\nu$ the Bessel's function of second type. $\nu \in (0, \infty)$ is a parameter that determines the smoothness of the function; for $\nu \to \infty$ this kernel gets similar properties to the squared exponential kernel.

As a fundamental property of these functions, Moore-Aronszajn theorem ensures that, for every valid kernel $\mathscr{K}$ we can uniquely define the associated *reproducing kernel Hilbert space* $\mathcal{H}(\mathscr{K})$. Let us define

$$\mathcal{H}_0(\mathscr{K}) := \left\{ f(\cdot) = \sum_{i=1}^{n} c_i \mathscr{K}(x_i, \cdot) \middle| c_1, \dots c_n \in \mathbb{R}, x_1, \dots x_n \in \mathcal{X} \right\}, \tag{B.12}$$

the space of finite linear combinations of kernel functions. Also, define the scalar product

$$\langle f, g \rangle_{\mathscr{K}} := \sum_{i=1}^{n} \sum_{j=1}^{n} c_i c'_j \mathscr{K}(x_i, x'_j),$$

where, following eq. (B.12), $f(\cdot) = \sum_{i=1}^{n} c_i \mathscr{K}(x_i, \cdot)$ and $g(\cdot) = \sum_{i=1}^{n} c'_i \mathscr{K}(x'_i, \cdot)$. The corresponding norm is trivially

$$\|f\|^2_{\mathscr{K}} = \langle f, f \rangle_{\mathscr{K}}.$$

Then, Moore-Aronszajn theorem ensures that the space

$$\mathcal{H}(\mathscr{K}) := \overline{\mathcal{H}_0(\mathscr{K})}, \quad \|\cdot\|_{\mathcal{H}} := \|\cdot\|_{\mathscr{K}},$$

is an Hilbert space, and satisfies the reproducing kernel property, namely for every $f \in \mathcal{H}(\mathscr{K})$ and every $x \in \mathcal{X}$

$$f(x) = \langle f, \mathscr{K}(x, \cdot) \rangle_{\mathscr{K}}.$$

Moreover, no other space exists with these properties.

# C | Thompson Sampling on Multi-dimensional Action Spaces

The paper "On Thompson Sampling for Smoother-than-Lipschitz Bandits" [47] studies the performance of Thompson Sampling (TS) in continuum-armed bandits where the action space is continuous and the reward function is nonparametric but smooth. The main goal is to move beyond classical parametric settings (linear bandits, Gaussian processes) and provide regret guarantees for TS under weak structural assumptions on the reward function, specifically, when the function has Lipschitz continuous derivatives. This assumption corresponds exactly to the one done by [74] and, in the context of RL, by the Smooth MDPs. Their assumption on the reward boils down to being in $\mathrm{W}^{\nu,\infty}(\mathcal{A})$ for some $\nu > 0$.

The authors rely on the general framework developed by Russo and Van Roy [107], which shows that the Bayesian regret of Thompson Sampling can be controlled by a notion of sequential complexity of the function class: the Eluder dimension. In this sense, the paper is not really about Thompson Sampling per se, but about showing that, for sufficiently "simple" function classes, Thompson Sampling automatically explores well enough. Once the Eluder dimension is small, good regret bounds essentially follow as a black box.

From this perspective, the contribution of the paper boils down to the following question: can we show that $\mathrm{W}^{\nu,\infty}(\mathcal{A})$ has low Eluder dimension? If the answer is yes, then Thompson Sampling is provably efficient without any explicit discretization or optimism.

The authors give a positive answer to this question in a very specific setting. They show that for functions defined on a one-dimensional domain, i.e. $\mathcal{A} = [-1, 1]$, smoothness severely restricts how "independent" different actions can be. Intuitively, if a function is smooth and has a large value at some point, then it must also be large on a whole neighborhood around that point. This limits the length of $\varepsilon-$independent sequences and, as a result, the eluder dimension.

However, this is also where the main limitation appears. The argument is inherently one-dimensional. It relies on ordering points along a line and on the fact that neighborhoods are intervals whose size can be directly related to smoothness. As soon as the action space becomes multidimensional, this intuition breaks down. In higher dimensions, a function can remain highly localized while still being smooth, and controlling the overlap of these "large-value regions" becomes substantially harder. At this point, a natural question is left open by the paper: is the restriction to one-dimensional action spaces merely technical, or does it reflect a genuine limitation of the eluder-dimension approach?

As a side effect of the work on comparing Smooth MDPs with general function approximation, in this thesis we extended the Eluder-dimension analysis of smooth function classes to arbitrary-dimensional action spaces, providing both upper and lower bounds for the eluder dimension of smooth nonparametric functions on $[-1, 1]^m$. In fact, theorems 32 and 31 show that calling $\mathcal{F}$ a ball in the space $\mathrm{W}^{\nu,\infty}$ with norm constrained by $B$ and ignoring constant and logarithmic terms

$$\dim_{\mathrm{E}}(\mathcal{F}, \varepsilon) \asymp B^{m/\nu} \varepsilon^{-m/\nu}.$$

This directly generalizes Theorem 3 form [47], which provided the upper bound in case $m = 1$, and completely removes the need for having just one dimension. In this way, we show that smoothness alone is sufficient to control the sequential complexity of the problem even in high dimension.

Overall, this generalization significantly strengthens the theoretical foundations of Thompson Sampling in continuous domains. While the original work already provides valuable insights and introduces the right conceptual framework, extending the eluder-dimension analysis to multiple dimensions is crucial given that most practical applications of Thompson Sampling involve inherently high-dimensional action spaces [46]. In this sense, the present result should be seen not as a correction, but as a natural completion of an approach that was already pointing in the right direction.

# D | Details of the numerical simulations

To ensure reproducibility, we report here the details of the numerical simulation done in 6.3.1. These are important to ensure the truthfulness of the results and the claims based on empirical validation.

**Training Details.** The algorithms were implemented in PYTHON3.7. Each experiment was executed using five random seeds (corresponding to the first five natural numbers), and the computations were distributed across five parallel processes using the JOBLIB library. The total computational time for the first experiment was of 189935 seconds, more than two days and four hours.

**Compute.** We used a server with the following specifications:

- CPU: 88 INTEL(R) XEON(R) CPU E7-8880 V4 @ 2.20GHZ CPUS,
- RAM: 94,0 GB.

As mentioned, we parallelized the computing for the five different random seeds, therefore only five of the 88 cores were actually used. In fact, algorithm 6.2 cannot be parallelized in an efficient way without breaking the structure which provides the theoreticl guarantees.

# E | Missing Proofs

This section provides the complete proofs for several statements that were omitted from the main text to ensure a smoother flow of the exposition.

## E.1. Proofs from Chapter 3

**Proposition 9.** *Assume RegretBound*$(\mathscr{A}|\mathscr{M}, K, \delta) \leq C(\mathscr{M}, \delta)K^{\alpha}$*, for some* $\alpha < 1$*, where* $\mathscr{A}$ *is an algorithm and* $\mathscr{M}$ *an MDP class. Defined*

$$\overline{\pi}^K(\cdot|s) := K^{-1}\sum_{k=1}^{K}\pi^k(\cdot|s)$$

*the average policy played by* $\mathscr{A}$ *on all the episodes. Then, outputting* $\overline{\pi}^K(\cdot|s)$ *at the end of the* $K$ *episodes of interaction,* $\mathscr{A}$ *enjoys a sample complexity bound of the following form*

$$\mathit{SampleComplexity}(\mathscr{A}|\mathscr{M}, \varepsilon, \delta) \leq \left\lceil H\left(\frac{2C(\mathscr{M}, \delta/2)}{\varepsilon}\right)^{\frac{1}{1-\alpha}} + \frac{32H^3\log(4/\delta)}{\varepsilon^2}\right\rceil.$$

*Proof.* Fix $\mathcal{M} \in \mathscr{M}$ and $\delta > 0$. By assumption, w.p. at least $1 - \delta/2$, the regret bound of $\mathscr{A}$ satisfies

$$R_K \leq \text{RegretBound}(\mathscr{A}|\mathscr{M}, K, \delta/2).$$

We can write

$$R_K = M_K + KJ^\star - \sum_{k=1}^{K} J^{\pi^k},$$

where

$$M_K := \sum_{k=1}^{K} V_1^\star(s_1^k) - J^\star - V_1^{\pi^k}(s_1^k) + J^{\pi^k}.$$

Here, $M_k$ forms a martingale with bounded increments. Indeed, $V_1^{\pi^k}$ is $\mathcal{F}_{k-1}-$measurable $\mathbb{E}[V_1^\star(s_1^k)|\mathcal{F}_{k-1}] = J^{\pi^k}$ and

$$\mathbb{E}[M_k|\mathcal{F}_{k-1}] = \mathbb{E}[M_{k-1}|\mathcal{F}_{k-1}] + \underbrace{\mathbb{E}\left[V_1^\star(s_1^k) - J^\star - V_1^{\pi^k}(s_1^k) + J^{\pi^k}|\mathcal{F}_{k-1}\right]}_{=0}.$$

Moreover, by assumption 6, $V_1^\star(s_1^k) - J^\star - V_1^{\pi^k}(s_1^k) + J^{\pi^k}$ is always in $[-H, H]$, which means it is also $2H-$subgaussian. So, proposition 8 ensures that, with probability at least $1 - \delta/2$,

$$|M_K| \leq 2\sqrt{2K\log(4/\delta)}H.$$

This proves that

$$KJ^\star - \sum_{k=1}^{K} J^{\pi^k} \leq \text{RegretBound}(\mathscr{A}|\mathscr{M}, K, \delta/2) + 2\sqrt{2K\log(4/\delta)}H.$$

By construction and linearity of the expectation, the return of policy $\overline{\pi}^K$ satisfies $J^{\overline{\pi}^K} = \frac{1}{K}\sum_{k=1}^{K} J^{\pi^k}$. Thus,

$$J^\star - J^{\overline{\pi}^K} = J^\star - \frac{1}{K}\sum_{k=1}^{K} J^{\pi^k} \leq \frac{\text{RegretBound}(\mathscr{A}|\mathscr{M}, K, \delta/2) + 2\sqrt{2K\log(4/\delta)}H}{K},$$

where the all this holds with probability at least $1 - \delta$. For fixed $\varepsilon > 0$, turning this into a sample complexity guarantee requires

$$\frac{\text{RegretBound}(\mathscr{A}|\mathscr{M}, K, \delta/2) + 2\sqrt{2K\log(4/\delta)}H}{K} \leq \varepsilon.$$

Using the assumption that $\text{RegretBound}(\mathscr{A}|\mathscr{M}, K, \delta) \leq \text{C}(\mathscr{M}, \delta)K^\alpha$, the previous is satisfied by the following value $K_{\min}$ for the number of episodes

$$K_{\min} = \left\lceil H\left(\frac{2\text{C}(\mathscr{M}, \delta/2)}{\varepsilon}\right)^{\frac{1}{1-\alpha}} + \frac{32H^3\log(4/\delta)}{\varepsilon^2}\right\rceil.$$

Since every episode $k$ the agent collects $H$ samples, this number must be multiplied by this constant. Taking the supremum over $\mathcal{M}$ in $\mathscr{M}$ gives the proof. □

**Proposition 52.** *For every $h \in [H]$ and every $f : \mathcal{S} \to [0, 1]$ there is $\omega_h \in \mathbb{R}^d$ such that*

$$\forall s \in \mathcal{S}, a \in \mathcal{A} \qquad \int_{\mathcal{S}} f(s')p_h(s'|s, a)ds' = \varphi_h(s, a)^\top \omega_h.$$

*Proof.* 1. assumption A of [54] $\implies$ definition 8 point 2. Let $f : \mathcal{S} \to [0, 1]$ and

$h \in [H]$. Then,

$$\int_{\mathcal{S}} f(s')p_h(s'|s,a)\mathrm{d}s' = \int_{\mathcal{S}} f(s')\varphi_h(s,a)^\top \mu_h(s')\mathrm{d}s'$$
$$= \varphi_h(s,a)^\top \underbrace{\int_{\mathcal{S}} f(s')\mu_h(s')\mathrm{d}s'}_{\omega_h}.$$

2. definition 8 point 2 $\implies$ assumption A of [54].

□

**Proposition 35.** *Let $\mathcal{D}$ be built as before for $\mathcal{V} \subseteq \mathbb{R}^d$ and, for each $v_t \in \mathcal{D}$, $y_t = v_t^\top \theta + \eta_t$, for some unknown $\theta \in \mathbb{R}^d$, and a random noise $\{\eta_t\}_{t=1}^{|\mathcal{D}|}$ that is independent and $\sigma-$subgaussianconditionally on $\mathcal{D}$. Define the LS estimator as*

$$\widehat{\theta} := G_n^{-1} \sum_{t=1}^{|\mathcal{D}|} v_t y_t \qquad G_n := \sum_{t=1}^{|\mathcal{D}|} v_t v_t^\top.$$

*Then, for every $\delta > 0$, fixing an arbitrary subset $\mathcal{V}' \subset \mathcal{V}$ with $|\mathcal{V}'| = k$,*

$$\mathbb{P}\left( \sup_{v \in \mathcal{V}'} |v^\top \theta - v^\top \widehat{\theta}| > \sqrt{\frac{2d\log(2k/\delta)}{n}}\sigma \right) \leq \delta.$$

*Proof.* The statement is a special case of Proposition 12 by [80]. That result shows, in our notation that

$$\mathbb{P}\left( \sup_{v \in \mathcal{X}'} |v^\top \theta - v^\top \widehat{\theta}| > \sqrt{\log(2k/\delta)} \sup_{v \in \mathcal{V}} \|v\|_{G_n^{-1}} \sigma \right) \leq \delta.$$

Indeed, for a dataset $\mathcal{D}$ built according to our assumption, one has, for any $v \in \mathcal{V}$,

$$\|v\|_{G_n^{-1}} \leq \sqrt{n^{-1}\|v\|_{G^{-1}}^2} \leq \sqrt{d/n},$$

by the optimal design assumption. □

## E.2. Proofs from Chapter 5

**Proposition 38.** *Fix $\delta > 0$ and $\mathcal{Z}' \subset \mathcal{Z}$ with $|\mathcal{Z}'| = k$. Then, with probability at least $1-\delta$,*

$$\sup_{z\in\mathcal{Z}'} \left| \mathscr{V}_N * \mathcal{T}_h \widehat{Q}_{h+1}(z) - \widehat{Q}_h(z) \right| \leq \sqrt{\frac{2d\log(2k/\delta)}{n_h}}(H+1+\varepsilon_{h+1}).$$

