Toward realistic reinforcement learning

Reda Ouhamma

Université de Lille, faculté des Sciences et Technologies
Ecole Doctorale des Mathématiques-Sciences du numérique et de leurs interactions

Thèse de Doctorat

Spécialité **Informatique**

présentée par
Reda Ouhamma

Toward Realistic Reinforcement Learning

Apprentissage par Renforcement Réaliste

sous la direction d'**Odalric-Ambrym Maillard**
et de **Vianney Perchet**.

Soutenue publiquement à **Villeneuve d'Ascq**, le **14 avril 2023** devant le jury composé de

M. Olivier **Cappé**	Directeur de recherche, Université PSL	Président du Jury
M. Olivier **Wintenberger**	Professeur, Sorbonne Univesité	Rapporteur
M. Aurélien **Garivier**	Professeur, Ecole Normale Supérieure de Lyon	Rapporteur
M^{me} Alexandra **Carpentier**	Professeur, Universität Postdam	Examinatrice
M^{me} Shipra **Agrawal**	Professeur associé, Columbia University	Examinatrice
M. Aditya **Gopalan**	Professeur associé, IISC	Examinateur
M. Odalric-Ambrym **Maillard**	Chargé de recherche, Inria	Directeur de thèse
M. Vianney **Perchet**	Professeur, Crest / ENSAE	Co-directeur de thèse

Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL),
UMR 9189 Équipe Scool, 59650, Villeneuve d'Ascq, France

À mon père, Omar, et mon grand père, El Mekki,
qui ont démontré par l'exemple, que le travail paie.

Résumé

Dans cette thèse de doctorat, nous considérons le défi de rendre l'apprentissage par renforcement (RL) plus adapté aux problèmes du monde réel sans perdre les garanties théoriques. Il s'agit d'un domaine de recherche très actif, car l'application au monde réel est l'objectif final de cette littérature ainsi que la motivation première des cadres spécifiques de l'apprentissage par renforcement. Les garanties théoriques sont, comme leur nom l'indique, l'assurance que la théorie peut fournir sur la performance et la fiabilité de nos stratégies. Le développement de ce domaine est crucial pour améliorer les algorithmes de RL interprétables. Notre travail est structuré autour de quatre contextes différents, nous commençons par une introduction au domaine et une revue générale de la littérature, y compris les bandits, les processus de Markov (MDP), certains objectifs d'apprentissage par renforcement, et quelques défis de RL réaliste.

La thèse se poursuit en spécifiant divers scénarios spécifiques ainsi que différentes approches pour relever quelques défis pertinents du RL. Nous nous attaquons d'abord à un scénario séquentiel d'identification de signe pour les bandits à bras multiples, où nous concevons une méthode générique pour définir des algorithmes, une nouvelle stratégie de preuve fournissant des limites d'erreur. Ensuite, nous présentons de nouvelles observations comparant les algorithmes adaptatifs aux oracles hors ligne. Notre deuxième contribution est une amélioration théorique de la régression linéaire séquentielle pour des limites de regret améliorées et une stabilité accrue, nous nous sommes inspirés de résultats bien établis et les avons adaptés au cadre stochastique, puis nous avons illustré les améliorations avec une application aux bandits linéaires. Une contribution significative de cette thèse est l'étude de la récente représentation de la famille exponentielle bilinéaire pour les MDPs à espaces continus. Nous avons pu faire des observations notables menant à des solutions explicites et à des garanties théoriques améliorées. Enfin, nous nous sommes attaqués au problème des gradients de politiques profondes où nous avons introduit une mesure d'erreur bien justifiée pour un apprentissage plus précis de la fonction de valeur. Le besoin de cette dernière amélioration a été fortement motivé par des travaux récents ainsi que par plusieurs expérimentations que nous avons fournies.

Les résultats de cette thèse démontrent un progrès dans la littérature RL, tant sur le plan pratique que théorique, offrant des perspectives et des solutions précieuses pour la communauté RL. Nous pensons que les méthodes proposées font partie des solutions pour combler le fossé entre la théorie du RL et ses applications, faisant de cette thèse une contribution significative au domaine.

Abstract

This thesis explores the challenge of making reinforcement learning (RL) more suitable to real-world problems without losing theoretical guarantees. This is an interesting active research area because real-world problems are the final goal and the first motivation for the different RL settings, and theoretical guarantees are like their name suggests, the assurances that the theory can provide about the performance and reliability of our strategies. Developing this field is crucial for improving interpretable RL algorithms. Our work is structured around four different RL settings, and begins with an introduction to the field and a general review of relevant literature, including bandits, Markov Decision Processes (MDPs), a number of reinforcement learning objectives, and relevant realistic RL challenges.

The thesis proceeds by specifying various specific scenarios as well as different approaches to address the relevant RL challenges. We first tackle an online sign identification setting for multi-armed bandits, where we investigate a generic method to design algorithms, a novel proof strategy providing SOTA error bounds, and we present unprecedented observations when comparing adaptive algorithms to offline oracles. Our second contribution is a theoretical improvement of sequential linear regression for improved regret bounds and increased stability, we took inspiration from well established results that we adapted to the stochastic setting, we illustrated the improvements with an application to linear bandits. Another significant contribution of this thesis is studying the recent bilinear exponential family representation for continuous MDPs, we were able to make notable observations leading to tractability and improved theoretical guarantees. Finally, we tackled the setting of deep policy gradients where we introduced a principled loss for a more accurate value function learning, the need for this improvement was strongly motivated by recent work as well a several experiments that we provided.

The results of this research demonstrate substantial progress in the RL literature both practically and theoretically, offering valuable insights and solutions for the RL community. We believe that the proposed methods show the potential to close the gap between purely theoretical RL and applications-motivated RL, making this thesis a significant contribution to the field.

To the layman reader

If you can't explain it to a six-year-old,
you don't understand it yourself
Albert Einstein.

We are interested in a universal task, one that regularly faces every living entity, making decisions in sequential settings. Humans tackle such problems on a daily basis, and with varying importance or possible consequences. From choosing one's clothes every morning to deciding which career to pursue, and from impacting the personal comfort through the day to determining the future quality of life.

Naturally, the decisions must be made for a purpose, and can be part of a bigger plan. This so called purpose is a subjective concept that is almost never identical for different interacting entities, mathematically, it is commonly represented using a utility function that an entity seeks to maximize. This mapping is supposed to encode the entity's preferences and the impact of its choices. In this context, various problems with different complexities and specifities give rise to numerous possible formulations.

As part of our motivation for this manuscript, we shall consider a tangible example that speaks to everyone, concertizes our vision, and allows us deduce generic principles that will inspire the different problems we wish to tackle. For this purpose, I will examine the closest example to my personal life: the PhD student's journey.

Reinforcement learning Classical machine learning involves training a model to make predictions or decisions based on a fixed set of data. The goal is to build a model that is able to generalize to new, unseen data. In the latter, learning is based on the input data and the corresponding outputs that have been provided to it. Reinforcement learning, on the other hand, is interaction-based and the data is provided sequentially instead of a batch manner.

In simple terms, reinforcement learning is a way for a computer or robot to learn how to do something by trying different things and seeing which ones work the best. Imagine you're playing a game where you have to find the treasure. The computer or robot is like a little

explorer trying to find the treasure, and it gets a reward every time it finds the treasure. The more treasure it finds, the more it will want to keep playing the game. The computer or robot will try different things to find the treasure, and if it does something that helps it find the treasure more often, it will keep doing that. If it does something that doesn't help it find the treasure, it will try something else. Through this process, the computer or robot will learn how to find the treasure more efficiently.

The case for RL instead of ML One of the key promises of RL is that it allows learners to improve through experience, without requiring explicit programming of rules or behaviors. This can be particularly useful in complex or dynamic environments, where it may be difficult to pre-define a fixed set of rules or behaviors. RL algorithms can adapt to changing environments and learn to make optimal decisions based on the feedback they receive.

Another promise of RL is that it has the potential to enable autonomous systems to learn and adapt to their surroundings in real-time. This can be beneficial in a variety of applications, such as self-driving cars, which need to be able to respond to changing traffic conditions and make decisions that maximize safety and efficiency.

Overall, the promise of RL is to enable agents to learn and adapt to their environments in order to optimize their performance and achieve their goals.

A primer on RL terminology We define some necessary keywords of the RL jargon.

First, we define the *environment*, *i.e.* a context in which the learner operates and makes decisions. For example, the environment could be a physical space, such as an inverted pendulum that a person controls using their hands. Next, an *action* is a decision taken by the learner. For example, the action of a person to push the pendulum forward, backward, or to not push it. A *state* is the current situation or context in which the learner finds itself, like the angle or the abscissa of our pendulum. The state may include information about the environment, like the wind, the current velocity, or the person's past decisions. The learner is called an *agent* in RL, an entity that is learning and taking actions in an environment. Like the person that controls the inverted pendulum in our example.

Then, let's define some terms that encode the environment or agent's reactions and decisions. For instance, a *policy* is a strategy or set of rules that the agent uses in its interaction with the environment. The latter can be deterministic or randomized, *e.g.* an agent can play each possible action with a given probability. The reaction of the environment to an action in a given state is called a *reward*. A reward is the outcome or consequence of an action taken by the agent, it provides feedback to encourage or discourage the agent from taking specific actions.

Finally, we define the most important quantity in reinforcement learning. The *value function* is a measure of the expected long-term reward or utility of a given state or action. The value function helps the agent decide which actions are most likely to lead to good long-term outcomes. The value function is defined in conjunction with a policy of the agent, and it helps the latter evaluate potential consequences of different actions and choose the ones that most likely lead to great rewards.

To summarize, we say that an agent interacts with an environment, decides which actions to play using their policy, receives a reward and moves to a new state. An agent represents the efficacy of a given policy by estimating its corresponding value function. Based on the value estimate, the agent updates their policy when they deem it appropriate and receives new rewards as feedback from the environment on how to improve their decision making process.

RL challenges RL is still an open field with many open questions, which is fortunate for us. In the following, we enumerate and explain the main challenges facing an RL agent through a concrete example.

Exploration-exploitation Consider an innocent undergraduate, embarking in the adventure that is a PhD, full of hope and happy to finally have the freedom of doing research in their favorite field. The latter is high spirited and doesn't know how to spend their energy, they are thus confronted with their first (of many) dilemma: probe the literature in depth or parse relevant papers and pursue a first result. While the former translates to spending months in exploring and searching for pertinent research, the latter can quickly lead to publishable content and to harnessing the student's strengths. This compromise is at the heart of the reinforcement learning challenge and is called the exploration-exploitation trade-off.

Figure 1 – Answer of DALL.E 2 for the prompt "PhD student in the wild, drawing style"

Definition. In reinforcement learning, the *exploration-exploitation trade-off* is the fundamental dilemma faced by a learner trying to learn about its environment and maximize its profit. On one hand, the learner needs to explore, or try out different actions and be in new states, in order to learn more about the environment and improve its decision-making. On the other hand, the agent also needs to exploit, or take advantage of what it has learned so far, in order to maximize its profit in the short term.

The exploration-exploitation trade-off arises because exploration can be risky and may not lead to an immediate payoff, while exploitation allows the agent to maximize its profit

based on its current learner, but may not lead to long-term learning or optimization. As a result, the learner must find a balance between exploration and exploitation in order to learn effectively and maximize its reward over time.

Our PhD candidate faces the exploration-exploitation trade-off from the beginning of their studies, for instance, they must balance their desire to utilize known techniques to improve existing work whit their mission to learn key techniques from their area of interest. Therefore, this tension between exploration and exploitation manifests itself from the first months of their research.

Continuous spaces Let's say that our PhD student has now spent enough time to become well-informed about their chosen field. And let's assume for our case that their advisor has given them the freedom to choose their specific topic. The second challenge that confronts our candidate is selecting one specific subject. For instance, a scholar interested in RL has to choose from multi-armed bandits, structured bandits (linear, Lipschitz, generalized...), tabular MDPs, continuous MDPs, etc. And that for all of the latter, they can choose between stationary or non-stationary, theory or practice, regret minimization or pure exploration and much more. Agreed a layman reader wouldn't understand these technical terms, yet they would surely grasp the frustration that a junior candidate is faced with when they are spoiled for choice to this extent.

Definition. One major difficulty in sequential decision making is *scaling the strategies to continuous spaces*, *i.e.* environments with large or infinite number of states and actions. Indeed, it can be very challenging to handle the exploration-exploitation dilemma in continuous spaces. Indeed, balancing this trade-off may require the agent to explore a large (possibly infinite) number of states in order to learn about the environment effectively.

Function approximation Faced with the aforementioned challenge, it may be impractical for our PhD candidate to evaluate every possible course of action explicitly. In these cases, a popular solution is to resort to function approximation techniques. Our scholar can assume some kind of model for the rewards and or transitions, then they can optimize their policy by learning in a functional space. The latter can be easier and is especially interesting in the case of parametric spaces or functions with particular structures.

Definition. *Function approximation* in RL refers to the setting where some functional representation is assumed to approximate the value function or the policy. Major challenges

arise with the latter. For instance, these techniques can be computationally intensive if the class is overly intricate, do not necessarily provide accurate approximations if the class is overly simplistic, and may even be unrealistic.

Computational efficiency Tractability is an important consideration in machine learning, indeed algorithms often involve complex computations. In RL, computational efficiency is even more regnant since algorithms almost always involve iterative processes, such as updating an estimator of the value function or re-optimizing the agent's policy after new observations.