*Proof.* By design of algorithm 5.2, $\widehat{Q}_h(z) = \varphi_N(z)^\top \widehat{\theta}_h$, where $\widehat{\theta}_h$ is chosen as the LS solution of a linear regression problem on data $\{\varphi_N(z_h^t), q_h^t\}_t$, where

$$q_h^t = \left[\beta^+ y_h^{t+} - \beta^- y_h^{t-}\right] + \left[\beta^+ \widehat{V}_{h+1}(s_{h+1}^{t+}) - \beta^- \widehat{V}_{h+1}(s_{h+1}^{t-})\right].$$

The first term satisfies

$$\begin{aligned}
\mathbb{E}\left[\beta^+ y_h^{t+} - \beta^- y_h^{t-} \middle| z_h^t\right] &= \mathbb{E}\left[\beta^+ r_h(z_h^t + \omega^+) | z_h^t\right] - \mathbb{E}\left[\beta^- r_h(z_h^t + \omega^-) \middle| z_h^t\right] \\
&= \beta^+ \mathbb{E}\left[r_h(z_h^t + \omega^+) | z_h^t\right] - \beta^- \mathbb{E}\left[r_h(z_h^t + \omega^-) \middle| z_h^t\right] \\
&= \beta^+ \mathscr{V}_N^+ * r_h(z_h^t) - \beta^- \mathscr{V}_N^- * r_h(z_h^t) \\
&= \mathscr{V}_N * r_h(z_h^t),
\end{aligned}$$

where the second equality comes from the definition of $\omega^+$ and $\omega^-$ and the third one by equations 5.4 and 5.5. Similarly, the second one satisfies

$$\begin{aligned}
\mathbb{E}\left[\beta^+ \widehat{V}_{h+1}(s_{h+1}^{t+}) - \beta^- \widehat{V}_{h+1}(s_{h+1}^{t-}) \middle| z_h^t\right] &= \mathbb{E}\left[\beta^+ \int_{\mathcal{Z}} \widehat{V}_{h+1}(s') p_h(s' | z_h^t + \omega^+) \mathrm{d}s' \middle| z_h^t\right] \\
&\quad - \mathbb{E}\left[\beta^- \int_{\mathcal{Z}} \widehat{V}_{h+1}(s') p_h(s' | z_h^t + \omega^-) \mathrm{d}s' \middle| z_h^t\right] \\
&= \beta^+ \int_{\mathcal{Z}} \widehat{V}_{h+1}(s') [\mathscr{V}_N^+ * p_h](s' | z_h^t) \mathrm{d}s' \\
&\quad - \beta^- \int_{\mathcal{Z}} \widehat{V}_{h+1}(s') [\mathscr{V}_N^- * p_h](s' | z_h^t) \mathrm{d}s' \\
&= \int_{\mathcal{Z}} \widehat{V}_{h+1}(s') [\mathscr{V}_N * p_h](s' | z_h^t) \mathrm{d}s'.
\end{aligned}$$

Together, they imply

$$\mathbb{E}[q_h^t | z_h^t] = \mathscr{V}_N * \mathcal{T}_h \widehat{Q}_{h+1}(z_h^t).$$

Also, from the fact that $r_h$ is bounded in $[0,1]$ and $1-$subgaussian and $\widehat{V}_h$ is bounded in $[-H+h-\varepsilon_{h+1}, H-h+\varepsilon_{h+1}]$, we have that $q_h^t$, conditionally on $z_h^t$, is $\sigma-$subgaussian for $\sigma \leq (H+1+\varepsilon_{h+1})$. All the assumptions are satisfied to apply proposition 35:

- $\mathcal{V} = \{\varphi_N(z)\}_{z\in\mathcal{Z}}$

- $y_t := q_h^t$, conditionally on $z_t = z_h^t$ have mean $\mathscr{V}_N * \mathcal{T}_h \widehat{Q}_{h+1}(z_h^t)$ that is a trigonometric polynomial of degree not larger than $N$. So, we can write it as $\varphi_N(z_t)^\top \theta$ for some $\theta \in \mathbb{R}^d$.
- The noise $y_t - \varphi_N(z_t)^\top \theta$ is $\sigma-$subgaussian with $\sigma \leq H + 1 + \varepsilon_{h+1}$.
- Dataset $\mathcal{D}_h$ is built as in section 5.2.1 for $n = n_h$

The thesis writes as

$$\mathbb{P}\left(\sup_{z\in\mathcal{Z}'}\left|\mathscr{V}_N * \mathcal{T}_h\widehat{Q}_{h+1}(z) - \varphi_N(z)^\top\widehat{\theta}_h\right| > \sqrt{\frac{2d\log(2k/\delta)}{n}}(H+1+\varepsilon_{h+1})\right) \leq \delta.$$

Replacing $\widehat{Q}_h = \varphi_N(z)^\top\widehat{\theta}_h$ completes the proof. □

## E.3. Proofs from Chapter 6

In this section, we show the complete proof of the regret bound for CINDERELLA. As in the original paper [78], we are performing the proofs for $d_h$ that may differ for different stages $h$ of the process. The case of $d_h = d$ follows immediately.

### E.3.1. Definition of the Quasi-optimal Solution

In this subsection, we create what we will call a quasi-optimal solution for the MDP. Define $\boldsymbol{\theta}_h^\star$ in the following recursive way

$$\theta_{h,n}^\star := \operatorname*{arg\,min}_{\theta_n\in\mathcal{B}_{h,n}} \max_{z\in\mathcal{Z}_{h,n}} |\phi_h(z)^\top\theta_n - \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}^\star](z)|, \tag{E.1}$$

So that $\boldsymbol{\theta}_h^\star$ is constructed by stacking these $\theta_{h,n}^\star$ terms. We have the following result.

**Theorem 46.** *The approximately optimal $Q-$function satisfies*

$$\forall h = 1,\dots,H \qquad \|Q[\boldsymbol{\theta}_h^\star](\cdot) - Q_h^\star(\cdot)\|_{L^\infty} \leq (H-h)\mathcal{I}.$$

*Proof.* We perform the proof by induction, with the case $h = H + 1$ being trivial.

$$\begin{aligned}
\|Q[\boldsymbol{\theta}^\star_h](\cdot) - Q^\star_h(\cdot)\|_{L^\infty} &= \left\|Q[\boldsymbol{\theta}^\star_h](\cdot) - \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot) + \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot) - Q^\star_h(\cdot)\right\|_{L^\infty} \\
&= \left\|Q[\boldsymbol{\theta}^\star_h](\cdot) - \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot) + \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot) - \mathcal{T}_h Q^\star_{h+1}(\cdot)\right\|_{L^\infty} \\
&\leq \left\|Q[\boldsymbol{\theta}^\star_h](\cdot) - \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot)\right\|_{L^\infty} \\
&\quad + \left\|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot) - \mathcal{T}_h Q^\star_{h+1}(\cdot)\right\|_{L^\infty} \\
&\leq \left\|Q[\boldsymbol{\theta}^\star_h](\cdot) - \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot)\right\|_{L^\infty} + \left\|Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot) - Q^\star_{h+1}(\cdot)\right\|_{L^\infty},
\end{aligned}$$

where in the last passage we have used the non-expansivity of the Bellman operator. At this point, the second part is bounded as

$$\|Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot) - Q^\star_{h+1}(\cdot)\|_{L^\infty} \leq (H - h - 1)\mathcal{I},$$

by inductive hypothesis, while the first one satisfies, by definition of inherent Bellman error and (E.1)

$$\|Q[\boldsymbol{\theta}^\star_h](\cdot) - \mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](\cdot)\|_{L^\infty} \leq \mathcal{I}.$$

Combining the two results gives the thesis. □

### E.3.2. Algorithm Analysis

We will now continue our work by proving that CINDERELLA is able to achieve a regret guarantee for this setting. First, we set up some additional notation.

**Additional notation** Collecting the solution of the optimization algorithm in the variables $\overline{\boldsymbol{\theta}}^k_h$, $\widehat{\boldsymbol{\theta}}^k_h$ and $\overline{\boldsymbol{\xi}}^k_h$, we define

$$\overline{Q}^k_h(z) := Q_h[\overline{\boldsymbol{\theta}}^k_h](z), \qquad \overline{V}^k_h(s) := \max_{a \in \mathcal{A}} \overline{Q}^k_h(s, a).$$

Moreover, in the rest of the section, once fixed a function $V : \mathcal{S} \to [0, 1]$, we define the Bellman error at step $h$ of episode $k$ as

$$\eta^k_h(V) := r^k_h - r_h(s^k_h, a^k_h) + V(s^k_{h+1}) - \mathop{\mathbb{E}}_{s' \sim p_h(\cdot|s^k_h, a^k_h)}[V(s')].$$

Furthermore, given a function $Q_{h+1} \in \mathcal{Q}_{h+1}$, we define

$$\mathring{\theta}_{h,n}(Q_{h+1}) := \arg\min_{\theta_n \in \mathcal{B}_{h,n}} \max_{z \in \mathcal{Z}_{h,n}} |\phi_h(z)^\top \theta_n - \mathcal{T}[Q_{h+1}](z)|, \tag{E.2}$$

and, as before, we collect all these vectors in $\mathring{\boldsymbol{\theta}}_h(Q_{h+1})$. Finally, we call

$$\Delta_h(Q_{h+1})(z) := \mathcal{T}_h Q_{h+1}(z) - Q[\mathring{\boldsymbol{\theta}}_h(Q_{h+1})](z).$$

### E.3.3. Decomposition of the Estimated Solution

We start by proving a proposition which establishes a relation between the variables of the optimization problem (6.7).

**Proposition 53.** *Let $\{\xi^k_{h,n}, \overline{\theta}^k_{h,n}, \widehat{\theta}^k_{h,n}\}_{h,n}$ be in the feasible region of Problem (6.7) at episode $k$ of the process. We have*

$$\overline{\theta}^k_{h,n} = \overline{\xi}^k_{h,n} + \mathring{\theta}_{h,n}(\overline{Q}^k_{h+1}) + \underbrace{{\Lambda^k_{h,n}}^{-1} \sum_{\tau=1}^{k-1} \mathbf{1}\{\rho_h(z^\tau_h) = n\} \phi^\tau_h \Delta(\overline{Q}^k_{h+1})(z^\tau_h)}_{T1^k_{h,n}} \tag{E.3}$$

$$- \underbrace{{\Lambda^k_{h,n}}^{-1} \mathring{\theta}_{h,n}(\overline{Q}^k_{h+1})}_{T2^k_{h,n}} + \underbrace{{\Lambda^k_{h,n}}^{-1} \sum_{t=1}^{k-1} \mathbf{1}\{\rho_h(z^\tau_h) = n\} \phi^\tau_h \eta^\tau_h(\overline{V}_{h+1})}_{T3^k_{h,n}} \tag{E.4}$$

*Proof.* For simplicity, let us abbreviate $\rho^k_{h,n} := \mathbf{1}\{\rho_h(z^k_h) = n\}$. By construction,

$$\begin{aligned}
\widehat{\theta}^k_{h,n} &= {\Lambda^k_{h,n}}^{-1} \sum_{t=1}^{k-1} \rho^\tau_{h,n} \phi^\tau_h (r^\tau_h + \overline{V}^k_{h+1}(s^\tau_{h+1})) \\
&= {\Lambda^k_{h,n}}^{-1} \sum_{\tau=1}^{k-1} \rho^\tau_{h,n} \phi^\tau_h \left( r_h(z^\tau_h) + \mathop{\mathbb{E}}_{s' \sim p_h(\cdot|z^\tau_h)} [\overline{V}^k_{h+1}(s')] \right) + \underbrace{{\Lambda^k_{h,i}}^{-1} \sum_{t=1}^{k-1} \rho^\tau_h \phi^\tau_h \eta^k_h(\overline{V}^k_{h+1}(s^\tau_{h+1}))}_{T3^k_{h,n}} \\
&= {\Lambda^k_{h,n}}^{-1} \sum_{\tau=1}^{k-1} \rho^\tau_{h,n} \phi^\tau_h \mathcal{T}_h[\overline{Q}^k_{h+1}](z^\tau_h) + T3^k_{h,n},
\end{aligned}$$

where we have used the definition of Bellman's optimality operator. By definition, we have

$$\mathcal{T}_h[\overline{Q}^k_{h+1}](z^\tau_h) = Q[\overset{\circ}{\boldsymbol{\theta}}_h(\overline{Q}^k_{h+1})](z^\tau_h) + \Delta_h(\overline{Q}^k_{h+1})(z^\tau_h),$$

which, when multiplied by $\rho^\tau_{h,n}$ lets only the term corresponding to the region $n$ remain. Then, by Equation (E.2), we have

$$\rho^\tau_{h,n}\mathcal{T}_h[\overline{Q}^k_{h+1}](z^\tau_h) = \rho^\tau_{h,n}\phi_h(z^\tau_h)^\top \overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1}) + \rho^\tau_{h,n}\Delta_h(\overline{Q}^k_{h+1})(z^\tau_h).$$

Using this fact, we have

$$\begin{aligned}
{\Lambda^k_{h,n}}^{-1}\sum_{\tau=1}^{k-1}\rho^\tau_{h,n}\phi^\tau_h\mathcal{T}_h[\overline{Q}^k_{h+1}](z^\tau_h) &= {\Lambda^k_{h,n}}^{-1}\sum_{\tau=1}^{k-1}\rho^\tau_{h,n}\phi^\tau_h\left({\phi^\tau_h}^\top\overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1}) + \Delta_h(\overline{Q}^k_{h+1})(z^\tau_h)\right)\\
&= {\Lambda^k_{h,n}}^{-1}\sum_{\tau=1}^{k-1}\rho^\tau_{h,n}\phi^\tau_h{\phi^\tau_h}^\top\overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1}) + T1^k_{h,n}.
\end{aligned}$$

For the remaining term, note that

$$\begin{aligned}
{\Lambda^k_{h,n}}^{-1}\sum_{\tau=1}^{k-1}\rho^\tau_{h,n}\phi^\tau_h{\phi^\tau_h}^\top\overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1}) &= {\Lambda^k_{h,n}}^{-1}\left(-\lambda\overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1}) + \lambda\overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1}) + \sum_{\tau=1}^{k-1}\rho^\tau_{h,n}\phi^\tau_h{\phi^\tau_h}^\top\overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1})\right)\\
&= {\Lambda^k_{h,n}}^{-1}\left(-\lambda\overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1}) + \Lambda^k_{h,n}\overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1})\right)\\
&= \overset{\circ}{\theta}_{h,n}(\overline{Q}^k_{h+1}) - T2^k_{h,n}
\end{aligned}$$

□

The objective of the next lemmas is to show that the terms arising from the previous proposition are small with high probability.