Definition. *Computational efficiency* is directly correlated to the performance and practicality of an RL algorithm in real-world scenarios. Unfortunately, and due to the nature of the setting, RL algorithms may need to update/optimize certain functions a number of times proportional to the amount of data. Therefore, RL methods can be computationally intensive and those with theoretical guarantees are often intractable. Accordingly, computational efficiency constitutes a key challenge in the development and application of RL algorithms, and can seriously affect the impact and popularity of this literature.

Other RL challenges RL is a very active field of research and has been applied to a different practical domains, *e.g.* robotics, advertisements, games, etc. Consequently, every potential application in this dynamic literature brings up new concerns and challenges specific to the particular setting's purpose. A common issue is the *credit assignment*, *i.e.* how to assign rewards to particular actions when a series of decisions has been made. Handling *non-stationary environments* is another issue that comes up often as the world is naturally non-stationary. Also, when used to solve real world problems, the issue of *sample complexity* arises, indeed RL algorithms may need a consequential amount of interactions before learning, and this can be very limiting in real life. Finally, an essential challenge in the deployment of RL algorithms is the *safety* and reliability of the agent. Indeed, a self-driving car cannot be trusted to drive in the real world if it's learning its policy from scratch and without prior knowledge.

Overall, the literature on RL is large and diverse, and as it continues to grow and evolve, researchers and practitioners raise new difficulties in conjunction with their respective applications. The latter is beneficial and tackling such challenges head on advances the state of the art and practicality of RL. The PhD student's example and many other applications of RL that ordinary mortals face consistently are a major personal incentives in this thesis.

Conclusion It is particularly frustrating for our PhD student when they recognize that RL frameworks have the ability to address their conundrum and many other important challenges,

yet the applicability of RL is hindered by practical limitations. It is also disheartening when certain RL problems are deemed resolved even though the theoretical framework is not well-aligned with reality.

Drawing inspiration from the previous examples and limitations, my particular interest in this thesis is to improve the practicality of RL frameworks and algorithms. In other words, I aimed to increase the applicability of RL algorithms by working towards reducing stringent assumptions, limiting the need for extensive prior knowledge, effectively utilizing available information, and ultimately replacing flawed structures or procedures with more realistic equivalents that better reflect the complexity of real-world scenarios. This way, my purpose is to make RL algorithms and theory more widely applicable to real-world problems.

Contents

List of Acronyms

C

CMDP	Constrained Markov Decision Process

D

DP	Dynamic Programming
DQN	Deep Q-Network

I

i.i.d.	independent and identically distributed

M

MAB	Multi-Armed Bandits
MCTS	Monte-Carlo Tree Search
MDP	Markov Decision Process
MLE	Maximum Likelihood Estimation

O

OFU	Optimism in the Face of Uncertainty

P

PAC	Probably Approximately Correct

POMDP Partially Observable Markov Decision Process

R

RL Reinforcement Learning

List of Symbols

Mathematical notations

$\mathbb{N}$	set of integers
$[n]$	range of integers $\{1, \ldots, n\}$
$\mathbb{R}_+$	set of positive reals $\{\tau \in \mathbb{R} : \tau \geq 0\}$
$\mathbb{R}$	set of real numbers
$\mathbb{N}_+$	set of positive integers $\mathbb{N} \cap \mathbb{R}_+$
$\|x\|$	Euclidean norm for a vector $x \in \mathbb{R}^n$
$\|z\|$	absolute value $\|z\| = z^+ + z^-$
z^-	negative part $z^- = z^+ - z$
z^+	positive part $\max(z, 0)$
e_i	normal basis vectors $[0 \ldots 0\ 1\ \ldots 0]^\top$ in $\mathbb{R}^n$ for $i \in [n]$, where 1 appears in the i^{th} position
I_n	the identity matrix with dimension $n \times n$
$M^\top$	transpose of a matrix M
$\|A\|_2$	the induced L_2 matrix norm $\sqrt{\max_{i \in [n]} \lambda_i(A^\top A)}$
$P \succ 0$	a symmetric matrix $P \in \mathbb{R}^{n \times n}$ is positive definite
$o(\cdot), \mathcal{O}(\cdot), \Omega(\cdot)$	Landau notations for positive functions: $f(x) = o(g(x)$ means that $g(x) \neq 0$ and $f(x)/g(x) \to 0$ for $x \to \infty$, $f(x) = \mathcal{O}(g(x))$ means that there exists $x_0, K > 0$ such that $f(x) \leq Kg(x)$ from $x \geq x_0$, and $f(x) = \Omega(g(x))$ means $g(x) = \mathcal{O}(f(x))$
$\mathbb{E}$	expectation under a probabilistic model
$\mathcal{P}_\mathcal{X}$	set of probability measures on a measurable space $\mathcal{X}$

$\mathbb{V}$	variance under a probabilistic model

Markov Decision Processes

$\mathcal{M}$	an MDP instance
$\mathcal{S}$	set of states $s \in \mathcal{S}$, 2
$\mathcal{A}$	set of actions $a \in \mathcal{A}$, 2
$R(s,a)$	reward function $R : s, a \to R(s,a) \in [0,1]$
$P(s' \mid s,a)$	transition distribution $s' \sim P(s' \mid s,a)$
π	policy
$\pi^\star$	optimal policy
V	state value function (* for optimal value, $^\pi$ for policy value) , 5
Q	state-action value function (* for optimal value, $^\pi$ for policy value) , 84
$\mathcal{T}$	Bellman operator (* for optimality, $^\pi$ for evaluation)
$\mathcal{D}$	dataset

Multi-armed bandits

μ_a	reward vector (or function), $\mu_a : \mathcal{A} \to \mathbf{R}$

Linear regression

x_t	Coordinate presented to the agent at time t
y_t	Label of the t'th coordinate
θ^*	$\in \mathbf{R}^d$, True parameter of the linear model
θ_t	Estimation of the parameter at time t
$\mathcal{C}_t$	High probability confidence interval for the parameter at time t
λ	Regularization parameter
Λ_t	Design matrix at time t, $\Lambda_t := \lambda I + \sum_{s=1}^{t} x_s x_s^\top$

Chapter 1

Introduction

1.1 Motivation and theoretical formalism

We are interested in sequential decision making, where a learner interacts with some environment and must take a sequence of decisions in order, such that the outcome of each decision may influence the options available for subsequent decisions. In this context, various specific problems with different complexities and diverse objectives give rise to numerous possible formulations. In particular, we are interested in the reinforcement learning framework, a sub-field of machine learning where data is not provided beforehand but is revealed sequentially as the learner interacts with the environment.

In this Chapter, we will clarify the main concepts of our chosen theoretical frameworks, then we will enumerate and provide high-level intuition for the different objectives studied in this manuscript. Finally, we will describe in further detail the tasks that are studied, and give a summary of our contributions.

Reinforcement learning is a type of machine learning that involves trial and error interactions with an environment, *i.e.* learning from the consequences of actions. More precisely, the learner receives rewards or punishments for its actions, and it adjusts its behavior accordingly in order to maximize the reward.

In reinforcement learning, the learner tries to maximize its utility by taking decisions that lead to favorable outcomes. The agent learns to choose actions that will lead to the greatest reward over time, based on its past experiences and the feedback it receives from the environment. Reinforcement learning has been applied to a variety of applications, including natural language processing, robotics, and recommendation systems.

1.1.1 Markov decision process

The MDP framework is the most popular for reinforcement learning. In this structure, the learner that interacts with the world is called an *agent*, the agent's available decisions are named *actions*, and the utility that a decision achieves is called a *reward*. The literature studies reinforcement learning mathematically as a Markov decision processes (MDP) where the agent i) chooses actions based on past observations ii) observes an immediate reward sampled from an unknown function, and iii) moves to a new state following an unknown Markovian transition mapping. The latter implies that given the current state and action, the next state is independent of previous state-action pairs. In layman's terms, we say that a transition is Markovian if the future depends only on the present state and not on the past history.

Infinite-horizon RL A popular formulation of RL is the infinite-horizon Markov Decision Problem (MDP) with -possibly- continuous states $s \in \mathcal{S}$, actions $a \in \mathcal{A}$, transition distribution $s_{t+1} \sim \mathbf{P}(. \mid s_t, a_t)$ and reward function $r_t \sim r(s_t, a_t)$. Let $\pi(a|s)$ denote a stochastic policy, the agent repeatedly interacts with the environment by sampling action $a_t \sim \pi(.|s_t)$, receives reward r_t and moves to a state s_{t+1}.

In Infinite-horizon RL, the value of a policy in a given state is measured as:

$$V^{\pi}(s) \triangleq \mathbb{E}_{\tau \sim \pi}\left[\sum_{t=0}^{\infty} \gamma^t r_t | s_0 = s\right],$$

where $\gamma \in [0, 1)$ is a discount factor accounting for the preference of present rewards over -possibly larger- future ones. This value function evaluates the performance of an agent and is often linked to the objective even though the latter can vary depending on the application.

Episodic RL Another prominent formulation for RL (Osband, Russo, and Van Roy, 2013; Azar, Osband, and Munos, 2017; Dann, Lattimore, and Brunskill, 2017), Episodic RL is a tuple $\mathcal{P} = \langle \mathcal{S}, \mathcal{A}, \mathbf{P}, r, K, H \rangle$, where the state (resp. action) space $\mathcal{S}$ (resp. $\mathcal{A}$) might be continuous. In episodic RL, the agent interacts with the environment in $K \in \mathbf{N}$ episodes consisting of $H \in \mathbf{N}$ steps. Episode k starts by observing state s_1^k. Then, for $t = 1, \dots H$, the agent draws action a_t^k from a (possibly time-dependent) policy $\pi_t(s_t^k)$, observes the reward $r_t \sim r(s_t^k, a_t^k) \in [0, 1]$, and moves to a new state $s_{t+1}^k \sim \mathbf{P}(. \mid s_t^k, a_t^k)$ according to the transition function $\mathbf{P}$.

Roughly speaking, episodic RL was introduced as a way of breaking down the RL problem into smaller problems, where the agent learns from the rewards they receive at the end of each episode. This allows the agent to focus on the short-term decisions rather than the long-term ones and makes the problem more manageable. It is also more interesting in practice due to

the reduced amount of updates to the policy, which is usually correlated with an improved stability of the agent's performance.

In episodic RL, the performance of a policy π is measured by the total expected reward V_1^π starting from a state $s \in \mathcal{S}$, the value function and the state-action value functions at step $h \in [H]$ are defined as

$$V_h^\pi(s) \triangleq \mathbf{E}\left[\sum_{t=h}^{H} r_t \mid s_h = s\right], \quad Q_h^\pi(s,a) \triangleq \mathbf{E}\left[\sum_{t=h}^{H} r_t \mid s_h = s, a_h = a\right].$$

Multi-armed bandits This is a special case of the MDP model where there is only one state, *i.e.* without a transition structure. In other terms, the agent chooses actions, observes the corresponding reward signals and repeats the process again, *i.e.* it stays in the same state.

The multi-armed bandits setting is classically motivated by slot machines, also known as one-armed bandits, of which the setting got its name. Given a fixed amount of coins T and a number K of slot machines (arms) with possibly different rewards, the agent allocates their wealth sequentially between the machines in order to gain as much money possible, under the assumption that each machine is characterized by a different payoff. Mathematically, arm $k \in \{1, \dots, K\}$ is modeled by a distribution of probability D_k, with mean μ_k. At time t, the agent picks an arm j and receives reward r_t sampled from the distribution D_j. In an abstract sense, the agent seeks to identify the arm with the highest mean in the shortest amount of time.

1.1.2 Realistic reinforcement learning

Here we shed some light onto the subject of this thesis "towards realistic RL" and provide some high level insights to assure the reader of its relevance.

Reinforcement learning aims to solve a wide range of real-world problems, including natural language processing, robotics, and game playing. Therefore, it is only natural to expect these concrete problems to be plausible under the studied theoretical formalism and assumptions. For example, when training an autonomous car to navigate safely, we assume that the transition and reward models are well represented in an -efficiently- learnable model. In practice, applying reinforcement learning involves dealing with real-world challenges such as noisy or incomplete data, limited computational resources, and the need to balance exploration and exploitation without prior knowledge. As a result, the gap between reinforcement learning theory and practice can sometimes be significant, and bridging this gap requires a combination of theoretical understanding and practical experience.

For previous reasons and more, we make seeking *realistic* practices in theoretical reinforcement learning a driving goal for this thesis. This purpose is visible in our work with different

levels of wingspan, and we believe that the journey of *realistic RL* is still long and contains several -deceptively small- building blocks and contributions.

1.2 Reinforcement learning objectives

There are several different performance measures that can be used to evaluate the effectiveness of an agent in reinforcement learning. One common measure is the expected return, which is the sum of the discounted rewards that an agent receives over time. Another measure is the cumulative reward, which is the total reward received by an agent over a certain period of time. Other measures include the average reward per time step, which reflects the average reward received by an agent at each time step, and the average reward per episode, which reflects the average reward received by an agent over the course of an episode. These performance measures can be used to compare the performance of different agents or to assess the performance of a single agent over time.