**Lemma 1.** *For any $z \in \mathcal{Z}$, any time-step $h$ and any $n$, we have*

$$|\phi_h(z)^\top T1^k_{h,n}| \leq \|\phi_h(z)\|_{{\Lambda^k_{h,n}}^{-1}}\sqrt{p^k_{h,n}\mathcal{I}},$$

*where*

$$p^k_{h,n} := \sum_{\tau=1}^{k-1}\mathbf{1}\{\rho_h(z^\tau_h) = n\},$$

*and in particular*

$$\|T1^k_{h,n}\|_{\Lambda^k_{h,n}} \leq \sqrt{p^k_{h,n}}\mathcal{I}.$$

*Proof.* By definition,

$$\begin{aligned}
|\phi_h(z)^\top T1^k_{h,n}| &= \left|\phi_h(z)^\top {\Lambda^k_{h,n}}^{-1} \sum_{\tau=1}^{k-1} \mathbf{1}\{\rho_h(z^\tau_h) = n\}\phi^\tau_h \Delta(\overline{Q}^k_{h+1})(z^\tau_h)\right| \\
&\leq \|{\Lambda^k_{h,n}}^{-1}\phi_h(s,a)\|_{\Lambda^k_{h,n}} \left\|\sum_{\tau=1}^{k-1} \mathbf{1}\{\rho_h(z^\tau_h) = n\}\phi^\tau_h \Delta(\overline{Q}^k_{h+1})(z^\tau_h)\right\|_{{\Lambda^k_{h,n}}^{-1}} \\
&= \|\phi_h(s,a)\|_{{\Lambda^k_{h,n}}^{-1}} \left\|\sum_{\tau=1}^{k-1} \mathbf{1}\{\rho_h(z^\tau_h) = n\}\phi^\tau_h \Delta(\overline{Q}^k_{h+1})(z^\tau_h)\right\|_{{\Lambda^k_{h,n}}^{-1}}.
\end{aligned}$$

At this point, leave the first term and, in the second we rewrite the sum for the indices where the indicator function is not zero, getting

$$\left\|\sum_{\tau'=1}^{p^k_{h,n}} \phi^{\tau'}_h \Delta(\overline{Q}^k_{h+1})(z^{\tau'}_h)\right\|_{{\Lambda^k_{h,n}}^{-1}}.$$

We can then use Lemma 8 from [151], taking $a_i = \phi^{\tau'}_h$ and $b_i = \Delta(\overline{Q}^k_{h+1})(z^{\tau'}_h)$. As $|b_i| \leq \mathcal{I}$, by the low inherent Bellman error assumption, this gives

$$\left\|\sum_{\tau'=1}^{p^k_{h,n}} \phi^{\tau'}_h \Delta(\overline{Q}^k_{h+1})(z^{\tau'}_h)\right\|_{{\Lambda^k_{h,n}}^{-1}} \leq \sqrt{p^k_{h,n}}\mathcal{I},$$

which ends the proof of the first part. The second one comes from

$$\begin{aligned}
\|T1^k_{h,n}\|_{\Lambda^k_{h,n}} &= \left\|{\Lambda^k_{h,n}}^{-1} \sum_{\tau'=1}^{p^k_{h,n}} \phi^{\tau'}_h \Delta(\overline{Q}^k_{h+1})(z^{\tau'}_h)\right\|_{\Lambda^k_{h,n}} \\
&= \left\|\sum_{\tau'=1}^{p^k_{h,n}} \phi^{\tau'}_h \Delta(\overline{Q}^k_{h+1})(z^{\tau'}_h)\right\|_{{\Lambda^k_{h,n}}^{-1}} \\
&\leq \sqrt{p^k_{h,n}}\mathcal{I}.
\end{aligned}$$

□

We proceed bounding the second part of the sum.

**Lemma 2.** *For any $z \in \mathcal{Z}$, any time-step $h$ and any $n$, we have*

$$|\phi_h(z)^\top T2^k_{h,n}| \leq \|\phi_h(z)\|_{{\Lambda^k_{h,n}}^{-1}} \lambda^{-1} \mathcal{R}_{h,n},$$

*and, in particular*

$$\|T2^k_{h,n}\|_{\Lambda^k_{h,n}} \leq \lambda^{-1} \mathcal{R}_{h,n}.$$

*Proof.* By definition,

$$\begin{aligned}
|\phi_h(z)^\top T2^k_{h,n}| &= \left|\phi_h(z)^\top {\Lambda^k_{h,n}}^{-1} \mathring{\theta}_{h,n}(\overline{Q}^k_{h+1})\right| \\
&\leq \|{\Lambda^k_{h,n}}^{-1} \phi_h(z)\|_{\Lambda^k_{h,n}} \|\mathring{\theta}_{h,n}(\overline{Q}^k_{h+1})\|_{{\Lambda^k_{h,n}}^{-1}} \\
&= \|\phi_h(z)\|_{{\Lambda^k_{h,n}}^{-1}} \|\mathring{\theta}_{h,n}(\overline{Q}^k_{h+1})\|_{{\Lambda^k_{h,n}}^{-1}} \\
&\leq \|\phi_h(z)\|_{{\Lambda^k_{h,n}}^{-1}} \lambda^{-1} \|\mathring{\theta}_{h,n}(\overline{Q}^k_{h+1})\|_2 \\
&\leq \|\phi_h(z)\|_{{\Lambda^k_{h,n}}^{-1}} \lambda^{-1} \mathcal{R}_{h,n},
\end{aligned}$$

where the second inequality comes from the fact that $\Lambda^k_{h,n}$ is the sum of $\lambda I$ and a positive semi-definite matrix, and the last one from the fact that $\mathring{\theta}_{h,n}(\overline{Q}^k_{h+1}) \in \mathcal{B}_{h,n}$. The second part comes from

$$\|T2^k_{h,n}\|_{\Lambda^k_{h,n}} = \|{\Lambda^k_{h,n}}^{-1} \mathring{\theta}_{h,n}(\overline{Q}^k_{h+1})\|_{\Lambda^k_{h,n}} = \|\mathring{\theta}_{h,n}(\overline{Q}^k_{h+1})\|_{{\Lambda^k_{h,n}}^{-1}} \leq \lambda^{-1} \mathcal{R}_{h,n}.$$

□

The last part of the sum is more complex and, in order to bound it, we need to define a failure event.

### E.3.4. Failure Event

For every $h, k, n$ we define the failure event in the following way:

$$F^k_{h,n} := \left\{ \exists V \in \mathcal{V}_h : \left\| \sum_{t=1}^{k-1} \mathbf{1}\{\rho_h(z^\tau_h) = n\} \phi^\tau_h \eta^\tau_h(V) \right\|_{{\Lambda^k_{h,n}}^{-1}} > \sqrt{\beta^k_{h,n}} \right\},$$

for a threshold $\sqrt{\beta_{h,n}^k}$ to be defined. The first step to bound the probability of this event is to compute the covering number of the function space $\mathcal{V}_h$

**Proposition 54.** *The $\varepsilon-$covering number of $\mathcal{V}_h$ in infinity norm satisfies*

$$\log \mathcal{N}(\varepsilon, \mathcal{V}_h) \leq \mathcal{O}\left(Nd \log\left(\mathcal{R}_{h,\max}/\varepsilon\right)\right),$$

*where* $\mathcal{R}_{h,\max} := \sup_{n\in[N_h]} \mathcal{R}_{h,n}$.

*Proof.* Note that the covering number of $\mathcal{V}_h$ is not higher than the one of $\mathcal{Q}_h$. Indeed, let $\mathcal{Q}_h^\varepsilon$ be a $\varepsilon-$cover for $\mathcal{Q}_h$ and define

$$\mathcal{V}_h^\varepsilon := \{V(s) = \max_{a\in\mathcal{A}} Q(s,a) : Q \in \mathcal{Q}_h^\varepsilon\}.$$

Then, taking any $V \in \mathcal{V}_h$ there is $Q \in \mathcal{Q}_h$ such that $V(s) = \max_{a\in\mathcal{A}} Q(s,a)$. At this point, taking

$$\widehat{V}(s) = \max_{a\in\mathcal{A}} \widehat{Q}(s,a),$$

where $\widehat{Q} \in \mathcal{Q}_h^\varepsilon$ such that $\|Q - \widehat{Q}\|_\infty \leq \varepsilon$, we have that $\widehat{V} \in \mathcal{V}_h^\varepsilon$ by definition and

$$\|Q - \widehat{Q}\|_\infty \leq \|V - \widehat{V}\|_\infty \leq \varepsilon.$$

Now, the question is reduced to covering $\mathcal{Q}_h$. by definition, every $Q_h \in \mathcal{Q}_h$ takes the form

$$Q_h(z) = \phi_h(z)^\top \theta_{h,\rho_h(z)},$$

where $\rho_h : \mathcal{Z} \to [N_h]$ and every vector $\theta_{h,n}$ is $d_h-$dimensional with the norm bounded by $\mathcal{R}_{h,n}$. As the domain $\mathcal{Z}$ is partitioned into regions $\mathcal{Z}_{h,n}$, if we get a family of coverings $\mathcal{Q}_{h,n}^\varepsilon$ that cover each function in $\mathcal{Q}_h$ restricted to the set $\mathcal{Z}_{h,n}$, we can obtain a covering of $\mathcal{Q}_h$ by defining it in this way

$$\mathcal{Q}_h^\varepsilon := \left\{Q : Q|_{\mathcal{Z}_{h,n}} = Q_n \qquad Q_n \in \mathcal{Q}_{h,n}^\varepsilon\right\},$$

i.e., we cover every function $Q$ by taking the nearest cover function on each region. As a result, the total number of elements in $\mathcal{Q}_h^\varepsilon$ corresponds to the product $\prod_{n=1}^{N_h} |\mathcal{Q}_{h,n}^\varepsilon|$ of the elements in the smaller sets, so that we have only to estimate $|\mathcal{Q}_{h,n}^\varepsilon|$ for every $n$. To this aim, note that the functions in this set are all linear, as we are restricting to $\mathcal{Z}_{h,n}$, so

that, applying a standard bound for the covering number of spaces of linear functions we can find $\mathcal{Q}^{\varepsilon}_{h,n}$ such that

$$|\mathcal{Q}^{\varepsilon}_{h,n}| = \mathcal{O}\left((\mathcal{R}_{h,n}/\varepsilon)^{d_h}\right).$$

Therefore, the total covering size amounts to

$$|\mathcal{Q}^{\varepsilon}_{h}| = \mathcal{O}\left(\left(\sup_{n\in[N_h]} \mathcal{R}_{h,n}/\varepsilon\right)^{Nd}\right),$$

which completes the proof. □

The former result allows to upper bound the probability of the failure event.

**Theorem 55.** *For a choice of*

$$\sqrt{\beta^k_{h,n}} = \widetilde{\mathcal{O}}\left((H+\sigma)\sqrt{d_h + \log(\mathcal{N}(1/\sqrt{k}, \mathcal{V}_h)) + \log(1/\delta)}\right),$$

*we have* $\mathbb{P}(F^k_{h,n}) \leq \delta$.

*Proof.* Let $\mathcal{V}^{\varepsilon}_h$ be a $\varepsilon$ cover of $\mathcal{V}_h$ in infinity norm, so that $|\mathcal{V}^{\varepsilon}_h| = \mathcal{N}(\varepsilon, \mathcal{V}_h)$. Now, for every $V \in \mathcal{V}_h$ we call

$$\widetilde{V} \in \mathcal{V}^{\varepsilon}_h : \ \|\widetilde{V} - V\|_{L^\infty} \leq \varepsilon.$$

In this way, we have

$$\begin{aligned}
\left\|\sum_{t=1}^{k-1} \rho^{\tau}_{h,n}\phi^{\tau}_h\eta^{\tau}_h(V)\right\|_{{\Lambda^k_{h,n}}^{-1}} &= \left\|\sum_{t=1}^{k-1} \rho^{\tau}_{h,n}\phi^{\tau}_h\left(r^k_h - r_h(s^k_h,a^k_h) + V(s^k_{h+1}) - \mathop{\mathbb{E}}_{s'\sim p_h(\cdot|s^k_h,a^k_h)}[V(s')]\right)\right\|_{{\Lambda^k_{h,n}}^{-1}} \\
&\leq \left\|\sum_{t=1}^{k-1} \rho^{\tau}_{h,n}\phi^{\tau}_h\left(r^k_h - r_h(s^k_h,a^k_h) + \widetilde{V}(s^k_{h+1}) - \mathop{\mathbb{E}}_{s'\sim p_h(\cdot|s^k_h,a^k_h)}[\widetilde{V}(s')]\right)\right\|_{{\Lambda^k_{h,n}}^{-1}} \\
&\quad + \left\|\sum_{t=1}^{k-1} \rho^{\tau}_{h,n}\phi^{\tau}_h\left(V(s^k_{h+1}) - \widetilde{V}(s^k_{h+1})\right)\right\|_{{\Lambda^k_{h,n}}^{-1}} \\
&\quad + \left\|\sum_{t=1}^{k-1} \rho^{\tau}_{h,n}\phi^{\tau}_h\left(\mathop{\mathbb{E}}_{s'\sim p_h(\cdot|s^k_h,a^k_h)}[\widetilde{V}(s') - V(s')]\right)\right\|_{{\Lambda^k_{h,n}}^{-1}}.
\end{aligned}$$

As we have

$$|V(s_{h+1}^k) - \widetilde{V}(s_{h+1}^k)| \leq \varepsilon, \qquad |\mathop{\mathbb{E}}_{s' \sim p_h(\cdot|s_h^k, a_h^k)}[\widetilde{V}(s') - V(s')]| \leq \varepsilon,$$

the same argument used in the proof Lemma 1, allows us to bound the last two terms with $2\sqrt{p_{h,n}^k}\varepsilon$. The remaining term corresponds, for fixed $\widetilde{V}$ and indicating with $\mathcal{F}_h^k$ the filtration generated by all the events of the process up to step $h$ of episode $k$, to

$$\left\| \sum_{t=1}^{k-1} \underbrace{\rho_{h,n}^\tau \phi_h^\tau}_{\mathcal{F}_h^k - meas.} \underbrace{\left( r_h^k - r_h(s_h^k, a_h^k) + \widetilde{V}(s_{h+1}^k) - \mathop{\mathbb{E}}_{s' \sim p_h(\cdot|s_h^k, a_h^k)}[\widetilde{V}(s')] \right)}_{\mathbb{E}[\cdot|\mathcal{F}_h^k]=0} \right\|_{{\Lambda_{h,n}^k}^{-1}},$$

where the first term is an $\mathbb{R}^{d_h}-$valued stochastic process which is entirely determined by the current state and current action, while the second is zero-mean conditioned on the rest of the process, and $(\sigma + H)-$subgaussian thanks to assumptions 7 and 6. This last fact allows to apply Theorem 1 from [2], obtaining with probability at least $1 - \delta$, for fixed $\widetilde{V}$,

$$\left\| \sum_{t=1}^{k-1} \rho_{h,n}^\tau \phi_h^\tau \left( r_h^k - r_h(s_h^k, a_h^k) + \widetilde{V}(s_{h+1}^k) - \mathop{\mathbb{E}}_{s' \sim p_h(\cdot|s_h^k, a_h^k)}[\widetilde{V}(s')] \right) \right\|^2_{{\Lambda_{h,n}^k}^{-1}}$$

$$\leq 2(H+\sigma) \log \left( \frac{\det \lambda I^{-1/2} \det {\Lambda_{h,n}^k}^{1/2}}{\delta} \right) \leq (H+\sigma)^2 \left( d_h \log(\lambda^{-1}) + d_h \log((1 + kL_\phi)) + \log(\delta^{-1}) \right).$$

Applying a union bound, it follows that the same result holds for every function in $\mathcal{V}_h^\varepsilon$ if we replace $\log(\delta^{-1})$ with $\log(\mathcal{N}(\varepsilon, \mathcal{V}_h)/\delta)$. Merging this result with what we have found before, we have that, with probability at least $1 - \delta$, for all $V \in \mathcal{V}_h$, we have

$$\left\| \sum_{t=1}^{k-1} \rho_{h,n}^\tau \phi_h^\tau \eta_h^\tau(V) \right\|_{{\Lambda_{h,n}^k}^{-1}} \leq (H+\sigma) \sqrt{d_h \log(\lambda^{-1}) + d_h \log((1 + kL_\phi)) + \log(\mathcal{N}(\varepsilon, \mathcal{V}_h)) + \log(1/\delta)}$$
$$+ 2\sqrt{p_{h,n}^k}\varepsilon.$$

If we just bound $\sqrt{p_{h,n}^k} \leq \sqrt{k}$, we can take $\varepsilon = 1/\sqrt{k}$ to have that the previous is bounded

by

$$\sqrt{\beta_{h,n}^k} := (H+\sigma)\sqrt{d_h \log(\lambda^{-1}) + d_h \log((1+kL_\phi)) + \log(\mathcal{N}(1/\sqrt{k}, \mathcal{V}_h)) + \log(1/\delta)} + 2,$$

which completes the proof. □

From this theorem, a simple corollary follows by just taking a union bound over $h \in [H], n \in [N_h], k \in [K]$:

**Corollary 56.** *For a choice of*

$$\sqrt{\beta_{h,n}^k} = \widetilde{\mathcal{O}}\left((H+\sigma)\sqrt{d_h + \log(\mathcal{N}(1/\sqrt{k}, \mathcal{V}_h)} + (H+\sigma)\log(N_h HK/\delta))\right),$$

*we have*

$$\mathbb{P}\left(\bigcup_{h,n,k} F_{h,n}^k\right) \leq \delta.$$

From now on, we are going to indicate the good event $E$ as the opposite of the event defined in the previous corollary

$$E := \bigcap_{h,n,k} (F_{h,n}^k)^c,$$

moreover, $\sqrt{\beta_{h,n}^k}$ will always be the quantity defined by Corollary 56. We conclude the section proving the following result.

**Lemma 3.** *Under the event $E$, for any $z \in \mathcal{Z}$, any time-step $h$ of episode $k$ and any $n$, we have*

$$|\phi_h(z)^\top T3_{h,n}^k| \leq \|\phi_h(z)\|_{{\Lambda_{h,n}^k}^{-1}} \sqrt{\beta_{h,n}^k},$$

*and, in particular*

$$\|T3_{h,n}^k\|_{{\Lambda_{h,n}^k}^{-1}} \leq \sqrt{\beta_{h,n}^k}.$$

*Proof.* By definition,

$$
\begin{aligned}
|\phi_h(z)^\top T3^k_{h,n}| &= \left|\phi_h(z)^\top {\Lambda^k_{h,n}}^{-1} \sum_{t=1}^{k-1} \mathbf{1}\{\rho_h(z_h^\tau) = n\}\phi_h^\tau \eta_h^\tau(\overline{V}_{h+1})\right| \\
&\leq \|{\Lambda^k_{h,n}}^{-1}\phi_h(z)\|_{\Lambda^k_{h,n}} \left\|\sum_{t=1}^{k-1} \mathbf{1}\{\rho_h(z_h^\tau) = n\}\phi_h^\tau \eta_h^\tau(\overline{V}_{h+1})\right\|_{{\Lambda^k_{h,n}}^{-1}} \\
&= \|\phi_h(z)\|_{{\Lambda^k_{h,n}}^{-1}} \left\|\sum_{t=1}^{k-1} \mathbf{1}\{\rho_h(z_h^\tau) = n\}\phi_h^\tau \eta_h^\tau(\overline{V}_{h+1})\right\|_{{\Lambda^k_{h,n}}^{-1}}.
\end{aligned}
$$

Under the good event $E$, we have that the second term is bounded by $\sqrt{\beta^k_{h,n}}$, for the choice defined in Corollary 56. The second part comes from the fact that

$$
\begin{aligned}
\|T3^k_{h,n}\|_{\Lambda^k_{h,n}} &= \left\|{\Lambda^k_{h,n}}^{-1} \sum_{t=1}^{k-1} \mathbf{1}\{\rho_h(z_h^\tau) = n\}\phi_h^\tau \eta_h^\tau(\overline{V}_{h+1})\right\|_{\Lambda^k_{h,n}} \\
&= \left\|\sum_{t=1}^{k-1} \mathbf{1}\{\rho_h(z_h^\tau) = n\}\phi_h^\tau \eta_h^\tau(\overline{V}_{h+1})\right\|_{{\Lambda^k_{h,n}}^{-1}} \\
&\leq \sqrt{\beta^k_{h,n}}.
\end{aligned}
$$

□

### E.3.5. Quasi-optimal solution and good event

Thanks to the previous result, we are able to show that the quasi-optimal solution is feasible at all episodes $k = 1, \dots K$ with high probability.

**Theorem 57.** *Consider the optimization problem* (6.7)*, where* $\sqrt{\alpha^k_{h,n}}$ *is set to* $\sqrt{\beta^k_{h,n}} + \sqrt{p^k_{h,n}} + \lambda^{-1}\mathcal{R}_{h,n}$*. Then, under the good event* $E$*, the quasi-optimal solution* $\{\boldsymbol{\theta}^\star_h\}_{h=1}^H$ *(see* (E.1)*) is feasible for* (6.7) *at any episode* $k$*.*

*Proof.* We perform the proof by induction on the time-step $h$, starting from $h = H$. Assuming that for every $n \in [N_h]$ the choice $\theta^\star_{h+1,n}$ is feasible we have that the choice

$$
\overline{Q}_{h+1}(z) := Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}](z)
$$

is also feasible. With this choice, proposition 53 ensures that, for every $n$,

$$\overline{\theta}_{h,n} = \overline{\xi}_{h,n} + \overset{\circ}{\theta}_{h,n}(Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}]) + T1^k_{h,n} - T2^k_{h,n} + T3^k_{h,n}. \tag{E.5}$$

Now, to show that also $\boldsymbol{\theta}^\star_h$ is feasible, we have to prove that, substituting each $\overline{\theta}_{h,n} = \theta^\star_{h,n}$ in the previous equation, we get a value for $\overline{\xi}_{h,n}$ such that $\|\overline{\xi}_{h,n}\|_{\Lambda^k_{h,n}} \leq \sqrt{\alpha^k_{h,n}}$.

First note that, by definition, $\theta^\star_{h,n} = \overset{\circ}{\theta}_{h,n}(Q_{h+1}[\boldsymbol{\theta}^\star_{h+1}])$, so that the previous equation simplifies to

$$-\overline{\xi}_{h,n} = T1^k_{h,n} - T2^k_{h,n} + T3^k_{h,n}. \tag{E.6}$$

At this point we have by triangular inequality

$$\begin{aligned}
\|\overline{\xi}_{h,n}\|_{\Lambda^k_{h,n}} &\leq \|T1^k_{h,n}\|_{\Lambda^k_{h,n}} + \|T2^k_{h,n}\|_{\Lambda^k_{h,n}} + \|T3^k_{h,n}\|_{\Lambda^k_{h,n}} && \text{(E.7)}\\
&\leq \sqrt{p^k_{h,n}}\mathcal{I} + \lambda^{-1}\mathcal{R}_{h,n} + \sqrt{\beta^k_{h,n}} && \text{(E.8)}\\
&= \sqrt{\alpha^k_{h,n}}. && \text{(E.9)}
\end{aligned}$$

where the second comes from Lemmas 1, 2 and 3. This completes the proves that each $\theta^\star_{h,n}$ is feasible, meaning that also $\boldsymbol{\theta}^\star_h$ is, and completes the inductive step. □

From the feasibility of $\boldsymbol{\theta}^\star_h$ a simple corollary follows.

**Corollary 58.** *Under the good event $E$, at each episode $k$, the $V-$function $\overline{V}^k_1(\cdot)$ estimated from $\overline{\boldsymbol{\theta}_1}$ as solution of solution of the optimization problem* (6.7) *satisfies*

$$\forall s \in \mathcal{S} \qquad V^\star_1(s) - \overline{V}^k_1(s) \leq (H-1)\mathcal{I}.$$

*Proof.* By design of the optimization problem, and the fact $\{\boldsymbol{\theta}^\star_h\}_{h=1}^H$ is feasible under $E$, we have

$$\overline{V}^k_1(s) \geq V[\boldsymbol{\theta}^\star_1](s).$$

Now,

$$\begin{aligned}
V^\star_1(s) - \overline{V}^k_1(s) &\leq V^\star_1(s) - V[\boldsymbol{\theta}^\star_1](s)\\
&= \max_{a\in\mathcal{A}} Q^\star_1(s,a) - \max_{a\in\mathcal{A}} Q[\boldsymbol{\theta}^\star_1](s,a)\\
&\leq \max_{a\in\mathcal{A}} Q^\star_1(s,a) - Q[\boldsymbol{\theta}^\star_1](s,a)\\
&\leq \|Q^\star_1(\cdot) - Q[\boldsymbol{\theta}^\star_1](\cdot)\|_{L^\infty} \leq (H-1)\mathcal{I},
\end{aligned}$$

where the last passage is possible by theorem 46. □

Moreover, we are also able to prove that the solution found by the optimization procedure (6.7) is an "almost fixed" point of the Bellman optimiality operator.

**Proposition 59.** *Under the good event $E$, the $Q-$function computed with the optimization algorithm at each step $k$ satisfies,*

$$\forall z \in \mathcal{Z} \qquad \left|\overline{Q}_h^k(z) - \mathcal{T}_h[\overline{Q}_{h+1}^k](z)\right| \leq \mathcal{I} + 2\sqrt{\alpha_{h,\rho_h(z)}^k}\|\phi_h(z)\|_{{\Lambda_{h,\rho_h(z)}^k}^{-1}}.$$

*Proof.* By Proposition 53, we have, taking $n := \rho_h(z)$,

$$\begin{aligned}
|\overline{Q}_h^k(z) - \mathcal{T}_h[\overline{Q}_{h+1}^k](z)| &\leq |\phi_h(z)^\top \overline{\theta}_{h,n} - \mathcal{T}_h[\overline{Q}_{h+1}](z)| \\
&\quad + \left|\phi_h(z)^\top(\overline{\xi}_{h,n} + T1_{h,n}^k - T2_{h,n}^k + T3_{h,n}^k)\right| \\
&\leq \mathcal{I} + \left|\phi_h(z)^\top \overline{\xi}_{h,n}\right| + \left|\phi_h(z)^\top T1_{h,n}^k\right| \\
&\quad + \left|\phi_h(z)^\top T2_{h,n}^k\right| + \left|\phi_h(z)^\top T3_{h,n}^k\right|.
\end{aligned}$$

The first term satisfies, by the constrain in the program,

$$\left|\phi_h(s,a)^\top \overline{\xi}_{h,n}\right| \leq \|\overline{\xi}_{h,n}\|_{\Lambda_{h,n}^k}\|\phi_h(s,a)\|_{{\Lambda_{h,n}^k}^{-1}} \leq \sqrt{\alpha_{h,n}^k}\|\phi_h(s,a)\|_{{\Lambda_{h,n}^k}^{-1}}.$$

The other terms satisfy, thanks to lemmas 1,2,3, satisfies

$$\left|\phi_h(z)^\top T1_{h,n}^k\right| + \left|\phi_h(z)^\top T2_{h,n}^k\right| + \left|\phi_h(z)^\top T3_{h,n}^k\right| \leq \sqrt{\alpha_{h,n}^k}\|\phi_h(s,a)\|_{{\Lambda_{h,n}^k}^{-1}}.$$

□

## E.4. Regret Bound

Using all the results that we have proved, we can finally bound the regret

**Theorem 60.** *Under event $E$, CINDERELLA, for $\lambda = 1$ achieves a regret bound of order*

$$R_K \leq \widetilde{\mathcal{O}}\left(\sum_{h=1}^{H}\sqrt{KN_h}\left((L_\phi + \sqrt{d_h})\sup_n \mathcal{R}_{h,n} + d_h + \sqrt{d_h \log(\mathcal{N}(1/\sqrt{K}, \mathcal{V}_h))}\right) + KH\mathcal{I}\sqrt{d_h}\right).$$

*Proof.* By Corollary 58, the regret is bounded by

$$R_K = \sum_{k=1}^{K} V_1^*(s_1^k) - V_1^{\pi_k}(s_1^k) \leq KH\mathcal{I} + \sum_{k=1}^{K} \overline{V}_1^k(s_1^k) - V_1^{\pi_k}(s_1^k).$$

Note that, under the good event,

$$\begin{aligned}
\overline{V}_h^k(s_h^k) - V_h^{\pi_k}(s_h^k) &= \overline{Q}_h^k(z_h^k) - V_h^{\pi_k}(s_h^k) \\
&\overset{\text{Prop. 59}}{\leq} \mathcal{I} + 2\sqrt{\alpha^k_{h,\rho_h(z_h^k)}}\|\phi_h(z_h^k)\|_{{\Lambda^k_{h,\rho_h(z_h^k)}}^{-1}} + \mathcal{T}_h[\overline{Q}_{h+1}^k](z_h^k) - V_h^{\pi_k}(s_h^k) \\
&= \mathcal{I} + 2\sqrt{\alpha^k_{h,\rho_h(z_h^k)}}\|\phi_h(z_h^k)\|_{{\Lambda^k_{h,\rho_h(z_h^k)}}^{-1}} + \mathop{\mathbb{E}}_{s'\sim p_h(\cdot|z_h^k)}[\overline{V}_{h+1}^k(s') - V_{h+1}^{\pi_k}(s')].
\end{aligned}$$