1.2.1 Regret minimization

Regret minimization is a concept in reinforcement learning that refers to the idea of computing the policy leading to maximization of cumulative reward. The latter requires the agent to strategically control the actions in order to learn the transition and reward functions to a sufficient level of precision. This tension between learning the unknown environment and reward maximization is quantified as *regret*: the typical performance measure of an episodic RL algorithm. *Regret* is defined as the difference between the *expected cumulative reward* or *value* collected by the optimal agent that knows the environment and the expected cumulative reward or value obtained by an agent that has to learn about the unknown environment. Formally, the regret over K episodes is

$$\mathcal{R}(K) \triangleq \sum_{k=1}^{K} \left(V_1^{\pi^\star}(s_1^k) - V_1^{\pi_t}(s_1^k) \right).$$

where π_t is the policy of the agent at episode t and $\pi^\star$ is the optimal policy, *i.e.* for all possible initial states $s \in \mathcal{S}$ we have $\pi^\star \in \arg\max_\pi V_1^\pi(s)$.

We choose to defer the other definitions of regret to their respective chapters, *e.g.* those relating to **Infinite-horizon RL** or to **Multi-armed bandits**. In essence, the objective is similar: minimizing the regret incurred by an agent in every setting. The differences are manifested in the structures of interaction or in the different definitions of the value functions.

1.2.2 Pure exploration

In reinforcement learning, pure exploration refers to situations where an agent's primary goal is to gather as much information about the world as possible, rather than maximizing the cumulative reward. In these settings, the environment may or may not reveal the rewards to the agent, and the goal is to learn as much as possible in order to return a policy that optimizes some criterion. The exploration and exploitation trade-off is also present in this setting, through the decision of whether to try a new action that may lead to a better reward or to continue using a known action to obtain a better estimate of its value and / or transition dynamics. Pure exploration settings can be useful for gathering data and understanding the structure of an environment.

To illustrate this objective, consider the **Best-policy identification** setting. In the latter, the agent observes both transitions and rewards to attempt returning an ε-optimal policy with large probability. In technical terms, if we denote V^π the value function of the agent following policy π and V^* the optimal value function then the objective is: given $\varepsilon \in \mathbf{R}_+$ and $\delta \in (0, 1)$, interact with the environment and return as fast as possible a policy π such that:

$$P\left(\forall s \in \mathcal{S}, \quad V^\pi(s) \geq V^*(s) - \varepsilon\right) \geq 1 - \delta.$$

To conclude, regret minimization and pure exploration are two pertinent objectives that continue to attract the attention of researchers in the reinforcement learning field. While in pure exploration an agent's primary goal is to gather as much information about its environment as possible through exploration, regret minimization rather entails maximizing the cumulative reward which can be more restrictive for the agent since it cannot explore possibly fruitless actions. In both cases, the agent must balance the trade-off between exploration and exploitation.

1.3 Outline and Contributions

The purpose of this section is to summarize the different contributions of this thesis in the fields of bandits, linear regression, and reinforcement learning.

Prior to outlining and elaborating our contributions, we provide an overview of our motivation at a high-level. In reality, our efforts and aspirations can be summarized in the question *"how can we make the RL framework, algorithms, and assumptions more realistic?"*.

It is only natural that we begin this effort by turning to the simplest model for sequential decision making: Multi-armed bandits. We consider the finite number of arms case in **Chapter 3** and study a pure exploration objective called Thresholding bandits. We propose a Frank-Wolfe based method, extend the setting to a wider class of losses, beat the state-of-the-art

both theoretically and empirically, and apply our generic proof scheme to improve existing algorithms. The findings of this Chapter were published at the *Neural Information Processing Systems* conference in 2021 as a spotlight paper (top 3%).

In **Chapter 4**, we consider the popular linear regression problem. We uncover an old result of the adversarial setting and adapt it to the stochastic case, we show that this algorithm should be the default one instead of ridge regression. We apply this modification to linear bandits and show that it enables removing an omnipresent assumption. Our analysis is useful both theoretically and for the practitioner. The results of this Chapter were accepted for publication at the *Neural Information Processing Systems* conference in 2021.

In order to progress, we study continuous state action MDPs in **Chapter 5**. Bewildered by the popularity of the unrealistic linear MDP model, we consider a recently proposed MDP representation. We recall the expressive power of the latter and unveil a novel and crucial property about it that enables us to design an algorithm with tractable planning. We also show several results of independent interest, *e.g.* we show that we can forgo clipping value functions therefore removing a superfluous step of non-linearity. The contributions in this Chapter were published in the *The AAAI Conference on Artificial Intelligence* in 2023 as an oral presentation.

In **Chapter 6**, we study deep policy gradients, an infinite-horizon MDP setting where the value function and policy are modeled using deep neural networks and optimized using gradient descent. Driven by recently unveiled pitfalls of existing algorithms, we show that a simple modification in the value function loss can lead to significant improvement. We first motivate the latter and provide an intuitive proof of concept, then we demonstrate empirically that 1) our method improves over the standard training loss, and 2) our intuition is indeed accurate and our predicted are corroborated by the simulations. The contributions presented in this Chapter were accepted for publication at the *International Conference on Learning Representations* in 2021.

To summarize, our contributions are organized as depicted in Figure 1.1, between improvements of existing methods and contributions that introduce novel principles and algorithms. Our goal through all of them is unique: contribute to the RL research by improving algorithms that are both theoretically sound and user-friendly for practitioners, and to support the advancement towards a more *realistic RL formalism*.

> **Remark 1.1.** *remarkbar The chapters in this thesis have been written to be standalone and can be read independently, catering to the diversity of settings studied. This allows for ease of readability and flexibility for the reader, who may only be interested in a specific topic without the need to go through the entire thesis.*

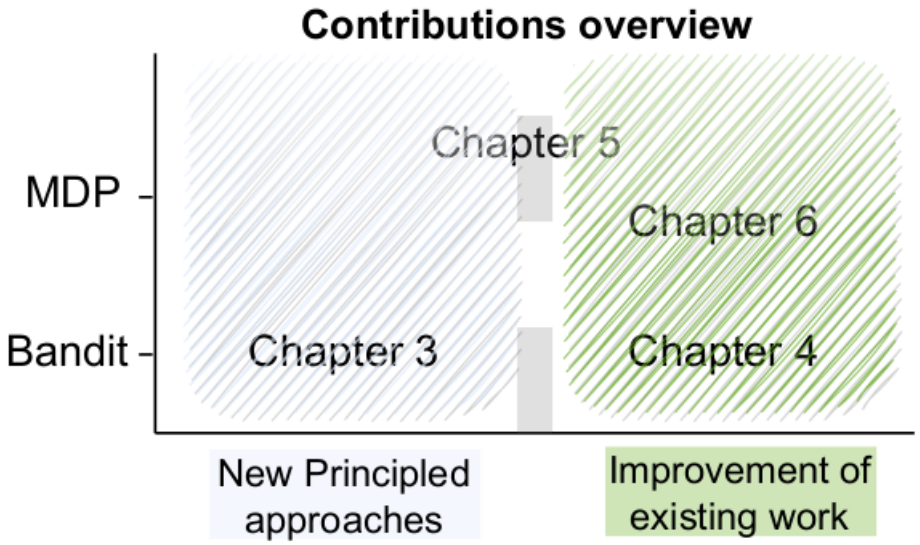


Figure 1.1 – Our contributions are divided by whether they improve existing algorithms and models or they propose entirely novel strategies. They are united in striving for reasonable structures.

1.3.1 Frank-Wolfe for thresholding bandits

In **Chapter 3**, we focus on a pure exploration setting called thresholding bandit. In the latter, a learner interacts with a K-armed bandit, nature provides it with a threshold and after a known time horizon it asks the learner to output their answer for "which arms' means are larger than the threshold". An agent is evaluated by the total number of mistakes it makes, weighted or not, after ending the interaction. In this game, the learner's loss depends on the unknown bandit parameters. Therefore, at every time-step, the agent needs to act with two independent objectives in mind: *improving the loss estimate*, and sampling arms that can possibly yield the most significant *loss reduction*. The latter is a quintessential challenge facing an agent in this setting, and is a manifestation of the famous exploration-exploitation trade-off.

This setting has been investigated in many recent papers and several algorithms were proposed with good empirical performances yet sub-optimal theoretical guarantees. We extend this bandit problem to more general loss functions, with possibly gap-dependent weights. Moreover, building on the popular Frank-Wolfe optimization approach, we are able to propose a generic method to design strategies for this setting. Our algorithms are (perhaps surprisingly) not based on typical RL concepts like optimism, explore-than-commit, nor successive elimination. Furthermore, we provide an simple and intuitive analysis that works for a wide class of index-based algorithms including all previously introduced ones for this setting. Our analysis not only yields to state-of-the-art theoretical guarantees for our algorithm but improves the bounds for existing algorithms as well. In addition, we evaluate our algorithm empirically and show how it outperforms existing techniques. Finally, we provide new non-trivial observations

about this setting, *e.g.* how adaptive-methods can empirically outperform non-adaptive oracles by a significant margin.

1.3.2 The forward algorithm and application to linear bandits

In our quest of tackling increasingly complex sequential decision making settings, in **Chapter 4** we are interested in studying bandit problems with a continuous action space. Several works in this direction make structural assumptions about the reward model, the most popular one being the linear rewards assumption. In the latter, it is also assumed that the reward function is bounded (usually in $[0, 1]$) and that the learner is aware of this. While considering bounded rewards is very reasonable, providing the bound to an agent before the interaction is restrictive in our opinion. We investigate methods of relaxing the latter by attacking the underlying linear regression challenge, specifically, we are interested in studying the stochastic linear regression setting without assuming a known bound on observations. We motivate this setting by highlighting the theoretical and empirical weaknesses of the standard ridge regression.

This problem has been solved for adversarial setting in the seminal paper (Vovk, 1997) by adopting the forward algorithm, which ingeniously adds a regularization term depending on the future coordinate. Naturally, we investigated this method in the stochastic setting to determine whether it also alleviates the "bound knowledge" assumption in this case. We answer the latter positively, and we show the true bounds of standard regression when stripped from knowing the range of observations. Furthermore, we show that this algorithm lends itself very conveniently to the linear bandit setting. Finally, we show through numerical experiments the success due to this simple modification. The latter is indeed a small adjustment, yet it paves the way towards realistic scenarios in which the observations' range is not known a priori.

1.3.3 MDPs with Continuous state and actions spaces

We take the pursuit of *realistic RL* challenge to the next level by attempting to advance the work on continuous state-action MDPs. In **Chapter 5**, we consider the episodic RL setting in which an agent interacts with the environment in $K \in \mathbf{N}$ epochs of length $H \in \mathbf{N}$. A learner in this framework must update its policy at the beginning of every epoch based on past information only. Currently, the main challenge in this setting is the representation choice. Indeed, while linear MDPs model has become ubiquitous in literature, it remains a purely abstract assumption that only helps the analysis without any concrete example or application.

In this perspective, we investigate a recently introduced representation based on the exponential family of distributions (Chowdhury, Gopalan, and Maillard, 2021). Unlike the class of linear MDPs, the Bilinear exponential family is very expressive and includes real examples like Tabular, Factored MDPs, and Linear Quadratic Regulators. After a thorough motivation of this

little-appreciated functional class, we unveil a novel observation assimilating the latter to an approximate linear representation. Consequently, we are able to provide a tractable algorithm for MDPs of this family in the episodic RL setting. We also provide a regret bound under mild regularity assumptions that exhibits an optimal dependence on H and K. Finally, the presented algorithm and analysis showcase certain improvements over the literature, *e.g.* clipping the value function is irrelevant.

1.3.4 Improved value estimator for deep policy gradients

In **Chapter 6**, probably our most practical work, we consider discounted infinite horizon MDPs with continuous state-action spaces. Our objective in this effort is to join the race for principled deep policy gradient algorithms with strong empirical performances. In fact, this research direction has been very prolific since the emergence of toy environments that mimic real life scenarios by encoding the laws of physics. Indeed, RL has been very successful in learning the dynamics of these tasks, ranging from playing various video games to controlling real robots.

We became interested in this problem after diverse observations by a handful of recent papers pinpointing significant discrepancies between the driving principles and insights that drive algorithms on and what is implemented in practice. For instance, the belief that neural network approximators allow for a smooth learning of value functions has been challenged showing that the deep policy framework fails to provide a decent fit. Motivated by these findings, we motivate then propose to adopt the residual error as a loss for critics in the actor-critic framework. We provide evidence that this methods fits the values better and demonstrate a consistent and logical performance boost on a variety of challenging tasks, including environments with sparse rewards signals. Finally, we provide empirical evidence that our method factually reduces the policy gradient variance, further demonstrating the soundness of our intuition, unlike previous methods that were shown to bias the estimators without variance reduction.

List of publications

The publications of this thesis were accepted for publication in international conferences with proceedings.