Now, if we define the quantity

$$\zeta_h^k := \mathop{\mathbb{E}}_{s'\sim p_h(\cdot|s_h^k,a_h^k)}[\overline{V}_{h+1}^k(s') - V_{h+1}^{\pi_k}(s')] - \overline{V}_{h+1}^k(s_{h+1}^k) + V_{h+1}^{\pi_k}(s_{h+1}^k),$$

we can rewrite the previous relation as a telescopic sum:

$$\begin{aligned}
\overline{V}_1^k(s_1^k) - V_1^{\pi_k}(s_1^k) &\leq \mathcal{I} + 2\sqrt{\alpha^k_{1,\rho_h(z_1^k)}}\|\phi_1(z_1^k)\|_{{\Lambda^k_{1,\rho_1(z_1^k)}}^{-1}} + \mathop{\mathbb{E}}_{s'\sim p_1(\cdot|z_1^k)}[\overline{V}_2^k(s') - V_2^{\pi_k}(s')] \\
&= \mathcal{I} + 2\sqrt{\alpha^k_{1,\rho_h(z_1^k)}}\|\phi_1(z_1^k)\|_{{\Lambda^k_{1,\rho_1(z_1^k)}}^{-1}} + \zeta_1^k + \underbrace{\overline{V}_2^k(s_2^k) - V_2^{\pi_k}(s_2^k)}_{\text{same quantity at next time step}} \\
&\leq H\mathcal{I} + \sum_{h=1}^{H} 2\sqrt{\alpha^k_{h,\rho_h(z_h^k)}}\|\phi_h(z_h^k)\|_{{\Lambda^k_{h,\rho_h(z_h^k)}}^{-1}} + \zeta_h^k.
\end{aligned}$$

This equation allows us to bound the regret, under the good event, as

$$R_K \leq KH\mathcal{I} + \sum_{k=1}^{K}\sum_{h=1}^{H} 2\sqrt{\alpha^k_{h,\rho_h(z_h^k)}}\|\phi_h(z_h^k)\|_{{\Lambda^k_{h,\rho_h(z_h^k)}}^{-1}} + \zeta_h^k.$$

We start from the last term, which we rewrite, by exchanging the index of the sums as

$$\sum_{k=1}^{K}\sum_{h=1}^{H} \zeta_h^k = \sum_{h=1}^{H}\sum_{k=1}^{K} \zeta_h^k.$$

For fixed $h$ the term $\zeta_h^k$ is a martingale difference and, if it is bounded almost surely in $[-C, C]$ for some constant $C > 0$, we can apply Hoeffding's inequality to prove that w.p. at least $1 - \delta$,

$$-2C\sqrt{K\log(1/\delta)} \leq \sum_{k=1}^{K} \zeta_h^k \leq 2C\sqrt{K\log(1/\delta)}.$$

In our case, due to our problem definition we have that $C$ may be chosen as $L_\phi \sup_n \mathcal{R}_{h,n}$. Imposing that the previous inequality works at every step at the same time, and summing over $h$, we get, w.p. $1 - \delta$,

$$\sum_{h=1}^{H}\sum_{k=1}^{K} \zeta_h^k \leq 2\sqrt{K\log(H/\delta)} \sum_{h=1}^{H} L_\phi \sup_n \mathcal{R}_{h,n}. \tag{E.10}$$

Now, we go on bounding the other term

$$\begin{aligned}\sum_{k=1}^{K}\sum_{h=1}^{H} 2\sqrt{\alpha^k_{h,\rho_h(z_h^k)}}\|\phi_h(z_h^k)\|_{{\Lambda^k_{h,\rho_h(z_h^k)}}^{-1}} &= \sum_{h=1}^{H}\sum_{k=1}^{K} 2\sqrt{\alpha^k_{h,\rho_h(z_h^k)}}\|\phi_h(z_h^k)\|_{{\Lambda^k_{h,\rho_h(z_h^k)}}^{-1}} \\ &\leq \sum_{h=1}^{H} \sqrt{K}\sqrt{\sum_{k=1}^{K} 4\alpha^k_{h,\rho_h(z_h^k)}\|\phi_h(z_h^k)\|^2_{{\Lambda^k_{h,\rho_h(z_h^k)}}^{-1}}}\end{aligned} \tag{E.11}$$

Where the last passage is due to the Cauchy-Schwartz inequality. The last sum can be rewritten by summing over the regions instead of the episodes:

$$\sum_{k=1}^{K} 4\alpha^k_{h,\rho_h(z_h^k)}\|\phi_h(z_h^k)\|^2_{{\Lambda^k_{h,\rho_h(z_h^k)}}^{-1}} = \sum_{n=1}^{N_h}\sum_{i_n=1}^{p^K_{h,n}} 4\alpha^{i_n}_{h,n}\|\phi_h(z_h^{i_n})\|^2_{{\Lambda^{i_n}_{h,n}}^{-1}},$$

where the index $i_n$ enumerates the episodes $k$ where $z_h^k \in \mathcal{Z}_{h,n}$. At this point, recall that we have set $\sqrt{\alpha^k_{h,n}} = \sqrt{\beta^k_{h,n}} + \sqrt{p^k_{h,n}} + \lambda^{-1}\mathcal{R}_{h,n}$, so that, in particular

$$\begin{aligned}\alpha^k_{h,n} &\leq 3\beta^k_{h,n} + 3p^k_{h,n} + 3\lambda^{-1}\mathcal{R}_{h,n} \\ &\leq 3p^k_{h,n}\mathcal{I}^2 + 3(H+\sigma)^2 d_h + 3(H+\sigma)^2\log(\delta^{-1}\mathcal{N}(1/\sqrt{k}, \mathcal{V}_h)) + 3\lambda^{-1}\mathcal{R}_{h,n}.\end{aligned}$$

This way, the previous sum $\sum_{n=1}^{N_h}\sum_{i_n=1}^{p_{h,n}^K} 4\alpha_{h,n}^{i_n}\|\phi_h(z_h^{i_n})\|^2_{\Lambda_{h,n}^{i_n}{}^{-1}}$ can be again decomposed in few terms, according to the decomposition of $\alpha_{h,n}^k$.

$$12\sum_{n=1}^{N_h}\sum_{i_n=1}^{p_{h,n}^K} p_{h,n}^K\mathcal{I}^2\|\phi_h(z_h^{i_n})\|^2_{\Lambda_{h,n}^{i_n}{}^{-1}} = 12\sum_{n=1}^{N_h} p_{h,n}^K\mathcal{I}^2\sum_{i_n=1}^{p_{h,n}^K}\|\phi_h(z_h^{i_n})\|^2_{\Lambda_{h,n}^{i_n}{}^{-1}}.$$

Note that, by construction of the matrix $\Lambda_{h,n}^{i_n}$, we can use Lemma 11 from [2] to have

$$\sum_{i_n=1}^{p_{h,n}^K}\|\phi_h(z_h^{i_n})\|^2_{\Lambda_{h,n}^{i_n}{}^{-1}} \le 2d_h\log((\lambda + p_{h,n}^K L_\phi^2/d_h)) - \log(\det(\lambda I))),$$

where by fixing $\lambda = 1$ we get

$$\sum_{i_n=1}^{p_{h,n}^K}\|\phi_h(z_h^{i_n})\|^2_{\Lambda_{h,n}^{i_n}{}^{-1}} \le 2d_h\log((1 + p_{h,n}^K L_\phi^2/d_h)) \le 2d_h\log(KL_\phi^2+1).$$

Therefore, we can bound the whole term as

$$\begin{aligned}12\sum_{n=1}^{N_h}\sum_{i_n=1}^{p_{h,n}^K} p_{h,n}^K\mathcal{I}^2\|\phi_h(z_h^{i_n})\|^2_{\Lambda_{h,n}^{i_n}{}^{-1}} &\le 12\sum_{n=1}^{N_h} p_{h,n}^K\mathcal{I}^2 2d_h\log(KL_\phi^2+1)\\ &= 24K\mathcal{I}^2 d_h\log(KL_\phi^2+1). \qquad \text{(E.12)}\end{aligned}$$

The remaining terms can be bounded in a simpler way: by just calling $\Psi := (H+\sigma)^2 d_h + (H+\sigma)^2\log(\delta^{-1}\mathcal{N}(1/\sqrt{k},\mathcal{V}_h)) + \mathcal{R}_{h,n}$, we have

$$\begin{aligned}\sum_{n=1}^{N_h}\sum_{i_n=1}^{p_{h,n}^K} 12\Psi\|\phi_h(z_h^{i_n})\|^2_{\Lambda_{h,n}^{i_n}{}^{-1}} &= 12\Psi\sum_{n=1}^{N_h}\sum_{i_n=1}^{p_{h,n}^K}\|\phi_h(z_h^{i_n})\|^2_{\Lambda_{h,n}^{i_n}{}^{-1}}\\ &\le 12\Psi\sum_{n=1}^{N_h} 2d_h\log(KL_\phi^2+1)\\ &\le 24\Psi N_h d_h\log(KL_\phi^2+1),\end{aligned}$$

where we have used Lemma 11 by [2] as before. This result, together with Equation (E.12) can be inserted into the previous Equation (E.11) to get

$$\sum_{k=1}^{K}\sum_{h=1}^{H} 2\sqrt{\alpha^k_{h,\rho_h(z_h^k)}}\|\phi_h(z_h^k)\|_{\Lambda^k_{h,\rho_h(z_h^k)}{}^{-1}} \le \sum_{h=1}^{H}\sqrt{K}\sqrt{\sum_{k=1}^{K} 4\alpha^k_{h,\rho_h(z_h^k)}\|\phi_h(z_h^k)\|^2_{\Lambda^k_{h,\rho_h(z_h^k)}{}^{-1}}}$$
$$\le \sqrt{24}\sum_{h=1}^{H}\sqrt{K}\sqrt{(K\mathcal{I}^2+\Psi N_h)d_h\log(KL_\phi^2+1)}$$
$$\le \sqrt{24}\sum_{h=1}^{H}(K\mathcal{I}+\sqrt{\Psi N_h K})\sqrt{d_h\log(KL_\phi^2+1)}.$$

Therefore, the full bound on the reget can be written, also thanks to Equation (E.10) as

$$R_K \le 2\sqrt{K\log(H/\delta)}\sum_{h=1}^{H} L_\phi \sup_n \mathcal{R}_{h,n} + \sqrt{24}\sum_{h=1}^{H}(K\mathcal{I}+\sqrt{\Psi N_h K})\sqrt{d_h\log(KL_\phi^2+1)}.$$

Ignoring terms that are logarithmic in $K, H, L_\phi, 1/\delta$ and passing to the $\widetilde{\mathcal{O}}$ notation, we get

$$R_K \le \widetilde{\mathcal{O}}\left(\sqrt{K}\sum_{h=1}^{H}(L_\phi \sup_n \mathcal{R}_{h,n} + \sqrt{d_h\Psi N_h}) + KH\mathcal{I}\sqrt{d_h}\right).$$

Now, note that by definition of $\Psi$, the previous can be rewritten as

$$R_K \le \widetilde{\mathcal{O}}\Bigg((H+\sigma)\sum_{h=1}^{H}\sqrt{KN_h}\Big((L_\phi+\sqrt{d_h})\sup_n \mathcal{R}_{h,n}$$
$$+ d_h + \sqrt{d_h\log(\mathcal{N}(1/\sqrt{K},\mathcal{V}_h))}\Big) + KH\mathcal{I}\sqrt{d_h}\Bigg).$$

□

Merging the previous result with Proposition 54, which bounds the size of the covering, and the fact that the good event is designed to have probability at least $1-\delta$ (Corollary 56), we get the main result.

**Corollary 61.** *Assume that $L_\phi = \mathcal{O}(1)$ and $\sup_{h\in[H],n\in[N_h]}\mathcal{R}_{h,n} = \mathcal{O}(\sqrt{d_h})$. Then, with probability at least $1-\delta$, CINDERELLA (Algorithm 6.1), with $\lambda = 1$ achieves a regret*

*bound of order*

$$R_K = \widetilde{\mathcal{O}}\left(\sum_{h=1}^{H}(H+\sigma)N_h d_h\sqrt{K} + \sum_{h=1}^{H}\sqrt{d_h}\mathcal{I}K\right).$$

### E.4.1. Reduction from Mildly Smooth to Locally Linearizable

**Theorem 62.** *For any $f \in W^{\nu,\infty}(\mathcal{Z})$, there are $\theta_1, \dots \theta_n, \dots \theta_N$, all in $\mathbb{R}^{d_{\nu_*}}$, such that*

$$\|f(\cdot) - \phi(\cdot)^\top \theta_{\rho(\cdot)}\|_{L^\infty} \leq L_\nu(f)\varepsilon^\nu.$$

*Moreover, the components of each of the vectors $\theta_n$ satisfy*

$$|\theta_n[\boldsymbol{\alpha}]| \leq \begin{cases} \|f\|_{L^\infty} & \boldsymbol{\alpha} = 0 \\ \|f\|_{W^{\nu,\infty}} & \boldsymbol{\alpha} \neq 0. \end{cases}$$

*Proof.* Let $z \in \mathcal{Z}$ and $n = \rho(z)$. Then, if we take $\theta_n$ to be the vector with components

$$\theta_n[\boldsymbol{\alpha}] = \frac{D^{\boldsymbol{\alpha}} f(z^n)}{\boldsymbol{\alpha}!},$$

we have,

$$\begin{aligned} |f(z) - \phi(z)^\top \theta_{\rho(z)}| &= |f(z) - \phi(z)^\top \theta_n| \\ &= \left| f(z) - \sum_{\|\boldsymbol{\alpha}\|_1 \leq \nu_*} \frac{D^{\boldsymbol{\alpha}} f(z^n)}{\boldsymbol{\alpha}!}(x - z^n)^{\boldsymbol{\alpha}} \right| \\ &= |f(z) - T^{\nu_*}_{z^n}[f](z)|, \end{aligned}$$

where we have used both the definitions of $\phi$ and $\theta_n$. Then, using Theorem 51, we have

$$\begin{aligned} |f(z) - \phi(z)^\top \theta_{\rho(z)}| &= |f(z) - T^{\nu_*}_{z^n}[f](z)| \\ &\leq L_\nu(f)\|z - z^n\|^\nu_\infty. \end{aligned}$$

By definition of $\mathcal{Z}^\varepsilon$, we have $\|z - z^n\|_\infty = \|z - \rho(z)\|_\infty \leq \varepsilon$, which entails the first part of the thesis. For what concerns the second one, we bound the magnitude of the vectors $\theta_n$

component by component:

$$|\theta_n[\boldsymbol{\alpha}]| = \left|\frac{D^{\boldsymbol{\alpha}} f(z^n)}{\boldsymbol{\alpha}!}\right| \leq |D^{\boldsymbol{\alpha}} f(z^n)| \leq \begin{cases} \|f\|_{L^\infty} & \boldsymbol{\alpha} = 0 \\ \|f\|_{\mathrm{W}^{\nu,\infty}} & \boldsymbol{\alpha} \neq 0 \end{cases},$$

where the last comes from the definition of $\|f\|_{\mathrm{W}^{\nu,\infty}}$ □

**Theorem 48.** *Let $\mathcal{M}$ be a Mildly Smooth MDP, and $\varepsilon \leq 1/(2C_{\mathcal{T}}H)^{1/\nu}$. Then, choosing $\mathcal{U}_h = \{\mathcal{Z}_n\}_n^{N_h}$ as in (6.8) and taking $\phi_h$ as in (6.9)), the triple $(\mathcal{M},\mathcal{U},\phi)$ is a Locally Linearizable MDP, where $\mathcal{B}_{h,n}$ may be chosen in a way that*

$$L_\phi = (1 + 2\sqrt{d}_{\nu_*})H, \qquad \mathcal{R}_{h,n} = 2\sqrt{d}_{\nu_*}C_{\mathcal{T}}, \qquad \mathcal{I} \leq 2C_{\mathcal{T}}^{MS}\varepsilon^\nu,$$

*and $\sup_{h,\boldsymbol{\theta}_h \in \mathcal{B}_h} \|Q_h[\boldsymbol{\theta}_h]\|_{L^\infty} \leq 2H$.*

*Proof.* Let us define the sets

$$\mathcal{B}_{h,n} := \left\{\theta \in \mathbb{R}^{d_{\nu_*}} : \ \|\phi_h(\cdot)^\top \theta\|_{L^\infty(\mathcal{Z}_n)} \leq A_h, \ \|\theta\|_\infty \leq A'_h\right\}, \tag{E.13}$$

where $L^\infty(\mathcal{Z}_n)$ stands for the supremum norm restriceted to $\mathcal{Z}_n$. Now, let $Q_{h+1}[\boldsymbol{\theta}_{h+1}](z) = \phi(z)^\top \theta_{\rho(z)}$ with $\boldsymbol{\theta}_{h+1} \in \mathcal{B}_{h+1}$ for a set $\mathcal{B}_{h+1} = \bigtimes_{n=1}^N \mathcal{B}_{h+1,n}$. Two facts hold:

1. By the non-expansivity of the Bellman optimality operator we have

$$\|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}]\|_{L^\infty} \leq \|Q[\boldsymbol{\theta}_{h+1}]\|_{L^\infty} \leq A_{h+1}. \tag{E.14}$$

2. By the Mildly smooth assumption (Asm. 21), we have

$$\|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}]\|_{\mathrm{W}^{\nu,\infty}} \leq C_{\mathcal{T}}(\|Q_{h+1}[\boldsymbol{\theta}_{h+1}]\|_{L^\infty} + 1) \leq C_{\mathcal{T}}(1 + A_{h+1}). \tag{E.15}$$

Now, we can prove the low inherent Bellman error property. To do this we apply Theorem 62 which guarantees that there are $\theta_1, \dots \theta_n, \dots \theta_N$, all in $\mathbb{R}^{d_{\nu_*}}$, such that

$$\|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}](\cdot) - \phi_h(\cdot)^\top \theta_{\rho(\cdot)}\|_{L^\infty} \leq L_\nu(\mathcal{T}_h Q[\boldsymbol{\theta}_{h+1}])\varepsilon^\nu.$$

Thus, taking $\boldsymbol{\theta}_h$ as the matrix stacking all these $\theta_1, \dots \theta_n, \dots \theta_N$, we get that the inherent Bellman error is bounded by

$$\mathcal{I} \leq L_\nu(\mathcal{T}_h Q[\boldsymbol{\theta}_{h+1}])\varepsilon^\nu \leq \|\mathcal{T}_h Q[\boldsymbol{\theta}_{h+1}]\|_{\mathrm{W}^{\nu,\infty}}\varepsilon^\nu \leq C_\mathcal{T}(1+A_{h+1})\varepsilon^\nu.$$

What is missing is to prove that this choice satisfies $\boldsymbol{\theta}_h \in \mathcal{B}_h$.

By triangular inequality, the following bound on the supremum norm holds

$$\begin{aligned}\|Q_h[\boldsymbol{\theta}_h](\cdot)\|_{L^\infty} &= \|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}](\cdot) - Q_h[\boldsymbol{\theta}_h](\cdot)\|_{L^\infty} + \|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}](\cdot)\|_{L^\infty} \\ &\leq \mathcal{I} + \|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}](\cdot)\|_{L^\infty} \\ &\leq \mathcal{I} + \|Q_{h+1}[\boldsymbol{\theta}_{h+1}](\cdot)\|_{L^\infty} \\ &\leq C_\mathcal{T}(1+A_{h+1})\varepsilon^\nu + A_{h+1}.\end{aligned}$$

By theorem 62 (which holds for the Taylor feature map, while we are using $H \cdot \varphi$ here), the components of the $\theta_{h,n}$ parameters that form $\boldsymbol{\theta}_h$ satisfy

$$H \cdot |\theta_{h,n}[\boldsymbol{\alpha}]| \leq \begin{cases} \|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}]\|_{L^\infty} \leq \|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}]\|_{\mathrm{W}^{\nu,\infty}} & \boldsymbol{\alpha} = 0 \\ \|\mathcal{T}_h Q_{h+1}[\boldsymbol{\theta}_{h+1}]\|_{\mathrm{W}^{\nu,\infty}} & \boldsymbol{\alpha} \neq 0, \end{cases}$$

which means, by equation (E.15), that

$$|\theta_{h,n}[\boldsymbol{\alpha}]| \leq H^{-1}C_\mathcal{T}(1+A_{h+1}),$$

so that, taking the maximum of all the components, we have $\|\theta_{h,n}\|_\infty \leq H^{-1}C_\mathcal{T}(1+A_{h+1})$. Therefore, Equation (E.13) requires the condition

$$A_h \geq C_\mathcal{T}(1+A_{h+1})\varepsilon^\nu + A_{h+1} \qquad A'_h \geq H^{-1}C_\mathcal{T}(1+A_{h+1}).$$

If we set $\varepsilon \leq 1/(2C_\mathcal{T}H)^{1/\nu}$, the previous inequalities are satisfied for

$$A_h = H - h \qquad A'_h = 2C_\mathcal{T}.$$

Indeed, the following inequalities hold,

$$\begin{aligned} A_h &= H - h \\ &= H - h - 1 + 1 \\ &= A_{h+1} + 1 \\ &= A_{h+1} + \frac{2HC_{\mathcal{T}}}{2HC_{\mathcal{T}}} \\ &\geq A_{h+1} + 2C_{\mathcal{T}}\varepsilon^{\nu} \\ &\geq A_{h+1} + C_{\mathcal{T}}(1 + H - h - 1)\varepsilon^{\nu} \\ &= A_{h+1} + C_{\mathcal{T}}(1 + A_{h+1})\varepsilon^{\nu}. \end{aligned}$$

For what concerns the other term we can simply write

$$A'_h = 2C_{\mathcal{T}} \geq H^{-1}C_{\mathcal{T}}(1 + A_{h+1}).$$

This shows that

$$\sup_{h,\boldsymbol{\theta}_h \in \mathcal{B}_h} \|Q_h[\boldsymbol{\theta}_h]\|_{L^\infty} \leq \max_h A_h \leq 2H.$$

We now pass to the other points.

$$\mathcal{R}_{h,n} = \operatorname{diam}(\mathcal{B}_{h,n}) \leq \sqrt{d}_{\nu_*} A'_h = 2\sqrt{d}_{\nu_*} C_{\mathcal{T}},$$

where the presence of $\sqrt{d}_{\nu_*}$ is needed to pass from the $\infty-$norm in the definition of $\mathcal{B}_{h,n}$ to the norm two in the definition of $\operatorname{diam}(\mathcal{B}_{h,n})$. Instead, for the feature map we have

$$\begin{aligned} L_\phi &= \sup_{z \in \mathcal{Z}} \|\phi(z)\|_2 \\ &= H \sup_{z \in \mathcal{Z}} \|\psi^{\nu_*}_{\rho(z)}(z)\|_2. \end{aligned}$$

Note that, looking at the precise definition of $\psi^{\nu_*}_y(z)$ in (B.4), calling $\rho(z) = n$, this norm can be explicitly written as

$$\begin{aligned} \|\psi^{\nu_*}_n(z)\|_2 &= \sqrt{\sum_{\|\boldsymbol{\alpha}\|_1 \leq \nu_*} (z - z^n)^{2\boldsymbol{\alpha}}} \\ &\leq 1 + \sqrt{\sum_{1 \leq \|\boldsymbol{\alpha}\|_1 \leq \nu_*} (z - z^n)^{2\boldsymbol{\alpha}}} \\ &\leq 1 + \sqrt{\sum_{1 \leq \|\boldsymbol{\alpha}\|_1 \leq \nu_*} 2^2} = 1 + 2\sqrt{d}_{\nu_*}. \end{aligned}$$

Where the third passage is due to the fact that $\|z - z^n\|_\infty \leq 2$ as we have assumed $\mathcal{Z} = [-1, 1]^2$. □

# Acknowledgements (ENG)

These past four years at AirLab have been the best of my life. The guy who first walked in—immature and confused about what he wanted to do with his life—is a different person today (just slightly less immature). Below, I would like to thank some of the people who were most important on this journey. I've chosen not to use specific names, as I like to think that individuals can gather I'm thinking of them from the hints I drop—and also to avoid being sued.

I want to thank my colleagues, who supported me in so many projects and accompanied me to the most remote conferences in the world; including those who argue every single time about where to eat and who shares a bed with whom (okay, phrased like that, it sounds bad). My thanks go to those who brought a smile to every other machine learning researcher in the world by frequently exceeding the limit of ten alcoholic units per night, as well as all the lower bounds of online learning. It is thanks to them that we managed to make so many crazy friends in the global research community, from Vancouver to Vienna, via Washington and Baltimore; from the world champion of burping to the world champion of speeding tickets (over nine thousand Canadian dollars).

Speaking of gangs of madmen, I cannot fail to mention the COLT2024 crew, with whom I am still technically "agreed" to write the paper that will change the world: *DUAL Interior Point Algorithm (DUALIPA): the End of an Era for Reinforcement Learning*. I tried to get to work, but the only formula I managed to write was $< 3$. To the reader who might be bewildered by such feats, I must remind them that conferences are not just this, but precious moments of cultural elevation. In all honesty, I still solemnly remember our pilgrimages to the temples of the MLB, the NBA, and the spiritual asceticism sessions in Arnold Schwarzenegger's gym.

I thank all the thesis students I have supervised over the years, confident that we produced quality research together and hoping to be remembered as a role-model father figure, like Goku, Ging Freecss, Yujiro Hanma, and Gendo Ikari. I also thank the colleagues who stood by me during the worst moments, like when we were accused of copying a paper that came out on the exact day of the deadline we had submitted to, or when we were

rejected by a reviewer who didn't understand that $\cup_{n=1}^{\infty} E_n$ does not depend on $n$.Moving across the ocean, I thank my colleagues in Alberta, where I was a visiting researcher, for all the pizza scrounged at RL seminars and for assisting me with my personal bests in squat and bench press. I also thank our boss, who didn't let me splat myself on the skis in Banff, after physically forcing me to follow him despite the fact that I couldn't ski. It's a bit of a contradiction, actually, but I'll thank him anyway just in case he tries to force me to go again.I also thank my colleagues from my internship at G-Research, with whom I lived in London for an entire summer. I would have a thousand funny stories to tell, but it's all under NDA—except for that time I promised to take them to a place for "real Italians" and we ended up at Pizza Express.

Moving on to my life outside of work—which, surprisingly, exists—I must thank those who contributed directly to my work. For example, my great friend whom we like to call the "Sultan of Gallarate," with whom I took the Fourier Analysis course at the State University. It was there that I came into contact with the De La Vallée Poussin kernel (fig. 5.1), the backbone of the results in Chapter 5. In this regard, I also thank the very kind professor of the course, hoping she managed to cope with that student who suggested compactifying the space $\mathbb{R}^d$ into a single point. I thank all the friends I have in common with the Sultan. It is surprising how much the percentage of PhD students in that group has risen, and perhaps even more so the percentage of Yu-Gi-Oh players who travel all the way to Monza just to pull a Red-Eyes Black Dragon.

Staying on the theme of "where the results of this thesis come from," it is impossible not to mention the Racer bar in Crugnola. One evening, while I was there with my friends, lulled by the notes of Bellissima by Annalisa, I had a vision of Sukuna and his slashes hitting Mahoraga within the Malevolent Shrine domain. In that moment, I had the idea of partitioning the space with many straight lines, which I use in the proof of theorem 31. I thank those present that evening and, out of pure courtesy, those absent as well (though with less enthusiasm). A special mention and double thanks to those among you who play for the team sponsored by the Racer, San Giulio Crugnola. Basically the Barcelona of the Varese area, but with much more hunger for victory and, probably, a more demanding "third half." I'm not sure how appropriate it is to give all these details, but just in case, I thank the thesis reviewers as well so they might turn a blind eye.

Penultimate, but not least, I thank my friends from the PlayFitness gym, including anyone who reposts my Instagram stories, the "taleBani"—who, far from being an extremist group, take their name from my little cousin Bana (who obviously isn't actually called that)—and also the guys from the track and field team, even though I haven't actually gone to practice since roughly 2002.

Finally, I thank my supervisor, who gave me all the research freedom I asked for, and whom I am helping to prove to the world that Busto Arsizio is better than New York. I also obviously thank my family, with the commitment to produce a version of this thesis for children under ten as soon as possible, so that at the next Christmas party in Crenna we have a pastime other than Legos to entertain cousins, both big and small.

# Acknowledgements (IT)

Questi quattro anni passati in airlab sono stati i migliori della mia vita. Il ragazzo che è entrato, immaturo e confuso su quello che avrebbe voluto fare della sua vita, oggi è una persona diversa (solamente immatura). Di seguito, ringrazio alcune delle persone che sono state più importanti in questo percorso. Ho deciso di non fare nomi specifici, perché mi piace pensare che i singoli possano carpire che li penso dagli accenni che faccio, oltre che per non essere denunciabile.

Ringrazio i miei colleghi, che mi hanno supportato in tanti lavori, e accompagnato nelle conferenze più sperdute al mondo; includendo quelli che ogni volta litigano su dove andare a mangiare e chi condivide il letto con chi (ok, detta così suona male). Ringrazio quelli che hanno regalato un sorriso a tutti gli altri ricercatori del machine learning mondiale, superando spesso il limite di dieci unità alcoliche per serata, oltre che tutti i lower bound dell'online learning. E' grazie a loro se siamo riusciti a farci così tanti amici folli nella ricerca mondiale, da Vancouver a Vienna, passando per Washington e Baltimora; dal campione mondiale di rutti a quello di multe per eccesso di velocità (oltre novemila dollari canadesi). Parlando delle bande di matti non posso citare quella di COLT2024, con cui siamo ancora d'accordo per scrivere il paper che cambierà il mondo: *DUAL Interior Point Algorithm (DUALIPA): the End of an Era for Reinforcement Learning*. Ho provato a mettermi al lavoro, ma l'unica formula che ho saputo scrivere è stata $< 3$. Al lettore che potrebbe restare interdetto davanti a tali gesta, preme ricordare che le conferenze non sono solo questo, bensì preziosi momenti di elevazione culturale. A onor del vero, ricordo ancora con solennità i nostri pellegrinaggi presso i templi della MLB, della NBA e le sessioni di ascesi spirituale nella palestra di Arnold Schwarzenegger. Ringrazio tutti i tesisti che ho seguito in questi anni, sicuro di aver prodotto con loro ricerca di qualità e sperando di essere ricordato come un padre modello, come Goku, Ging Freecss, Yujiro Hanma e Gendo Ikari. Ringrazio anche i colleghi che mi sono stati vicini nei momenti peggiori, come quando siamo stati accusati di aver copiato un paper uscito il giorno esatto della deadline a cui avevamo mandato noi o quando siamo stati rigettati dal revisore che non capiva che $\cup_{n=1}^{\infty} E_n$ non dipende da $n$.