- Reda Ouhamma, Rémy Degenne, Pierre Gaillard, and Vianney Perchet. Online Sign Identification: Minimization of the Number of Errors in Thresholding Bandits. *Advances in Neural Information Processing Systems* 34 (2021), pp. 18577–18589 (used in **Chapter 3**)

- Reda Ouhamma, Odalric-Ambrym Maillard, and Vianney Perchet. Stochastic Online Linear Regression: the Forward Algorithm to Replace Ridge. *Advances in Neural Information Processing Systems* 34 (2021), pp. 24430–24441 (used in **Chapter 4, Chapter 5**)
- Reda Ouhamma, Debabrota Basu, and Odalric-Ambrym Maillard. Bilinear exponential family of mdps: Frequentist regret bound with tractable exploration and planning. *Association for the Advancement of Artificial Intelligence* (2023) (used in **Chapter 5**)
- Yannis Flet-Berliac, Reda Ouhamma, Odalric-Ambrym Maillard, and Philippe Preux. Learning value functions in deep policy gradients using residual variance. *International Conference on Learning Representations* (2021) (used in **Chapter 6**)

Chapter 2

Literature review

We motivate some key concepts discussed in this thesis and describe the current state-of-the-art and remaining challenges. Specifically, we discuss the literature of multi-armed bandits and Markov decision processes in the context of reinforcement learning, we provide brief overview of the origins and purposes of each formalism, and we lay out the major research directions relating to each setting.

Contents

2.1 Introduction

This chapter provides an overview of several key concepts in Reinforcement Learning (RL), including multi-armed bandits, Markov Decision Processes (MDPs), and different RL objectives. We will also explore some of the open questions in realistic RL and the challenges that arise when dealing with large-scale and complex problems.

Multi-armed bandits are a simple and fundamental problem in RL that has been widely studied and has many practical applications. MDPs are a more general framework for RL problems and are widely used in many applications. We will discuss the key concepts of MDPs, such as states, actions, rewards, and the value function. We will also discuss different variants of the MDP framework with their possible applications.

This chapter also examines different RL objectives, such as regret minimization and pure exploration, and various special manifestations of these objectives like best arm identification and thresholding bandits, which are important for many real-world applications. We also mention various concepts involved in solving this kind of RL problems, such as optimism-under-uncertainty, Thompson sampling and Track-and-stop

Finally, we will review some of the open questions in realistic RL, such as the challenges of dealing with large-scale and complex problems. We will inspect the importance of representations and structures that can simplify the problem and make it more tractable. We will also highlight the importance of addressing these sample and computational complexity challenges in order to make RL more widely applicable to real-world problems.

Overall, the chapter provides a broad yet non-exhaustive introduction to the field of RL, with a focus on the key concepts, variants, and open questions that are relevant to the core of this thesis.

Remark 2.1. *In this portion of the manuscript, we have made a conscious decision to be more verbose and steer away from discussing mathematical details. Our objective is not to provide a comprehensive, technical review of all relevant concepts, as there are already many comprehensive monographs available that serve that purpose. Instead, we aim to stimulate the reader's imagination and encourage discussion. By presenting the material in a more descriptive manner, we hope to generate a broader and more accessible understanding of the topic, and to encourage deeper exploration by the reader. Our focus is on creating a thought-provoking narrative that encourages the reader to engage with the material in a meaningful way.*

2.2 Stochastic multi-armed bandits

A stochastic multi-armed bandit problem is a set of distributions indexed by the available actions. A decision maker sequentially samples from the different distributions for a number of consecutive rounds. This first trails of this setting date back to the fifties where the seminal works (Robbins, 1952; Lai, Robbins, et al., 1985) introduced the stochastic multi-armed bandit problem. In a broader sense, these were among the pioneering works that paved the way for sequential statistics, *i.e.* where the sample size is not necessarily fixed or known in advance (Dodge and Romig, 1929; Wald, 1947). This novelty at the time was a significant step towards realistic and more efficient algorithms. Indeed, although the literature was far from being optimal, it still improved over known pre-defined time budgets. The latter was groundbreaking in the sense that it allowed statisticians a formalism where they could adjust their decisions continuously with the upstream of new information. This meant that the testing budgets were reduced and pointless samples were avoided.

In this modernization of research efforts on statistics, the departure was not from the fixed sample size exclusively, instead it also swept other outdated concepts from consideration, *e.g.* promoting multiple population statistics (Isbell, 1959; Bradt, Johnson, and Karlin, 1956; Bellman, 1956; Vogel, 1960). The latter was also motivated by the need for more applicability and better sample efficiency. Pragmatically, a statistician's goal is more often to compare multiple populations than just to confirm or infirm some property of a singular population. In the medicinal drug dosage problem for example, the doctor needs to determine the amount of medication to prescribe for their patient, and this can change over time with factors such as the resistance and reaction of the specific patient, the severity of their condition, the emergence of new drugs, and even to the comparison between several competing drugs with different properties. Overall, the fact that sequential decision making improves over classical statistics does not really require much motivation as it so clearly is an improvement in terms of applicability and prospects of gained efficiency.

Purpose and applications Nowadays, multi-armed bandits serve a variety of real world objectives. Indeed, there exist multiple scenarios where one needs to make decisions in a sequential manner in order to maximize rewards. For instance, multi-armed bandit can be used to model situations in which an agent must *allocate resources* (Lattimore, Crammer, and Szepesvári, 2015; Verma et al., 2019; Fontaine, Mannor, and Perchet, 2020), such as advertising budget, to different options in order to maximize the return on investment. Another considerable field of application is *clinical trials*, indeed, bandits can be model the selection of treatments or therapies in clinical trials (Berry, 1978; Villar, Bowden, and Wason, 2015; Aziz, Kaufmann, and Riviere, 2021), allowing researchers to learn which treatments are most effective over time. Moreover, bandits are also of great interest in the field of *network optimization*, they help in the allocation of resources in communication networks (Avner and Mannor, 2016; Li, Yang, et al., 2013; Cai et al., 2018; Gai, Krishnamachari, and Jain, 2010; Gai and Krishnamachari, 2011), allowing network operators to learn which strategies are most effective at optimizing network performance.

There still exist a wide variety of problems that are not mentioned here, *e.g. online advertising* to model the selection of ads displayed to the users and *personalization* to optimize their engagement or satisfaction. Overall, multi-armed bandits are a useful tool for modeling and solving decision-making problems in which the agent must learn which actions most probably lead to good outcomes over time.

Variants A consequence of the abundances of potential applications including bandits is that they induce a myriad of variations over the original MAB problem. Here we provide a non-exhaustive number of the many influential settings based on MABs.

Perhaps the most famous problem in this context is the *contextual MAB*. In contextual bandits (Abe and Long, 1999; Agarwal, Hsu, et al., 2014; Beygelzimer et al., 2011), the reward or utility of an action depends not only on the action itself, but also on the context in which the action is taken. For example, the utility of attending doctoral training may depend on the PhD student's prior education, interests, or current workload.

Combinatorial MABs is another compelling and popular setting. In combinatorial bandits (Anantharam, Varaiya, and Walrand, 1987; Caro and Gallien, 2007; Gai, Krishnamachari, and Jain, 2010; Chen, Wang, and Yuan, 2013), the agent must choose a combination of actions rather than a single action. This can be convenient for modeling scenarios where the learner must make multiple decisions or choices simultaneously, such as a PhD student allocating their time to multiple projects or papers.

Among the research directed at the previously mentioned continuous spaces challenge, *Linear bandits* were proposed as a structured variant of the (MAB) problem (Abe and Long, 1999; Auer, Cesa-Bianchi, and Fischer, 2002). In linear bandits, the reward of an action is modeled as a linear function of a set of -known- features or characteristics associated with the action. Several research efforts introduced optimism based algorithms for this setting (Abbasi-Yadkori, Pál, and Szepesvári, 2011; Dani, Hayes, and Kakade, 2008), along with near optimal regret bounds.

On the other hand, pure exploration objectives were also proposed with the multi-armed bandit setting. For instance, in *thresholding bandits* (Abernethy, Amin, and Zhu, 2016; Mukherjee et al., 2017; Tao et al., 2019), the final objective is to return arms whose reward exceeds a certain threshold value.

2.3 Markov Decision Processes

The Markov Decision Process (MDP) framework is a mathematical formalism for modeling problems of decision-making under uncertainty. There are many traces of this setting in the early fifties and sixties (Massé, 1946; Wald, 1947; Arrow, Harris, and Marschak, 1951; Arrow, Karlin, Scarf, et al., 1958; Dvoretzky, Kiefer, and Wolfowitz, 1952), this is not an exhaustive list as several objectives led to using similar frameworks at the time, *e.g.* inventory control, resource management, and sequential testing. Richard Bellman was the one who popularized MDPs in the early 1950s, indeed he introduced the framework rigorously and invented Dynamic programming in an effort to solve the optimization problems faced by engineers and economists (Bellman, 1966).

The motivation behind the MDP framework is to provide a flexible and generalizable tool for solving a wide range of optimization problems in which an agent must make decisions based on incomplete information about the environment. MDPs are particularly useful for

modeling problems in which the agent must take a sequence of actions over time, as they provide a way to represent the temporal structure of the problem.

The MDP framework has been widely applied in a variety of fields, including economics, operations research, engineering, and computer science. It has played a central role in the development of reinforcement learning (RL) (Sutton and Barto, 1998). MDPs have also been used to model a variety of real-world problems, such as inventory management, production planning, and robotic control (Powell, 2007).

Purpose The purpose of adopting MDPs in RL is to provide a formal framework for defining and solving RL problems, that is more general and realistic than the multi-armed bandit. An MDP consists of a set of states, actions, and a transition function that defines the probabilistic transitions between states as a result of taking actions. It also includes a reward function that defines the rewards received by the agent for taking certain actions in certain states. The goal of the RL agent is to learn a policy, which is a function that specifies the action to take in each state, that maximizes the expected cumulative reward over time.

MDPs provide a way to represent the temporal structure of RL problems, as they allow the agent to take sequential actions in order to achieve a goal. They also provide a way to incorporate uncertainty into the decision-making process, as the transitions and rewards are typically stochastic and the agent must learn to make decisions based on incomplete information.

MDPs have been applied to a wide range of RL problems, including control problems such as robotic manipulation and autonomous driving, as well as problems in economics and finance, such as portfolio management and auction design. They have also been used to model a variety of real-world problems, such as inventory management, production planning, and network routing.

Variants There are many variants of MDPs that have been proposed in literature. The major defining factor of these diverse alternatives appears in the way the environment or decision-making problem is structured.

In *Partially observable MDPs* (POMDPs), the agent only has partial observability of its state in the environment. This is suitable when modeling situations in which the agent has limited or noisy information about the environment (Thrun, 2002), such as in problems involving sensor uncertainty or hidden state variables. A different variant is the *Stochastic games* where there are multiple agents that can take actions simultaneously and influence the state of the environment. This is suitable for modeling situations where the actions of multiple agents are co-dependent (Kearns, Mansour, and Singh, 2013), such as in multi-player games or social dilemmas. Another variant of MDPs is *Decentralized MDPs*, in which the agent must make decisions based on the actions of other agents, who may be pursuing their own goals or

objectives. This fits scenarios where the agent must coordinate with other agents or respond to their actions (Amato et al., 2013), such as in distributed control or multi-agent systems.

On the other hand, we find a different kind of MDP variants where the contrast is in terms of algorithmic choices or structural assumptions. In other terms, these are variants of MDPs that differ in the way agents make decisions or the way policies are chosen and updated.

Reinforcement learning, the most pertinent variant for our manuscript, is where the agent learns a policy or strategy by interacting with the environment and receiving feedback in the form of rewards. This is useful for settings where the agent must learn from experience (Sutton and Barto, 1998), such as in problems involving exploration or uncertainty. Another algorithmic choice is *Dynamic programming*: in DP the agent uses a backward induction approach to solve the MDP by working from the final time step (horizon) backwards to the initial time step. Using the Bellman equations, the agent is able to compute the optimal value function and deduce the best policy to follow. This is suitable for interactions with a finite horizon or a known end time, and when the transition probabilities and rewards are known in advance. Please refer to the book (Bertsekas, 2000) for a complete overview. On a different notes, *Monte Carlo* methods were also proposed to solve MDPs. In MC algorithms, the agent estimate a policy's value function by sampling from the environment and averaging the rewards over multiple episodes. This is useful in problems with large state or action spaces, or when the transition probabilities or rewards are uncertain or hard to model.

2.4 Reinforcement learning objectives

The objectives in RL refer to the metrics used to evaluate the performance of an RL algorithm. These objectives typically reflect the ultimate goal of the agent, such as maximizing the cumulative reward, returning a policy within a certain distance of the optimal, or achieving a certain level of safety. Common performance objectives include the expected cumulative reward, the average reward per step, the value of the optimal policy, and the probability of reaching a goal state. Additionally, there are other objectives that are related to the stability and robustness of the algorithm, such as the variance of the achieved rewards and the convergence time. These performance objectives can be used to compare different RL algorithms and to guide the design and improvement of RL algorithms. In the real-world application of RL, it is crucial to identify the appropriate performance objectives and the according algorithm.

2.4.1 Regret minimization

Regret minimization is a key concept that is used in the field of reinforcement learning (RL) to evaluate the performance of RL algorithms. The idea is to compare the performance of a

strategy to that of an optimal decision-making algorithm, in terms of the cumulative rewards over time. Formally, regret is defined as the expected value of the difference between rewards of the optimal strategy and those obtained by the algorithm being evaluated.

There is a large body of literature on regret minimization in RL, with research focusing on various aspects of the problem such as designing algorithms with provably low regret, understanding the relationship between regret and other performance measures, and improving the sample complexity while keeping optimal bounds.