Passando sull'altra riva dell'oceano, ringrazio i colleghi dell'Alberta, dove sono stato ricer-

catore in visita, per tutta la pizza scroccata ai seminari di RL e per avermi assistito nel personale di squat e panca piana. Ringrazio anche nostro capo, che non ha permesso che mi spiattellassi sugli sci a Banff, dopo avermi costretto fiscamente a seguirlo pur non sapendo sciare. E' un po' un controsenso in effetti, ma io nel dubbio lo ringrazio lo stesso, mica che mi costringe ad andare di nuovo. Ringrazio anche i miei colleghi dell'internship a G-Research, con cui ho vissuto a Londra per un'intera estate. Avrei mille storie divertenti da raccontare, ma è tutto coperto da NDA, a parte quella volta che gli avevo promesso di portarli in un locale da "veri italiani" e poi siamo andati a Pizza Express.

Passando alla mia vita fuori dal lavoro, che sorprendentemente esiste, non posso mancare di ringraziare chi ha contribuito direttamente al mio lavoro. Ad esempio, il mio grandissimo amico che ci piace chiamare il Sultano di Gallarate, con cui ho seguito il corso Analisi di Fourier in università statale. E' lì che sono venuto in contatto con il nucleo di De La Vallée Poussin (fig. 5.1), la spina dorsale dei risultati del capitolo 5. Ringrazio a tal proposito anche la gentilissima professoressa del corso, sperando che se la sia cavata con quello studente che le proponeva di compattificare lo spazio $\mathbb{R}^d$ in un punto. Ringrazio tutti gli amici che ho in comune col Sultano. E' sorprendentemente quanto la percentuale di dottorandi in quel gruppo sia salita, e forse ancora di più quanto sia salita quella di giocatori di Yu-Gi-Oh che vanno fino a Monza per sbustare un Drago Nero Occhi Rossi.

Restando in tema di "da dove arrivano i risultati di questa tesi" mi è impossibile non citare il bar Racer di Crugnola. Una sera, mentre ero lì coi miei amici, cullato tra le note di Bellissima di Annalisa, ho avuto una visione di Sukuna, e dei suoi fendenti che colpivano Mahaoraga nel dominio del Malevolent Shrine. In quel momento mi è venuta l'idea di spezzettare lo spazio con tantissime linee dritte, che utilizzo nella dimostrazione del theorem 31. Ringrazio i presenti di quella serata e, per pura cortesia, anche gli assenti (ma con meno entusiasmo). Menzione d'onore e ringraziamento doppio per chi tra voi milita nella squadra sponsorizzata dal Racer, il San Giulio Crugnola. Fondamentalmente il Barcellona del varesotto, ma con molta più fame di vittoria e, probabilmente, un terzo tempo più impegnativo. Non so quanto sia opportuno dare tutti questi dettagli, ma nel dubbio ringrazio anche i revisori della tesi perché chiudano un occhio.

Penultimi, ma non per importanza, ringrazio i miei amici della palestra PlayFitness, compreso chiunque riposti le mie storie da finto muscoloso, i taleBani che, lungi dall'essere un gruppo estremista prendono il nome dalla mia cuginetta Bana (che ovviamente non si chiama così), e anche i ragazzi dell'atletica, anche se non vado ad allenarmi praticamente dal 2002.

Infine, ringrazio il mio supervisore, che mi ha lasciato tutta la libertà di ricerca che

chiedevo, e che sto aiutando a dimostrare al mondo che Busto Arsizio è meglio di New York. Ringrazio ovviamente anche la mia famiglia, con l'impegno di produrre il prima possibile una versione di questa tesi per bambini sotto i dieci anni, così alla prossima festa di Natale e Crenna abbiamo un passatempo diverso dai Lego per intrattenere i cugini grandi e piccoli.

# Bibliography

[1] Y. Abbasi-Yadkori and C. Szepesvári. Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 1–26. JMLR Workshop and Conference Proceedings, 2011.

[2] Y. Abbasi-Yadkori, D. Pál, and C. Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

[3] R. A. Adams and J. J. Fournier. *Sobolev spaces*, volume 140. Elsevier, 2003.

[4] A. Agarwal, N. Jiang, S. M. Kakade, and W. Sun. Reinforcement learning: Theory and algorithms. *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep*, 32:96, 2019.

[5] R. Agrawal. The continuum-armed bandit problem. *SIAM journal on control and optimization*, 33(6):1926–1951, 1995.

[6] F. Al-Ani, M. Wang, J. Charles, A. Ong, J. Forday, and V. Modi. Simulation-driven reinforcement learning in queuing network routing optimization. *arXiv preprint arXiv:2507.18795*, 2025.

[7] B. D. Anderson and J. B. Moore. *Optimal control: linear quadratic methods*. Courier Corporation, 2007.

[8] N. Aronszajn. Theory of reproducing kernels. *Transactions of the American mathematical society*, 68(3):337–404, 1950.

[9] K. Asadi, D. Misra, and M. Littman. Lipschitz continuity in model-based reinforcement learning. In *International Conference on Machine Learning*, pages 264–273. PMLR, 2018.

[10] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2):235–256, 2002.

[11] P. Auer, R. Ortner, and C. Szepesvári. Improved rates for the stochastic continuum-armed bandit problem. In *International Conference on Computational Learning Theory*, pages 454–468. Springer, 2007.

[12] P. Auer, T. Jaksch, and R. Ortner. Near-optimal regret bounds for reinforcement learning. *Advances in neural information processing systems*, 21, 2008.

[13] 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.

[14] M. G. Azar, R. Munos, and B. Kappen. On the sample complexity of reinforcement learning with a generative model. *arXiv preprint arXiv:1206.6461*, 2012.

[15] M. G. Azar, I. Osband, and R. Munos. Minimax regret bounds for reinforcement learning. In *International conference on machine learning*, pages 263–272. PMLR, 2017.

[16] F. Bacchiocchi, G. Genalti, D. Maran, M. Mussi, M. Restelli, N. Gatti, and A. M. Metelli. Autoregressive bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 937–945. PMLR, 2024.

[17] T. Bagby, L. Bos, and N. Levenberg. Multivariate simultaneous approximation. *Constructive approximation*, 18(4):569–577, 2002.

[18] R. Bellman. Dynamic programming. *science*, 153(3731):34–37, 1966.

[19] C. Berner, G. Brockman, B. Chan, V. Cheung, P. Dębiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, et al. Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*, 2019.

[20] D. Bertsekas. *Dynamic programming and optimal control: Volume I*, volume 4. Athena scientific, 2012.

[21] L. Besson and E. Kaufmann. What doubling tricks can and can't do for multi-armed bandits. *arXiv preprint arXiv:1803.06971*, 2018.

[22] L. Bisi, P. Liotet, L. Sabbioni, G. Reho, N. Montali, M. Restelli, and C. Corno. Foreign exchange trading: A risk-averse batch reinforcement learning approach. In *Proceedings of the First ACM International Conference on AI in Finance*, pages 1–8, 2020.

[23] E. Bøhn, E. M. Coates, D. Reinhardt, and T. A. Johansen. Data-efficient deep reinforcement learning for attitude control of fixed-wing uavs: Field experiments. *IEEE Transactions on Neural Networks and Learning Systems*, 35(3):3168–3180, 2023.

[24] H.-J. Bungartz and M. Griebel. Sparse grids. *Acta numerica*, 13:147–269, 2004.

[25] N. Cesa-Bianchi and G. Lugosi. Combinatorial bandits. *Journal of Computer and System Sciences*, 78(5):1404–1422, 2012.

[26] 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.

[27] S. R. Chowdhury, A. Gopalan, and O.-A. Maillard. Reinforcement learning in parametric mdps with exponential families. In *International Conference on Artificial Intelligence and Statistics*, pages 1855–1863. PMLR, 2021.

[28] K. Chua, R. Calandra, R. McAllister, and S. Levine. Deep reinforcement learning in a handful of trials using probabilistic dynamics models. *Advances in neural information processing systems*, 31, 2018.

[29] A. Cohen, T. Koren, and Y. Mansour. Learning linear-quadratic regulators efficiently with only $\sqrt{t}$ regret. In *International Conference on Machine Learning*, pages 1300–1309. PMLR, 2019.

[30] N. Da Costa, M. Pförtner, L. Da Costa, and P. Hennig. Sample path regularity of gaussian processes from the covariance kernel. *arXiv preprint arXiv:2312.14886*, 2023.

[31] A. Damiani, G. Manganini, A. M. Metelli, and M. Restelli. Balancing sample efficiency and suboptimality in inverse reinforcement learning. In *International Conference on Machine Learning*, pages 4618–4629. PMLR, 2022.

[32] C. de la Vallée Poussin. Sur la meilleure approximation des fonctions d'une variable réelle par des expressions d'ordre donné. *CR Acad. Sci. Paris*, 166:799–802, 1918.

[33] C. J. De La Vallée Poussin et al. *Leçons sur l'approximation des fonctions d'une variable réelle*. Paris, 1919.

[34] S. Dean, H. Mania, N. Matni, B. Recht, and S. Tu. Regret bounds for robust adaptive control of the linear quadratic regulator. *Advances in Neural Information Processing Systems*, 31, 2018.

[35] O. D. Domingues, P. Ménard, E. Kaufmann, and M. Valko. Episodic reinforcement learning in finite mdps: Minimax lower bounds revisited. In *Algorithmic Learning Theory*, pages 578–598. PMLR, 2021.

[36] S. Drago, M. Mussi, and A. M. Metelli. A refined analysis of ucbvi. *arXiv preprint arXiv:2502.17370*, 2025.

[37] S. Du, S. Kakade, J. Lee, S. Lovett, G. Mahajan, W. Sun, and R. Wang. Bilinear classes: A structural framework for provable generalization in rl. In *International Conference on Machine Learning*, pages 2826–2836. PMLR, 2021.

[38] D. Dũng, V. Temlyakov, and T. Ullrich. *Hyperbolic cross approximation*. Springer, 2018.

[39] L. C. Evans. *Partial differential equations*, volume 19. American Mathematical Society, 2022.

[40] Y. Fan, H. Dong, X. Zhao, and P. Denissenko. Path-following control of unmanned underwater vehicle based on an improved td3 deep reinforcement learning. *IEEE Transactions on Control Systems Technology*, 32(5):1904–1919, 2024.

[41] D. J. Foster, S. M. Kakade, J. Qian, and A. Rakhlin. The statistical complexity of interactive decision making. *arXiv preprint arXiv:2112.13487*, 2021.

[42] D. J. Foster, A. Krishnamurthy, D. Simchi-Levi, and Y. Xu. Offline reinforcement learning: Fundamental barriers for value function approximation. *arXiv preprint arXiv:2111.10919*, 2021.

[43] D. J. Foster, N. Golowich, and Y. Han. Tight guarantees for interactive decision making with the decision-estimation coefficient. In *The Thirty Sixth Annual Conference on Learning Theory*, pages 3969–4043. PMLR, 2023.

[44] S. Fujimoto, H. Hoof, and D. Meger. Addressing function approximation error in actor-critic methods. In *International conference on machine learning*, pages 1587–1596. PMLR, 2018.

[45] A. Garivier, T. Lattimore, and E. Kaufmann. On explore-then-commit strategies. *Advances in Neural Information Processing Systems*, 29, 2016.

[46] M. Gigli et al. Thompson sampling for performance marketing and delayed conversions. 2024.

[47] J. Grant and D. Leslie. On thompson sampling for smoother-than-lipschitz bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 2612–2622. PMLR, 2020.

[48] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In *International conference on machine learning*, pages 1861–1870. Pmlr, 2018.

[49] H. Hasselt. Double q-learning. *Advances in neural information processing systems*, 23, 2010.

[50] T. Hofmann, B. Schölkopf, and A. J. Smola. Kernel methods in machine learning. 2008.

[51] D. Janz, D. Burt, and J. González. Bandit optimisation of functions in the matérn kernel rkhs. In *International Conference on Artificial Intelligence and Statistics*, pages 2486–2495. PMLR, 2020.

[52] 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.

[53] C. Jin, Z. Allen-Zhu, S. Bubeck, and M. I. Jordan. Is q-learning provably efficient? *Advances in neural information processing systems*, 31, 2018.

[54] 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. PMLR, 2020.

[55] C. Jin, Q. Liu, and S. Miryoosefi. Bellman eluder dimension: New rich classes of rl problems, and sample-efficient algorithms. *Advances in neural information processing systems*, 34:13406–13418, 2021.

[56] S. Kakade, A. Krishnamurthy, K. Lowrey, M. Ohnishi, and W. Sun. Information theoretic regret bounds for online nonlinear control. *Advances in Neural Information Processing Systems*, 33:15312–15325, 2020.

[57] R. E. Kalman et al. Contributions to the theory of optimal control. *Bol. soc. mat. mexicana*, 5(2):102–119, 1960.

[58] M. Kanagawa, P. Hennig, D. Sejdinovic, and B. K. Sriperumbudur. Gaussian processes and kernel methods: A review on connections and equivalences. *arXiv preprint arXiv:1807.02582*, 2018.

[59] Y. Katznelson. *An introduction to harmonic analysis*. Cambridge University Press, 2004.

[60] M. J. Kearns and U. Vazirani. *An introduction to computational learning theory*. MIT press, 1994.

[61] T. Kim, M. Jang, and J. Kim. A survey on simulation environments for reinforce-

ment learning. In *2021 18th International Conference on Ubiquitous Robots (UR)*, pages 63–67. IEEE, 2021.

[62] G. S. Kimeldorf and G. Wahba. A correspondence between bayesian estimation on stochastic processes and smoothing by splines. *The Annals of Mathematical Statistics*, 41(2):495–502, 1970.

[63] R. Kleinberg. Nearly tight bounds for the continuum-armed bandit problem. *Advances in Neural Information Processing Systems*, 17, 2004.

[64] R. Kleinberg, A. Slivkins, and E. Upfal. Multi-armed bandits in metric spaces. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 681–690, 2008.