Most RL algorithms are based on the concept of optimism. *Optimism* in RL refers to the idea of taking actions that have a high potential for reward, *i.e.* the agent compares different actions optimistically by comparing their best possible outcomes (Agrawal, 1995; Kaelbling, 1993; Kaelbling, 1994). This can be accomplished by using optimistic initialization, which assigns a high initial value to the estimates of the value of the states or actions, or by using optimistic planning, which assumes that the unknown parts of the environment will be favorable. One example of an algorithm that uses optimism is UCRL (Upper Confidence RL), which is an optimistic algorithm that can be used for planning in uncertain environments.

Just as popular as the optimistic strategies is *Thompson sampling*. This is a strategy where the agent selects an action based on a probability weighting over the value of actions, this algorithm is older than Bandits or MDPs (Thompson, 1933). The said distribution is updated based on the observed rewards, and the action is again selected by sampling from the current distribution. This strategy can be useful when the agent is uncertain about the true value of the actions and they have a reasonable prior distribution for the rewards. Thompson sampling is the most popular Bayesian Multi-armed Bandit algorithm (Granmo, 2010; Scott, 2010; Chapelle and Li, 2011; May and Leslie, 2011; Agrawal and Goyal, 2012), it is also very commonly used for solving MDPs (Gopalan and Mannor, 2015; Agrawal and Jia, 2017).

Another popular concept in designing RL algorithms is *Explore-then-commit*. This is a strategy in which an agent explores different options or actions at the beginning of the decision-making process and then commits to the action that has the highest estimated value and for the rest of the process (Gittins and Jones, 1979). This strategy can be useful when the agent is uncertain about the true value of the actions or options but has a good idea of the relative value.

There exist other strategies in RL such as *Elimination*, in which an agent tries out different options or actions and eliminates the ones that have been found to be sub-optimal. This can be useful when the agent has a good idea of the relative value of the actions or options but is uncertain about their absolute value (Even-Dar et al., 2006). *Track-and-stop* is another common strategy, it is used when a reasonable estimation of the optimal allocation is available, through *e.g.* a tractable lower bound, and the agent's policy is then trying to match this estimated

allocation (Combes, Magureanu, and Proutiere, 2017; Garivier and Kaufmann, 2016; Degenne, Shao, and Koolen, 2020).

In addition to these, the research on regret minimization has also been extended to various specific scenarios such as adversarial setting and online learning with side observations. And it is worth noting that the mentioned strategies are not exhaustive nor mutually exclusive, indeed a number of recent efforts propose combinations of these algorithms.

2.4.2 Pure exploration

Pure exploration in RL refers to the problem of learning about the environment in order to identify the optimal policy or value function of the environment. In this problem, the agent may not know the reward function and it's objective is not to achieve a good reward but can rather be learning about the environment or returning a near optimal policy. Contrary to regret minimization, pure exploration encompasses a collection of disparate objectives, and with efforts focusing on various algorithmic. Also, in this setting and unlike the regret objective, the agent may be given a fixed confidence level instead of the time budget, in which case they need to decide when to stop the interaction and ensure an optimal response with the provided confidence level.

In the bandit literature, *Best arm identification* (BAI) is when the goal of the agent is to identify the arm with the highest expected reward in as few steps as possible. The best arm identification problem is a fundamental problem in the field of online learning and optimization. A number of algorithms have been proposed for BAI, some examples include UCB (Upper Confidence Bound) algorithm, Thompson Sampling and Bayesian optimization. Some key references for best arm identification problem are (Domingo, Gavalda, and Watanabe, 2002; Bubeck, Munos, and Stoltz, 2009; Audibert, Bubeck, and Munos, 2010). The extension of the latter to MDPs is called *Best policy identification* (BPI). The BPI problem refers to the problem of learning the best policy in an environment with unknown dynamics and/or unknown reward functions. The objective is to identify, as fast as possible, the policy that maximizes the expected cumulative reward. This is generally more challenging than BAI problem and various algorithms have been proposed to solve it. A key reference for best policy identification problem is the book (Fiechter, 1994; Zanette, Kochenderfer, and Brunskill, 2019; Al Marjani and Proutiere, 2021; Wagenmaker, Simchowitz, and Jamieson, 2022). A close variant of the latter is *Reward free* reinforcement learning. This is where there is no explicit feedback from the environment, in other words no rewards. RF methods rely on some kind of internal signal, such as the agent's curiosity or the uncertainty of its beliefs, to guide exploration. (Ménard et al., 2021) provides methods for the two settings of BPI and Reward-free learning and discuss the juxtaposed relationship of these concepts. Note that different kinds of pure exploration objectives exist in literature. For instance, *thresholding bandits* is a variation of the multi-armed bandit problem

where the goal is to identify the relative positions of the arms' expected rewards from a certain threshold (Steinwart, Hush, and Scovel, 2005; Locatelli, Gutzeit, and Carpentier, 2016; Chen, Lin, and Zhou, 2015; Tao et al., 2019). Thresholding bandits are relevant when the agent seeks to find all arms that have a pertinent level of reward instead of just finding the best one.

Regarding the prominent algorithms for this type of objectives, they are essentially similar to the popular strategies for regret minimization. For example, Optimism in the face of uncertainty is used for Best policy identification and reward free exploration (Ménard et al., 2021), Thompson Sampling was also proposed for a variety of exploration objectives (Russo, Van Roy, et al., 2017), and so on and so forth of previously introduced algorithmic concepts. A special kind of concepts is of greater interest in this setting, namely the *Information-theoretic measures*. These are used to quantify the amount of information gained by the agent when it explores different states or actions. One of the most widely used measures is the mutual information, which quantifies the degree of dependence between the state of the environment and the actions taken by the agent. Another commonly used measure is the conditional entropy, which quantifies the uncertainty of the agent's belief about the environment given its observations. Several works are based on this concept (Russo and Van Roy, 2014; Wagenmaker and Jamieson, 2022).

2.5 Realistic RL: Open questions and promising prospects

Reinforcement Learning research is a large body of literature and one of the largest subfield of machine learning. *Realistic RL* refers to the part of literature and techniques that can be applied to complex and realistic environments, such as robotics, autonomous vehicles, or at least to large-scale simulations mimicking the world.

The field of reinforcement learning (RL) has seen significant progress in both theoretical and practical research. The former focuses on understanding the fundamental properties of RL algorithms and developing provable guarantees for their performance. In contrast, practical RL focuses on designing implementable RL methods that handle real-world problems and environments, this often involve the use of deep learning techniques. However, there is often a disjunction between the theoretical and practical RL communities, with the former focusing on idealized problems and assumptions, and the latter focusing on the empirical performance of methods in real-world settings. This can make it challenging to find common ground, for this reason, there is a growing interest in developing theoretical frameworks that better match the settings and assumptions of realistic RL problems.

2.5.1 Representation

In reinforcement learning, the representation refers to how the agent's state, actions, and rewards are represented internally. Representation can be a defining factor of the problem's hardness and the subsequent algorithm complexity.

One of the most popular choices for representation in RL is the tabular one, where the agent's state is represented as a table with entries for each possible state, and the agent's action and value functions are also represented as tables. This method can be very sample efficient in small state spaces, but becomes infeasible in large or continuous state spaces, for several reasons such as the curse of dimensionality, the lack of differentiability, and the difficulty of exploring and learning in high-dimensional spaces. The curse of dimensionality refers to the exponential increase in the number of states as the dimensionality of the space increases. Another challenge is the lack of differentiability, which makes it difficult to use traditional optimization techniques such as gradient descent. Finally, exploring and learning in high-dimensional continuous spaces can be difficult, as the agent might need a very large number of interactions with the environment to learn a good policy.

For all previous reasons and more, continuous representations are indispensable in RL, and their choice is an important factor that can have a significant impact on the performance, sample efficiency, and tractability of the RL algorithm.

In this context, a popular direction in RL is the adoption of function approximation (Sutton, McAllester, et al., 1999; Melo, Meyn, and Ribeiro, 2008), where the agent's state, action, and value functions are represented by a parameterized function. This allows the agent to generalize from past experience to new states, and is supposed to be much more sample efficient in large or continuous state spaces. The choice of representation highly depends on the problem domain and the sensors available. Another important aspect to take in account is the generalization ability.

2.5.2 Structures in bandits

Structured multi-armed bandits deal with the problem of balancing exploration and exploitation in decision making when the action space is structured. Linear, Lipschitz, uni-modal and generalized linear are different possible assumptions that are commonly made about the underlying reward function in order to solve bandit problems efficiently.

The most popular bandits structure, and relevant to our manuscript in the linear reward assumption (Abbasi-Yadkori, Pál, and Szepesvári, 2011; Abeille and Lazaric, 2017). Indeed, linear bandits assume that the reward function is linear with respect to the parameters of the action. This assumption allows for the use of linear regression techniques (Vovk, 1997)

to estimate the parameters of the reward function and make more informed decisions. An extension of this structure is the Generalized linear multi-armed bandits (Filippi et al., 2010), where rewards are modeled by a Generalized Linear Model. A GLM is a flexible framework generalizes linear regression by assuming the response variable to be related to the explanatory variable through a link function on top of a linear model (Nelder and Wedderburn, 1972).

Note that there are many other relevant structures in literature, in we refer the interested reader to the following papers (Degenne, Shao, and Koolen, 2020; Combes, Magureanu, and Proutiere, 2017) for some examples of structures bandit instances as well as the state-of the art algorithms. Overall, it is true that structural assumptions in bandits have their flaws, but they are for most very realistic as they can model real-life scenarios, and they are useful in different settings because they entail more efficient solutions.

2.5.3 Structures in MDPs

Structured MDPs are an extension of the standard MDP framework where the state, action, or reward spaces have some additional structure. Examples of structured MDPs include Factored MDPs, which assume that the state space can be represented as a combination of smaller, independent components. This assumption allows for the use of techniques such as dynamic programming to solve the MDP more efficiently. MDPs with Graph feedback are another example, in which the state-action space is represented as a feedback graph, *i.e.* a directed graph where each node has an associated reward function, and each edge has an associated transition probability. (Dann, Mansour, et al., 2020) considers this setting and relates it to the bandits with side observations structure (Mannor and Shamir, 2011).

There exist certain techniques that allows researchers to handle generic structures, this often consists of tracking the optimal allocation given by lower bounds (Ok, Proutiere, and Tranos, 2018). However, the oracle allocation is often intractable and only currently available for infinite-horizon settings, *i.e.* obtaining tight -even intractable- lower bounds is still an active area of research in the finite horizon setting. Indeed, the stochasticity of transitions is a significant hurdle in this setting compared to bandits, especially when the state-action spaces are continuous. Currently, optimal finite time problem-dependent bounds for MDPs are still elusive, the current state of the art in this direction is the work (Tirinzoni, Al-Marjani, and Kaufmann, 2022) for the tabular episodic MDPs. This shows that the MDP literature has still got multiple open questions and the continuous state-action spaces area is still very challenging even to the purely theoretical researchers.

Linear MDPs is a recent class of representation where the dynamics and rewards are described by linear functions (Jin et al., 2020). This class of MDPs has gained a significant popularity because it entails a closed-form solutions using linear algebra, and they are computationally tractable and efficient. Also, using dynamic programming, the value can be represented

as a unique function which makes linear MDPs ideal for studying the convergence properties of the RL algorithms and analyzing their performance.

However, while it is true that the linear structure allows for efficient algorithms, the applicability of this class of MDPs is unsatisfactory. Indeed, the main limitation of linear MDPs is in assuming that the transition and value functions are linear. For instance, their isn't to this day a single example of finite dimensional linear transition function beyond tabular MDPs. This limited expressiveness doesn't seem to capture the complex relationships induced by stochastic transitions.

Overall, in MDPs the popular structures often suffer two possible limitations. The first consists of assumptions that are realistic and applicable, yet they entail intractable solutions and complicated algorithms that can't be implemented in practice. The second category includes structures for which very efficient solutions exist, unfortunately they are very unrealistic and therefore not suitable for modeling real world problems.

2.5.4 Compelling complexity measures

In this section, we will delve into the complexities of various classes of Markov Decision Processes (MDPs) and their significance in the field of Reinforcement Learning (RL). We will discuss three measures of complexity: Bellman rank, bilinear classes, and Eluder dimension, and how they are used to understand the properties of different classes of MDPs. Furthermore, we will also explore how these complexities are shaping the future of RL by providing insights into the fundamental limits of solving MDPs, studying the performance of different algorithms, and guiding the development of new methods and techniques. Understanding these complexities is crucial for making progress in the field of RL and for solving more challenging and realistic problems.

One way to measure the complexity of an MDP is by using the Eluder dimension (Russo and Van Roy, 2013), which measures the efficiency of predicting the value of actions not taken based on observed sample data. Eluder dimension has been used to study the complexity of various classes of MDPs, such as -possibly infinite- linear mixtures (Ayoub et al., 2020), and to provide general analyses for MDPs with small Eluder dimension (Wang, Salakhutdinov, and Yang, 2020; Ishfaq et al., 2021). It has been shown that the Eluder dimension provides optimal bounds for linear MDPs and generalized linear MDPs, it was also shown that this complexity is exactly equivalent to the information gain for reproducing kernel Hilbert spaces.

Another way to measure the complexity of an MDP is by using the Bellman rank (Jiang et al., 2017). Prior to defining the latter, let's define the average Bellman of a function f at step h when following policy π. This error is defined as the expected Bellman error of f at step h, when all previous actions were taken according to π. Now, the Bellman rank is basically the uniform

upper bound on the rank of all matrices formed by the Bellman errors of all possible functions f when following a greedy policy $\pi_{f'}$ of any possible function f'. In the case of tabular MDPs, it is simply the rank of the transition matrix matrix that represents the optimal value function. Bellman rank has been used to study the complexity of various classes of MDPs (Dong et al., 2020).

A third structural family is the Bilinear class of MDPs (Du, Kakade, Lee, et al., 2021), which connects the Bellman error, which measures the sub-optimality of a value function, to a sum of bilinear forms. Additionally, it allows for the use of data from a past value function to estimate a bilinear form for all value functions within the class. This essentially entails that data can be reused to evaluate multiple functions. The bilinear class subsumes block MDPs (Du, Krishnamurthy, et al., 2019), linear MDPs (Jin et al., 2020), linear quadratic regulators, factored MDPs (Kearns and Koller, 1999) and Bellman rank.

In conclusion, understanding the complexities of different classes of MDPs is crucial for making progress in the field of RL and for solving more challenging and realistic problems. The Bellman rank, Eluder dimension, and Bilinear classes are three ways to measure the complexity of an MDP and to study the properties of different classes of MDPs. These complexities are shaping the future of RL by providing insights into the fundamental limits of solving MDPs, studying the performance of different algorithms, and guiding the development of new methods and techniques.

2.5.5 Tractable RL

Reinforcement Learning is powerful for solving decision-making problems, and other than its sample complexity issues, it also comes with several computational challenges.

In RL, the policy and transition dynamics of the system should be modeled and the corresponding parameters must be estimated. In many problems, the transition dynamics are represented by a parametric probability density function, which can be complex and high-dimensional, making it difficult to estimate the parameters. This issue can be addressed by using various methods such as Gaussian Processes (Williams and Rasmussen, 2006), Maximum Likelihood Estimation (Levina and Bickel, 2004), and Variational Inference (Blei, Kucukelbir, and McAuliffe, 2017) to estimate the parameters of the transition dynamics.

Another manifestation of the computational efficiency hurdles is the optimization problem involving bilevel optimization problems where the outer problem is to optimize the policy, and the inner problem is to optimize the value function. In fact, a bilevel linear optimization problem is NP-hard in general (Jeroslow, 1985). Even in practice, such problems are known to be computationally challenging and may require specialized optimization techniques such as

gradient-based methods (Sinha, Khandait, and Mohanty, 2020) and meta-learning (Hospedales et al., 2020) to solve.

A third computational tractability hurdle is feature learning. For instance, the state space is often high-dimensional, making it difficult to learn the features of the state space that are relevant for the problem at hand. Practical approaches to deduce features for a task include using representation learning methods such as autoencoders (Vincent et al., 2008) and variational autoencoders (Kingma and Welling, 2013) that can be used to extract the relevant features from the high-dimensional state space. These methods learn a lower-dimensional representation of the state space that can be used to solve the RL task. On a more theoretical aspect, recent efforts in RL include causality structures (Bareinboim, Forney, and Pearl, 2015; Lu, Meisami, and Tewari, 2022; Lattimore, Crammer, and Szepesvári, 2015) that allow efficient and theoretically grounded methods for feature learning. Finally, we would like to mention research efforts for learning representations that are independent of RL, especially theoretically motivated ones like (Duvenaud et al., 2013; Malkomes, Schaff, and Garnett, 2016).

Finally, RL also comes with several less critical challenges. For example, the problem's horizon is often unknown and this can make the practitioner's job more challenging, it can however be usually solved by the doubling trick (Cesa-Bianchi, Freund, et al., 1997; Cesa-Bianchi and Lugosi, 2006) and we believe this should be more popular in literature. Also, the variance information even in the simple case of bandits is assumed to be known prior to learning, which is not realistic, and this can also be mitigated by specific adaptive regularization techniques (Durand, Maillard, and Pineau, 2018). Moreover, the bounds on variables are also important, *e.g.* for deriving confidence intervals, and they are usually assumed to be known. Overall, these limitations can lead to infeasible or unstable decisions, and using available information to the fullest extent is crucial for the success of RL algorithms. This involves effectively utilizing the information obtained from the environment and making assumptions that are applicable to a wide range of real life settings.

In conclusion, RL is a powerful method for solving sequential problems, but it also comes with several computational challenges. The complexity of parameter estimation, optimization problems, and the tractability of learning features are some of the main issues that need to be addressed in order to make progress in the field of RL. There are various methods in literature to address these challenges and to improve the performance of RL algorithms. However, despite these advances, there are still open problems and challenges that need to be addressed, and further research is needed to improve the scalability and efficiency of RL algorithms.

Chapter 3

Online sign identification for multi-armed bandits

In the fixed budget thresholding bandit problem, an algorithm sequentially allocates a budgeted number of samples to different arms (distributions). It then predicts whether the mean of each arm is larger or lower than a given threshold. We introduce a large family of algorithms (containing most existing relevant ones), inspired by the Frank-Wolfe algorithm, and provide a thorough yet generic analysis of their performance. This allowed us to construct new explicit algorithms, for a broad class of problems, whose losses are within a small constant factor of the non-adaptive oracle ones. Quite interestingly, we observed that adaptive methods empirically greatly out-perform non-adaptive oracles, an uncommon behavior in standard online learning settings, such as regret minimization. We explain this surprising phenomenon on an insightful toy problem. [1]

Contents

[1] This chapter is based on a collaboration with Rémy Degenne, Pierre Gaillard, and Vianney Perchet (Ouhamma, Degenne, et al., 2021) It was accepted for publication as a spotlight at the *32nd conference on advances in Neural Information Processing Systems* (*NeurIPS*).

3.1 Introduction

In a stochastic multi-armed bandit problem, a decision maker sequentially samples from different distributions in order to optimize a loss that depends on the unknown parameters of those distributions. As a consequence, a trade-off arises between gathering more samples from any possible distribution (to enhance the estimation of relevant parameters) and optimizing the allocation to minimize the final loss. We can distinguish two main categories of losses, focusing on "exploitation" vs "exploration". The former directly depends on the whole allocation of samples and the typical example is regret minimization (we refer to the recent monographs (Lattimore and Szepesvári, 2020; Bubeck, Cesa-Bianchi, et al., 2012; Slivkins et al., 2019) that cover this setting almost exhaustively). The later is a bit different; after the budget of samples is exhausted, the algorithms must answer one or several "questions" (on the different distribution) and its loss is related to the number of mistakes made; the typical application being best-arm identification and variants (Audibert, Bubeck, and Munos, 2010; Kaufmann, Cappé, and Garivier, 2016).

We investigate a class of pure exploration problems, called "thresholding bandit" (Locatelli, Gutzeit, and Carpentier, 2016; Tao et al., 2019). The key property of this class is that a question is asked about each distribution, and the probability of making a mistake decreases with the total information gathered on that distribution solely. The typical question the algorithm must answer is "is the mean of the distribution above or below some threshold?" (say, 0, for simplicity); giving the wrong answer can either incur a unit cost - independently from the distribution -, or a data-dependent cost (say, the distance to the threshold that represents the "risk" of that distribution). A typical application of thresholding bandits is crowdsourcing (Chen, Lin, and Zhou, 2015) where the objective is to distinguish workers with positive (vs. negative) efficiency; another one is bandit binary classification (Jain and Jamieson, 2019).

Some care must be taken when designing a performance criterion for a thresholding bandit problem, since any non-stupid algorithm will eventually answer all questions correctly (hence have a 0 loss) if it has enough samples. Furthermore, if distributions are sub-Gaussian (a rather mild assumption that we are going to make), the probability of making a single mistake decreases exponentially fast with the number of samples. As a consequence, the focus must be on controlling the exponential decay constant. We illustrate that issue on the unit cost problem described as follows. There are K different σ-sub-Gaussian distributions; the mean of distribution k is denoted by μ_k and the (variance-normalized) gap of distribution k to the threshold 0 is denoted by $\Delta_k := |\mu_k|/\sqrt{2\sigma^2}$. The algorithm has a budget of T samples to (sequentially) allocate to those distributions and, based on the $N_{k,T}$ samples of distribution k, it must decide the sign of μ_k; any mistake has a cost of one. We denote by $E_k \in \{0, 1\}$ an indicator of a wrong sign prediction of μ_k after exhausting the budget of T samples. The loss is

then $L_T^1 := \sum_k E_k$. It is not difficult to see that the expected number of mistakes could be of order $\sum_{k=1}^K \exp(-N_{k,T}\Delta_k^2)$.

In particular, sampling evenly across distributions (*i.e.* choosing $N_{k,T} = T/K$) gives an expected loss $\mathbb{E}[L_T^1] \approx \sum_k \exp(-\frac{T}{K}\Delta_k^2)$, which has an exponential decay in T. However, this uniform allocation is far from being optimal in term of the exponential decay constant. Computing an (approximate) optimal fixed allocation in hindsight is not difficult: just optimize the upper-bound of $\mathbb{E}[L_T^1]$. Since even the uniform allocation has a loss decaying exponentially, the performance of an algorithm should be measured not with respect to $\mathbb{E}[L_T^1]$ (see (Kaufmann, Cappé, and Garivier, 2016)) but rather in terms of $-\log(\mathbb{E}[L_T^1])/T$. The oracle that uses knowledge of the gaps Δ_k to optimize its fixed allocation verifies

$$\limsup_{T\to\infty} \frac{1}{T}\log(\mathbb{E}[L_T^1]) \leq -\frac{1}{\sum_k 1/\Delta_k^2} .$$

This unit cost framework has been investigated recently (Tao et al., 2019) with a simple yet effective algorithm called LSA (Logarithmic-Sample Algorithm) designed exclusively for this problem; it samples the distribution with the smallest current index defined as $\alpha N_{k,t}\hat{\Delta}_{k,t}^2 + \log N_{k,t}$, where $\hat{\Delta}_{k,t}$ is the empirical estimate of Δ_k and α is some parameter to be chosen. LSA is "optimal up to a constant", but the constant is unfortunately in the exponential decay, as it was proved that[2]

$$\limsup_{T\to\infty} \frac{1}{T}\log(\mathbb{E}[L_T^1]) \leq -\frac{1}{16020}\frac{1}{\sum_k 1/\Delta_k^2} \quad \text{for LSA.}$$

As we shall see, thanks to our new refined and general proof methodology, we can improve this result drastically (that implies choosing a totally different parameter $\alpha = 1$ instead of $1/10$ as suggested originally) without modifying the algorithm.

3.1.1 Contributions

We investigate the thresholding bandit problem with a weighted number of errors loss. Our contributions are twofold: 1) a generic method to design algorithms, with a generic proof, showing good performance on the weighted number of errors loss. 2) new lower-bounds and counter-intuitive results for the unit cost problem.

A generic algorithm with performance guarantees We propose a Frank-Wolfe inspired method to design bandit algorithms. We develop a proof technique to obtain loss bounds for the type of algorithms that our method produces, which we apply to the thresholding bandit

[2]See Remark 1 (Tao et al., 2019). This bound implies that LSA - with the specified choice of $\alpha = 0.1$ needs 16000 times more samples than the oracle to achieve the same performances.

with losses

$$L_T = \sum_{k=1}^{K} a_k E_k \quad \text{or} \quad L_T^{\Delta} = \sum_{k=1}^{K} \Delta_k E_k \,, \tag{3.1}$$

where $(a_k)_{k\in[K]}$ are known costs. The class of algorithms we analyze includes both LSA and APT (Anytime Parameter-free Thresholding) (Tao et al., 2019; Locatelli, Gutzeit, and Carpentier, 2016). We obtain precise non-asymptotic loss bounds for $\mathbb{E}[L_T]$; for instance, we improve the original bound of LSA by a factor 4005 (and APT by a factor 8). More importantly, we get a new algorithm whose expected error for the unit cost problem is within a factor 4 of the oracle. We emphasize again than those "constant" factors are in the exponential (and are not mere multiplicative constants).

Interestingly, this class of algorithms are *not* driven either by the "optimism under uncertainty" principle, a standard technique in multi-armed bandit (Auer, Cesa-Bianchi, and Fischer, 2002) nor "Explore-then-commit / Successive Elimination" (Perchet et al., 2016; Even-Dar et al., 2006).

New insights on the thresholding bandit problem First, the optimal allocation provided by the oracle of (Tao et al., 2019) in the unit cost problem has a M-shape (see Figure 3.1) because of two concurrent phenomena. On the one hand, the arms close to the threshold should not be pulled too much because their sign is difficult (if not impossible) to identify and it is a waste of budget. On the other hand, the signs of the arms far from the threshold are quickly well estimated and therefore should not be chosen too often either. The middle arms are the ones that need to be pulled the most frequently. As T gets larger, more and more budget is allocated to difficult arms. In Section 3.2.3, we provide a lower-bound that shows that this M shape is actually impossible to achieve for a sequential algorithm. Typically, the hollow inside of the M shape corresponds to arms whose sign cannot be well-estimated. In particular, it is not possible to distinguish arms that are very close to the threshold from the arms that are at the top of the M and should be pulled the most frequently according to the oracle.

Our second insight is corroborated by numerical simulations in Section 3.7. We show empirically that our algorithms not only match but also surpass the optimal non-adaptive sampling of the oracle. We conjecture that our algorithms take advantage of the chance due to noise that can move its estimate of the arm away from the threshold. In particular, when all the gaps Δ_k are equal, the non-adaptive optimal allocation should be uniform, which is significantly outperformed by adaptive algorithms. This suggests that adaptivity is crucial for this problem and may inspire future research directions to the multi-armed bandit community in order to prove theoretical guarantees for such phenomena.