[65] A. N. Kolmogorov and S. V. Fomin. *Introductory real analysis*. Courier Corporation, 1975.

[66] Q. Lan, Y. Pan, A. Fyshe, and M. White. Maxmin q-learning: Controlling the estimation bias of q-learning. *arXiv preprint arXiv:2002.06487*, 2020.

[67] S. Lange, T. Gabel, and M. Riedmiller. Batch reinforcement learning. In *Reinforcement learning: State-of-the-art*, pages 45–73. Springer, 2012.

[68] T. Lattimore and C. Szepesvari. Bandit algorithms. 2017. URL `https://tor-lattimore.com/downloads/book/book.pdf`.

[69] T. Lattimore, C. Szepesvari, and G. Weisz. Learning with good feature representations in bandits and in rl with a generative model. In *International conference on machine learning*, pages 5662–5670. PMLR, 2020.

[70] G. Li, L. Shi, Y. Chen, Y. Gu, and Y. Chi. Breaking the sample complexity barrier to regret-optimal model-free reinforcement learning. *Advances in Neural Information Processing Systems*, 34:17762–17776, 2021.

[71] Z. Li and J. Scarlett. Gaussian process bandit optimization with few batches. In *International Conference on Artificial Intelligence and Statistics*, pages 92–107. PMLR, 2022.

[72] T. P. Lillicrap, J. J. Hunt, A. Pritzel, N. Heess, T. Erez, Y. Tassa, D. Silver, and D. Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[73] P. Liotet, D. Maran, L. Bisi, and M. Restelli. Delayed reinforcement learning by

imitation. In *International Conference on Machine Learning*, pages 13528–13556. PMLR, 2022.

[74] Y. Liu, Y. Wang, and A. Singh. Smooth bandit optimization: generalization to holder space. In *International Conference on Artificial Intelligence and Statistics*, pages 2206–2214. PMLR, 2021.

[75] D. Maran and M. Restelli. Finite sample bounds for non-parametric regression: Optimal sample efficiency and space complexity. *arXiv preprint arXiv:2412.14744*, 2024.

[76] D. Maran, A. M. Metelli, and M. Restelli. Tight performance guarantees of imitator policies with continuous actions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 9073–9080, 2023.

[77] D. Maran, F. Bacchiocchi, F. E. Stradi, M. Castiglioni, N. Gatti, and M. Restelli. Bandits with ranking feedback. *Advances in Neural Information Processing Systems*, 37:80567–80608, 2024.

[78] D. Maran, A. M. Metelli, M. Papini, and M. Restelli. Local linearity: the key for no-regret reinforcement learning in continuous mdps. *Advances in Neural Information Processing Systems*, 37:75986–76029, 2024.

[79] D. Maran, A. M. Metelli, M. Papini, and M. Restelli. No-regret reinforcement learning in smooth mdps. In *Proceedings of the 41st International Conference on Machine Learning*, pages 34760–34789, 2024.

[80] D. Maran, A. M. Metelli, M. Papini, and M. Restelli. Projection by convolution: Optimal sample complexity for reinforcement learning in continuous-space mdps. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 3743–3774. PMLR, 2024.

[81] D. Maran, P. Olivieri, F. E. Stradi, G. Urso, N. Gatti, and M. Restelli. Online markov decision processes configuration with continuous decision space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 14315–14322, 2024.

[82] D. Maran, G. Tedeschi, E. Gusmeroli, and M. Restelli. Reinforcement learning vs optimal control: Sparse nonlinear dynamical systems between theory and practice. In *Eighteenth European Workshop on Reinforcement Learning*, 2025.

[83] P. Marbach and J. N. Tsitsiklis. Simulation-based optimization of markov reward processes. *IEEE Transactions on Automatic Control*, 46(2):191–209, 2002.

[84] B. Matérn. *Spatial variation*, volume 36. Springer Science & Business Media, 2013.

[85] H. Mehta. The l1 norms of de la vallée poussin kernels. *Journal of Mathematical Analysis and Applications*, 422(2):825–837, 2015.

[86] A. M. Metelli. *Exploiting environment configurability in reinforcement learning*, volume 361. IOS Press, 2022.

[87] A. M. Metelli, F. Mazzolini, L. Bisi, L. Sabbioni, and M. Restelli. Control frequency adaptation via action persistence in batch reinforcement learning. In *International Conference on Machine Learning*, pages 6862–6873. PMLR, 2020.

[88] 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.

[89] V. Mnih, K. Kavukcuoglu, D. Silver, A. A. Rusu, J. Veness, M. G. Bellemare, A. Graves, M. Riedmiller, A. K. Fidjeland, G. Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[90] A. Montenegro, M. Mussi, A. M. Metelli, and M. Papini. Learning optimal deterministic policies with stochastic policy gradients. *arXiv preprint arXiv:2405.02235*, 2024.

[91] A. Montenegro, F. Mansutti, M. Mussi, M. Papini, and A. M. Metelli. Reusing trajectories in policy gradients enables fast convergence. *arXiv preprint arXiv:2506.06178*, 2025.

[92] R. Munos. Error bounds for approximate value iteration. In *Proceedings of the National Conference on Artificial Intelligence*, number 2, page 1006. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2005.

[93] S. Mysore, B. Mabsout, R. Mancuso, and K. Saenko. Regularizing action policies for smooth control with reinforcement learning. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1810–1816. IEEE, 2021.

[94] Z. Németh. De la vallée poussin type approximation methods. 2016.

[95] J. R. Norris. *Markov chains*. Number 2. Cambridge university press, 1998.

[96] R. Ortner and D. Ryabko. Online regret bounds for undiscounted continuous reinforcement learning. *Advances in Neural Information Processing Systems*, 25, 2012.

[97] R. Ouhamma, D. Basu, and O. Maillard. Bilinear exponential family of mdps:

frequentist regret bound with tractable exploration & planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 9336–9344, 2023.

[98] A. Pacchiano, M. Phan, Y. Abbasi Yadkori, A. Rao, J. Zimmert, T. Lattimore, and C. Szepesvari. Model selection in contextual stochastic bandit problems. *Advances in Neural Information Processing Systems*, 33:10328–10337, 2020.

[99] M. Papini, M. Pirotta, and M. Restelli. Smoothing policies and safe policy gradients. *Machine Learning*, 111(11):4081–4137, 2022.

[100] M. Pei, H. An, B. Liu, and C. Wang. An improved dyna-q algorithm for mobile robot path planning in unknown dynamic environment. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 52(7):4415–4425, 2021.

[101] M. Pirotta, M. Restelli, and L. Bascetta. Policy gradient in lipschitz markov decision processes. *Machine Learning*, 100:255–283, 2015.

[102] R. F. Prudencio, M. R. Maximo, and E. L. Colombini. A survey on offline reinforcement learning: Taxonomy, review, and open problems. *IEEE Transactions on Neural Networks and Learning Systems*, 35(8):10237–10257, 2023.

[103] M. L. Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

[104] E. Rachelson and M. G. Lagoudakis. On the locality of action domination in sequential decision making. 2010.

[105] T. Ren, T. Zhang, C. Szepesvári, and B. Dai. A free lunch from the noise: Provable and practical exploration for representation learning. In *Uncertainty in Artificial Intelligence*, pages 1686–1696. PMLR, 2022.

[106] W. Rudin. Real and complex analysis, mcgraw-hill. *Inc.,*, 1974.

[107] D. Russo and B. Van Roy. Eluder dimension and the sample complexity of optimistic exploration. *Advances in Neural Information Processing Systems*, 26, 2013.

[108] S. Salsa and G. Verzini. *Partial differential equations in action: from modelling to theory*, volume 147. Springer Nature, 2022.

[109] S. Schaal. Learning from demonstration. *Advances in neural information processing systems*, 9, 1996.

[110] H.-J. Schmeisser and H. Triebel. Topics in fourier analysis and function spaces. *(No Title)*, 1987.

[111] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz. Trust region policy optimization. In *International conference on machine learning*, pages 1889–1897. PMLR, 2015.

[112] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

[113] M. H. Schultz. $l^{\infty}$-multivariate approximation theory. *SIAM Journal on Numerical Analysis*, 6(2):161–183, 1969.

[114] J. Silveira, J. A. Marshall, and S. N. Givigi. A simulation pipeline to facilitate real-world robotic reinforcement learning applications. In *2025 IEEE International systems Conference (SysCon)*, pages 1–8. IEEE, 2025.

[115] D. Silver, A. Huang, C. J. Maddison, A. Guez, L. Sifre, G. Van Den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484–489, 2016.

[116] M. Simchowitz and D. Foster. Naive exploration is optimal for online lqr. In *International Conference on Machine Learning*, pages 8937–8948. PMLR, 2020.

[117] S. R. Sinclair, S. Banerjee, and C. L. Yu. Adaptive discretization for episodic reinforcement learning in metric spaces. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3):1–44, 2019.

[118] S. Smolyak. Quadrature and interpolation formulas for tensor products of certain classes of functions. In *Soviet Math. Dokl.*, volume 4, pages 240–243, 1963.

[119] S. Sobolev. *Partial differential equations of mathematical physics*, volume 56. Courier Corporation, 1964.

[120] Z. Song and W. Sun. Efficient model-free reinforcement learning in metric spaces. *arXiv preprint arXiv:1905.00475*, 2019.

[121] E. M. Stein. *Singular integrals and differentiability properties of functions*. Number 30. Princeton university press, 1970.

[122] C. J. Stone. Optimal global rates of convergence for nonparametric regression. *The annals of statistics*, pages 1040–1053, 1982.

[123] R. S. Sutton. Dyna, an integrated architecture for learning, planning, and reacting. *ACM Sigart Bulletin*, 2(4):160–163, 1991.

[124] R. S. Sutton, A. G. Barto, et al. *Reinforcement learning: An introduction*. Number 1. MIT press Cambridge, 1998.

[125] W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[126] A. N. Tikhonov and A. A. Samarskii. *Equations of mathematical physics*. Courier Corporation, 2013.

[127] A. Tirinzoni, A. Al Marjani, and E. Kaufmann. Near instance-optimal pac reinforcement learning for deterministic mdps. *Advances in neural information processing systems*, 35:8785–8798, 2022.

[128] A. Tirinzoni, A. Al-Marjani, and E. Kaufmann. Optimistic pac reinforcement learning: the instance-dependent view. In *International Conference on Algorithmic Learning Theory*, pages 1460–1480. PMLR, 2023.

[129] V. Tkachuk, G. Weisz, and C. Szepesvári. Trajectory data suffices for statistically efficient learning in offline rl with linear $q^\pi$-realizability and concentrability. *Advances in Neural Information Processing Systems*, 37:83268–83313, 2024.

[130] A. Touati, A. A. Taiga, and M. G. Bellemare. Zooming for efficient model-free reinforcement learning in metric spaces. *arXiv preprint arXiv:2003.04069*, 2020.

[131] A. B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer, 2009.

[132] R. Tuo and C. Jeff Wu. A theoretical framework for calibration in computer models: Parametrization, estimation and convergence properties. *SIAM/ASA Journal on Uncertainty Quantification*, 4(1):767–795, 2016.

[133] S. Vakili. Open problem: Order optimal regret bounds for kernel-based reinforcement learning. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 5340–5344. PMLR, 2024.

[134] S. Vakili and J. Olkhovskaya. Kernelized reinforcement learning with order optimal regret bounds. *arXiv preprint arXiv:2306.07745*, 2023.

[135] S. Vakili and J. Olkhovskaya. Kernelized reinforcement learning with order optimal regret bounds. *arXiv preprint arXiv:2306.07745*, 2024.

[136] S. Vakili, N. Bouziani, S. Jalali, A. Bernacchia, and D.-s. Shiu. Optimal order simple regret for gaussian process bandits. *Advances in Neural Information Processing Systems*, 34:21202–21215, 2021.

[137] S. Vakili, K. Khezeli, and V. Picheny. On information gain and regret bounds in gaussian process bandits. In *International Conference on Artificial Intelligence and Statistics*, pages 82–90. PMLR, 2021.

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

[139] H. Van Seijen, H. Van Hasselt, S. Whiteson, and M. Wiering. A theoretical and empirical analysis of expected sarsa. In *2009 ieee symposium on adaptive dynamic programming and reinforcement learning*, pages 177–184. IEEE, 2009.

[140] L. R. Volevich and B. Paneyakh. Certain spaces of generalized functions and embedding theorems. *Russian Mathematical Surveys*, 20(1):1–73, 1965.

[141] B. Wang, Y. Yan, and J. Fan. Sample-efficient reinforcement learning for linearly-parameterized mdps with a generative model. *Advances in neural information processing systems*, 34:23009–23022, 2021.

[142] R. Wang, R. R. Salakhutdinov, and L. Yang. Reinforcement learning with general value function approximation: Provably efficient approach via bounded eluder dimension. *Advances in Neural Information Processing Systems*, 33:6123–6135, 2020.

[143] Y. Wang, S. Balakrishnan, and A. Singh. Optimization of smooth functions with noisy observations: Local minimax rates. *Advances in Neural Information Processing Systems*, 31, 2018.

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

[145] G. Weisz, P. Amortila, and C. Szepesvári. Exponential lower bounds for planning in mdps with linearly-realizable optimal action-value functions. In *Algorithmic Learning Theory*, pages 1237–1264. PMLR, 2021.

[146] G. Weisz, A. György, and C. Szepesvári. Online rl in linearly $q^\pi$-realizable mdps is as easy as in linear mdps if you learn what to ignore. *Advances in Neural Information Processing Systems*, 36:59172–59205, 2023.

[147] J. Whitehouse, A. Ramdas, and S. Z. Wu. On the sublinear regret of gp-ucb. *Advances in Neural Information Processing Systems*, 36:35266–35276, 2023.

[148] R. J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3):229–256, 1992.

[149] R. Wu, A. Sekhari, A. Krishnamurthy, and W. Sun. Computationally efficient

rl under linear bellman completeness for deterministic dynamics. *arXiv preprint arXiv:2406.11810*, 2024.

[150] Z. Yang, C. Jin, Z. Wang, M. Wang, and M. I. Jordan. Provably efficient reinforcement learning with kernel and neural function approximations. In *NeurIPS*, 2020.

[151] A. Zanette, A. Lazaric, M. Kochenderfer, and E. Brunskill. Learning near optimal policies with low inherent bellman error. In *International Conference on Machine Learning*, pages 10978–10989. PMLR, 2020.

[152] W. Zhang, D. Zhou, L. Li, and Q. Gu. Neural thompson sampling. *arXiv preprint arXiv:2010.00827*, 2020.

[153] Z. Zhang, Y. Chen, J. D. Lee, and S. S. Du. Settling the sample complexity of online reinforcement learning. In *The Thirty Seventh Annual Conference on Learning Theory*, pages 5213–5219. PMLR, 2024.