3.1.2 Additional related work

Zero-one loss Most of the literature on thresholding bandits (Locatelli, Gutzeit, and Carpentier, 2016; Mukherjee et al., 2017; Cheshire, Menard, and Carpentier, 2020) aims at minimizing the probability of making any sign error, i.e., minimizing the loss

$$L_T^* = \mathbb{I}\{\exists k \in [K],\ E_k = 1\} = \max_k E_k. \tag{3.2}$$

We already mentioned the algorithm APT (Locatelli, Gutzeit, and Carpentier, 2016), that gets an exponential decay of that loss (variants include variance estimation (Zhong, Huang, and Liu, 2017) and/or delayed feedbacks). Other algorithms exist, but based on the optimism principle (Katz-Samuels and Scott, 2018; Mukherjee et al., 2017). Unfortunately they suffer from a degraded exponential decay constant (by a factor bigger than 1000).

Another part of the literature focuses on the fixed confidence framework, where the objective is to answer some questions with some fixed probability of mistake (and obviously with a minimal sample budget). For instance, an objective could be to return any arm above some threshold as soon as possible (Kano et al., 2019; Degenne and Koolen, 2019), or the one closest to the threshold (Garivier, Ménard, et al., 2017), or just identifying that one arm is above that threshold (Kaufmann, Koolen, and Garivier, 2018), or even to control false discovery rates and variants (Jamieson and Jain, 2018; Jain and Jamieson, 2019).

Global loss, dynamic allocation and outliers detection The loss considered in thresholding bandits can be seen as a variant of a "global loss" (i.e., essentially non-linear) that has been extensively studied in the bandit literature (Agarwal, Foster, et al., 2011; Agrawal and Devanur, 2014; Mannor, Perchet, and Stoltz, 2014). However, the major difference is, again, that the optimal allocation is time dependent and that the loss converges exponentially fast to zero (no matter the algorithm). Similarly, Frank Wolfe algorithms have been introduced in this setting (Berthet and Perchet, 2017; Fontaine, Berthet, and Perchet, 2019); even though our algorithms share some similarities, they are intrinsically different for the same reasons.

Similarly, the problem investigated could be seen as a special case of bandit resource allocations (Koopman, 1953; Chen, Lin, and Zhou, 2015; Salehi et al., 2016; Devanur et al., 2019; Fontaine, Mannor, and Perchet, 2020) but where the loss is always decreasing with respect to the budget allocated per resource (hence again leading to a zero loss exponentially fast).

Finally the global objective of thresholding bandits is to obtain a synthetic view of how the means of distributions are spread on the real line (which ones are above/below some threshold). In that aspect, this problem sheds some similarities with outlier detection in multi-armed bandits (Katariya, Tripathy, and Nowak, 2019; Zhuang, Wang, and Wang, 2017; Zhu, Katariya, and Nowak, 2020).

3.2 Preliminaries

We describe here the weighted number of errors setting, in which an error on arm k has a known cost $a_k > 0$. The sum-of-gaps setting will be briefly investigated in Section 3.5.3. The environment is composed of $K > 1$ arms and an algorithm sequentially pulls them. After pulling arm $k \in [K]$, it observes a sample from a distribution ν_k with mean μ_k, and that sample is independent of past observations. The distribution ν_k is supposed σ-sub-Gaussian, that is

$$\forall \lambda \in \mathbb{R} : \mathbb{E}_{X \sim \nu_k} \left[\exp(\lambda(X - \mu_k)) \right] \leq \exp(\sigma^2 \lambda^2 / 2) \, .$$

The total number of rounds (and samples) T is known in advance and called the horizon. After pulling T arms, the task of the algorithm is to classify the arms depending on whether $\mu_k > \theta$ or not, where θ is a known threshold that we conveniently set to 0 (although it could be any other value, even different from arm to arm, without significant change to the analysis). Let $s_k \in \{-1, 1\}$ be the sign of $\mu_k - \theta$, equal to 1 iff $\mu_k - \theta > 0$. The algorithm returns for all arms an estimated sign $\hat{s}_k \in \{-1, 1\}$. The objective is to minimize the expected weighted number of missclassified arms, where a mistake on arm k has a known cost $a_k > 0$,

$$L_T = \sum_{k=1}^{K} a_k \mathbb{I}\{\hat{s}_k \neq s_k\} = \sum_{k=1}^{K} a_k E_k \, . \tag{3.3}$$

Note that the linear form of the loss is quite general: since $E_k \in \{0, 1\}$, any separable loss $\sum_k f_k(E_k)$ is the sum of a constant and $\sum_k a_k E_k$ for some costs a_k.

We conclude this description of the problem with notations used in the design of algorithms. Let $N_{k,t}$ and $\hat{\mu}_{k,t} = \frac{1}{N_{k,t}} \sum_{s=1}^{t} \mathbb{I}\{i_t = k\} X_t$ be the number of times the learner has pulled arm k up to round t (included) and the subsequent empirical mean of arm k repectively. Define further $\hat{\Delta}_{k,t} = |\hat{\mu}_{k,t} - \theta| / \sqrt{2\sigma^2}$ and $\Delta_k = |\mu_k - \theta| / \sqrt{2\sigma^2}$, respectively the empirical and the true (variance-normalized) gap of arm k to the threshold after t rounds.

3.2.1 Lower bound for the expected number of mistakes

Following the proof of (Tao et al., 2019) in a slightly more generic fashion (using exponential families with one parameter instead of Bernoulli distribution), we obtain a lower bound on the performance of any algorithm from which we get Theorem 3.1.

Theorem 3.1. (*Similar to Theorem 20 in (Tao et al., 2019)*) *Let $(\Delta_1, \dots, \Delta_K)$ be a sequence of gaps. Then for any algorithm and time horizon $T \geq K$, there exists an instance in which all arms*

$k \in [K]$ have Gaussian distributions with variance σ^2 and mean in $\{\Delta_k, -\Delta_k\}$ such that

$$\mathbb{E}[L_T] \geq \frac{1}{4} \min_{\sum_k N_k = T} \sum_{k=1}^{K} a_k e^{-4N_k \Delta_k^2} .$$

Proof. First, denote the expected loss on a bandit problem μ: $\mathbb{E}[L_T(\mu)] = \sum_{k=1}^{K} a_k \mathbb{P}_\mu\{\hat{s}_k \neq s_k\}$.

For each arm $k \in [K]$, define two values $\mu_k, \tilde{\mu}_k \in \mathbb{R}$, with $\mu_k < \theta < \tilde{\mu}_k$. Let $\mu = (\mu_k)_{k\in[K]}$. For some fixed one-parameter exponential family, we denote by $\mathrm{KL}(a, b)$ the Kullback-Leibler divergence between distributions with mean a and b.

Given a vector $\lambda \in \mathbb{R}^K$ with $\lambda_k \in \{\mu_k, \tilde{\mu}_k\}$ for all $k \in [K]$ and $S \subseteq [K]$, let λ_S be such that $\lambda_{k,S} \in \{\mu_k, \tilde{\mu}_k\}$ and $\lambda_{k,S} \neq \lambda_k$ for $k \in S$ and $\lambda_{j,S} = \mu_j$ for $j \notin S$.

For $S \in \mathcal{P}([K])$, let $S \pm i$ be equal to $S \cup \{i\}$ if $i \notin S$ and to $S \setminus \{i\}$ otherwise. Also, we denote by $(s_k(\lambda))$ be the signs of (λ_k). Then the following holds

$$\begin{aligned}
\sup_{S\in\mathcal{P}([K])} \mathbb{E}[L_T(\mu_S)] &\geq \frac{1}{2^K} \sum_{S\in\mathcal{P}([K])} \mathbb{E}[L_T(\mu_S)] \\
&= \frac{1}{2^K} \sum_{S\in\mathcal{P}([K])} \sum_{k=1}^{K} a_k \mathbb{P}_{\mu_S}\{\hat{s}_k \neq s_k(\mu_S)\} \\
&= \frac{1}{2^{K+1}} \sum_{S\in\mathcal{P}([K])} \sum_{k=1}^{K} a_k \mathbb{P}_{\mu_S}\{\hat{s}_k \neq s_k(\mu_S)\} + a_k \mathbb{P}_{\mu_{S\pm k}}\{\hat{s}_k \neq s_k(\mu_{S\pm k})\} \\
&= \frac{1}{2^{K+1}} \sum_{S\in\mathcal{P}([K])} \sum_{k=1}^{K} a_k \mathbb{P}_{\mu_S}\{\hat{s}_k \neq s_k(\mu_S)\} + a_k \mathbb{P}_{\mu_{S\pm k}}\{\hat{s}_k = s_k(\mu_S)\}.
\end{aligned}$$

For each arm k, we can bound the sum of the two probabilities from below. Let $\mathcal{E}_{k,S} = \{\hat{s}_k \neq s_k(\mu_S)\}$.

$$\mathbb{P}_{\mu_S}(\mathcal{E}_{k,S}) + \mathbb{P}_{\mu_{S\pm k}}(\overline{\mathcal{E}_{k,S}}) \geq \frac{1}{2} \exp\left(-\mathbb{E}_{\mu_S}[N_{k,T}] \,\mathrm{KL}(\mu_{k,S}, \mu_{k,S\pm k})\right),$$

so that, when plugged back in the previous equation, we get

$$\begin{aligned}
\sup_{S\in\mathcal{P}([K])} \mathbb{E}[L_T(\mu_S)] &\geq \frac{1}{2^{K+1}} \sum_{S} \sum_{k=1}^{K} \frac{1}{2} a_k \exp\left(-\mathbb{E}_{\mu_S}[N_{k,T}] \,\mathrm{KL}(\mu_{k,S}, \mu_{k,S\pm k})\right) \\
&\geq \frac{1}{4} \frac{1}{2^K} \sum_{S} \min_{N:\sum_k N_k = T} \sum_{k=1}^{K} a_k \exp\left(-N_k \,\mathrm{KL}(\mu_{k,S}, \mu_{k,S\pm k})\right)
\end{aligned}$$

$$\geq \frac{1}{4} \min_{N:\sum_k N_k = T} \sum_{k=1}^{K} a_k \exp\left(-N_k \max\{\mathrm{KL}(\mu_k, \tilde{\mu}_k), \mathrm{KL}(\tilde{\mu}_k, \mu_k)\}\right)$$

We finish the proof by recalling that for Gaussians with variance σ^2, $\mathrm{KL}(a,b) = \frac{(a-b)^2}{2\sigma^2}$. □

As can be seen from the proof above, the result holds more generally for one-parameter exponential-families where the gaps on the r.h.s are replaced with the Kullback-Leibler divergence.

3.2.2 Non-adaptive oracle

We now derive an optimal but unrealistic oracle, which requires prior knowledge of the gaps as input. Consider the algorithm that pulls each arm $N_{k,T}$ times, a number fixed in advance, then returns the sign of the empirical mean $\hat{\mu}_{k,T}$. Using Hoeffding's inequality, the expected loss verifies:

$$\mathbb{E}[L_T] = \sum_{k=1}^{K} a_k \mathbb{P}\left((\hat{\mu}_{k,T} - \theta)(\mu_k - \theta) < 0\right) \leq \sum_{k=1}^{K} a_k e^{-N_{k,T}\Delta_k^2} \tag{3.4}$$

We define the *non-adaptive oracle* as the allocation N_T which minimizes that upper bound. Its error probability has the same form as the lower bound of Theorem 3.1, but has a different constant in the exponential (1 instead of 4). We can solve that minimization problem and make the error bound more explicit.

Lemma 3.2. *Suppose that the arms are ordered such that $a_1\Delta_1^2 \leq \ldots \leq a_K\Delta_K^2$. There is a set $S = \{k_0, k_0+1, \ldots, K\}$ and a constant C_S such that the oracle non-adaptive algorithm has $N_{k,T} = 0$ for $k \notin S$ and $N_{k,T} = \left(C_S + \log(a_k\Delta_k^2)\right)/\Delta_k^2$ for $k \in S$ (see the proof below for details). The expected loss of that non-adaptive oracle is*

$$\mathbb{E}[L_T] \leq \sum_{k \notin S} a_k + \sum_{k \in S} a_k \exp\left(-\frac{T + \sum_{j\in S} \frac{1}{\Delta_j^2} \log\left(\frac{a_k\Delta_k^2}{a_j\Delta_j^2}\right)}{\sum_{j\in S} \frac{1}{\Delta_j^2}}\right). \tag{3.5}$$

Proof. The objective of the non-adaptive oracle section is to find an explicit solution of

$$\min_{\sum_k N_k = T} \sum_k a_k e^{-N_k\Delta_k^2}.$$

Introducing the Lagrange multiplier $\gamma \in \mathbb{R}$, it is straightforward that the solution is such that all N_k which are nonzero verify $\frac{\partial}{\partial N_k}(\sum_j a_j e^{-N_j\Delta_j^2}) = \gamma$. Then there exists a set S and a

constant $\gamma_S(T) > 0$ for which $k \notin S \implies N_k = 0$ and for $k \in S$, $N_k \neq 0$ and

$$a_k \Delta_k^2 e^{-N_k \Delta_k^2} = \gamma_S(T) \, .$$

That is, $N_k = \frac{1}{\Delta_k^2}(\gamma_S(T) + \log(a_k \Delta_k^2)) \, .$

We remark that $k \in S$ iff $\frac{1}{\Delta_k^2}(\gamma_S(T) + \log(a_k \Delta_k^2)) > 0$, which then implies that if $a_1 \Delta_1^2 \leq \ldots \leq a_K \Delta_K^2$, then $S = \{k_0, k_0 + 1, \ldots, K\}$ for some $k_0 \in [K]$.

Using the condition $\sum_k N_k = T$ to determine $\gamma_S(T)$, we get

$$\sum_{k=k_0}^{K} \frac{1}{\Delta_k^2}(\gamma_S(T) + \log(a_k \Delta_k^2)) = T \quad \implies \quad \gamma_S(T) = \frac{T + \sum_{k=k_0}^{K} \frac{1}{\Delta_k^2} \log \frac{1}{a_k \Delta_k^2}}{\sum_{k=k_0}^{K} \frac{1}{\Delta_k^2}} \, .$$

Finally, we can characterize k_0. Notice that $k \in S$ iff $\frac{1}{\Delta_k^2}(\gamma_S(T) + \log(a_k \Delta_k^2)) > 0$, i.e. iff

$$\frac{1}{\Delta_k^2}\left(\frac{T + \sum_{j=k_0}^{K} \frac{1}{\Delta_j^2} \log \frac{1}{a_j \Delta_j^2}}{\sum_{j=k_0}^{K} \frac{1}{\Delta_j^2}} + \log(a_k \Delta_k^2) \right) > 0 \quad \Leftrightarrow \quad T > \sum_{j=k_0}^{K} \frac{1}{\Delta_j^2} \log \frac{a_j \Delta_j^2}{a_k \Delta_k^2} \, .$$

Finally, let $H_k = \sum_{j=k+1}^{K} \frac{1}{\Delta_j^2} \log \frac{a_j \Delta_j^2}{a_k \Delta_k^2}$, with $H_0 = +\infty$ and $H_K = 0$. Then k_0 is the unique element of $[K]$ such that $H_{k_0} < T \leq H_{k_0 - 1}$. □

Note that the oracle is not pulling arms $1, \ldots, k_0 - 1$. These are the arms which are too close to the threshold (in a distance weighted by a_k) and thus too hard to classify to be worth trying. Giving up on those arms is not something that a non-oracle algorithm can do.

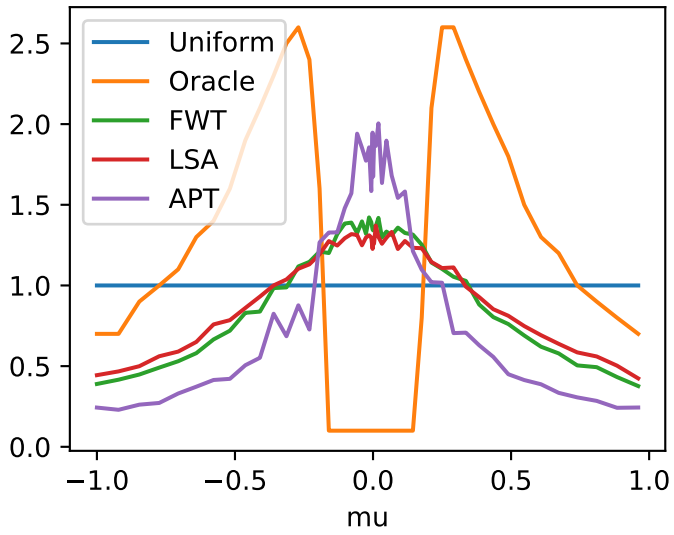


Figure 3.1 – Optimal and empirical sampling distributions with respect to μ.

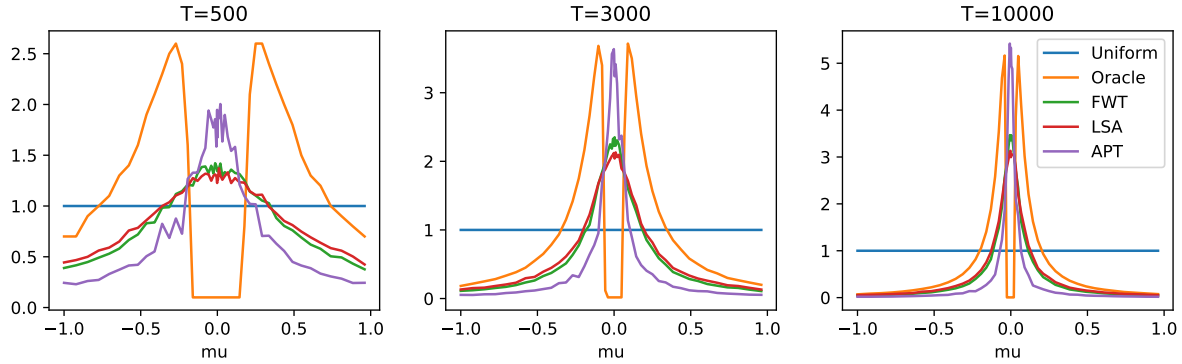


Figure 3.2 – Optimal sampling distribution and empirical sampling distribution with respect to μ when $\mu_k = \frac{(-1)^k k^2}{K^2}$ for $K = 50$ arms.

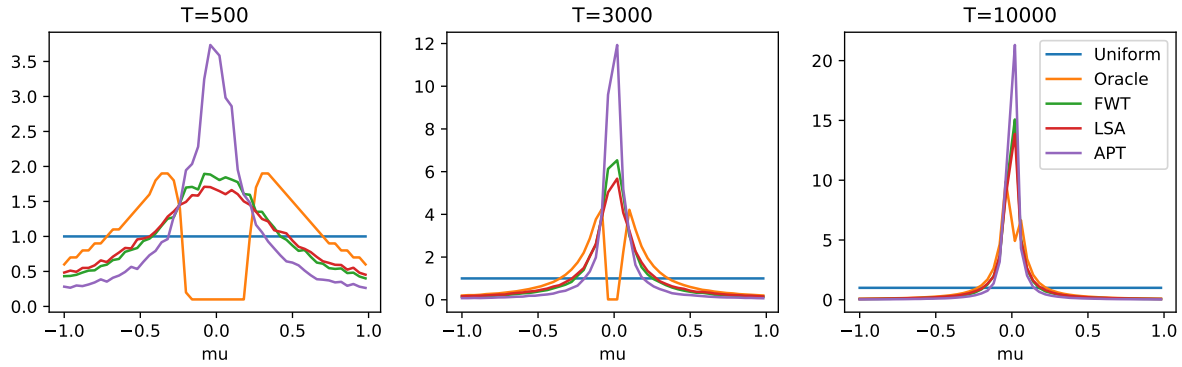


Figure 3.3 – Optimal sampling distribution and empirical sampling distribution with respect to μ when $\mu_k = \frac{(-1)^k k}{K}$ for $K = 50$ arms.

Figure 3.1 illustrates on an example ($\mu_k = (-1)^k (k/K)^2, k = 1, \dots, 50$, and $T = 500$) the shape of the optimal allocation (arms near the threshold should not be drawn) as well as the empirical sampling distributions of several algorithms that pull all arms. Next, we illustrate how this optimal allocation evolves with the horizon T.

Figures 3.2 and 3.3 represent the optimal non-adaptive sampling distribution if the means were known and the empirical sampling distribution of the algorithms for different numbers of iterations. As we can see, for the initial phase, the arms that are closest to the threshold should ideally not be drawn. Yet, as we will see in the next section, this is not possible for sequential algorithms. All arms must be sampled. We can see that this is indeed the case for all algorithms: the closer the arms are to the threshold, the more likely they are to be sampled.

3.2.3 A good algorithm must pull all arms

We provide a new lower bound for the thresholding bandit with unit-cost problem, to support the claim that it is not possible to avoid pulling the arms which are close to the threshold. Consider the following 4 Gaussian bandit models (with variances 1) with means

$$\mu_{+\varepsilon} = (\varepsilon, \dots, \varepsilon, \mu_{K_0+1}, \dots, \mu_K)\,, \qquad \mu'_{+} = (\mu_{K_0+1}, \dots, \mu_{K_0+1}, \mu_{K_0+1}, \dots, \mu_K)\,,$$
$$\mu_{-\varepsilon} = (-\varepsilon, \dots, -\varepsilon, \mu_{K_0+1}, \dots, \mu_K)\,, \qquad \mu'_{-} = (-\mu_{K_0+1}, \dots, -\mu_{K_0+1}, \mu_{K_0+1}, \dots, \mu_K)\,.$$

where $0 < \varepsilon < \mu_{K_0+1} \leq \ldots \leq \mu_K$, the value μ_{K_0+1} is large enough for the oracle to pull all arms on μ'_+ and $\varepsilon \leq \sqrt{\log(2)/(2T)}$.

Lemma 3.3. *If* $\mathbb{E}_{\tilde{\mu}}[L_T] \leq c_1 \min_{\sum_k N_k = T} \sum_k e^{-c_0 N_k \Delta_k^2}$ *for constants* c_0, c_1 *on* $\tilde{\mu} \in \{\mu'_+, \mu'_-\}$*, then*

$$\max_{\mu \in \{\mu_{+\varepsilon}, \mu_{-\varepsilon}\}} \mathbb{E}_\mu \left[\sum_{k=1}^{K_0} N_{k,T} \right] \geq \frac{1}{2(\mu_{K_0+1} - \varepsilon)^2} \left(c_0 \frac{T + H^{\log}}{H} + \log \frac{K_0}{32 c_1 H} \right)\,.$$

where $H = \frac{K_0}{\Delta^2_{K_0+1}} + \sum_{k=K_0+1}^{K} \frac{1}{\Delta_k^2}$ *and* $H^{\log} = \frac{K_0}{\Delta^2_{K_0+1}} \log \frac{1}{\Delta^2_{K_0+1}} + \sum_{k=K_0+1}^{K} \frac{1}{\Delta_k^2} \log \frac{1}{\Delta_k^2}$.

The proof idea is that if an algorithm has an expected loss close to the loss of the non-adaptive oracle, then it must pull linearly the arms which are close to the threshold.

Proof. We will prove that the number of pulls of arms $1, \dots, K_0$ cannot be too small. Formally, let $n_\varepsilon = \sum_{k=1}^{K_0} \mathbb{E}_{\mu_{+\varepsilon}} N_{k,T}$ be the expected number of pulls under $\mu_{+\varepsilon}$ of the arms with mean ε. We aim at showing that that number cannot be zero. We first prove that

$$\mathbb{P}_{\mu_{+\varepsilon}}(L_T > K_0/2) \geq \frac{1}{4}. \tag{3.6}$$

This follows from the basic inequalities,

$$\mathbb{P}_{\mu_{+\varepsilon}}(L_T(\mu_{+\varepsilon}) > K_0/2) + \mathbb{P}_{\mu_{-\varepsilon}}(L_T(\mu_{+\varepsilon}) \leq K_0/2) \geq \frac{1}{2} e^{-n_\varepsilon \operatorname{KL}(\varepsilon, -\varepsilon)} \geq \frac{1}{2} e^{-2\varepsilon^2 T}$$

In particular, for $\varepsilon \leq \sqrt{\frac{\log 2}{2T}}$, $\max\{\mathbb{P}_{\mu_{+\varepsilon}}(L_T(\mu_{+\varepsilon}) > K_0/2), \mathbb{P}_{\mu_{-\varepsilon}}(L_T(\mu_{-\varepsilon}) > K_0/2)\} \geq \frac{1}{4}$, and either Inequality 3.6 either holds for $\mu_{+\varepsilon}$, or we just need to switch the role of ε and $-\varepsilon$ in this proof. Suppose now that Inequality 3.6 holds for $\mu_{+\varepsilon}$.

The Kullback-Leibler divergence between $\mu_{+\varepsilon}$ and μ'_+ is

$$\begin{aligned}\sum_k \mathbb{E}_\mu[N_{k,t}]\,\mathrm{KL}(\mu_{k,+\varepsilon},\mu'_{k,+}) &= \sum_{k=1}^{K_0}\mathbb{E}_\mu[N_{k,t}]\,\mathrm{KL}(\varepsilon,\mu_{k_0+1})\\ &\geq \mathrm{kl}(\mathbb{P}_{\mu_{+\varepsilon}}(L_T>K_0/2),\mathbb{P}_{\mu'_+}(L_T>K_0/2))\\ &\geq \mathrm{kl}(\frac{1}{4},\frac{2}{K_0}\mathbb{E}_{\mu'_+}[L_T]) \geq \frac{1}{4}\log\frac{K_0}{2\mathbb{E}_{\mu'_+}[L_T]} - \log 2\,,\end{aligned}$$

We have proved that $n_\varepsilon \geq \frac{1}{\mathrm{KL}(\varepsilon,\mu_{K_0+1})}(\frac{1}{4}\log\frac{K_0}{2\mathbb{E}_{\mu'_+}[L_T]} - \log 2)$ and the final result is obtained by using the explicit form for the bound on $\mathbb{E}_{\mu'_+}[L_T]$. □

3.3 An interesting class of algorithms

We introduce and analyse a new class of algorithms for the thresholding bandit problem that we call *index-based* algorithms. That class unifies several existing algorithms, including APT (Locatelli, Gutzeit, and Carpentier, 2016) and LSA (Tao et al., 2019).

3.3.1 A generic algorithm

An index-based algorithm pulls the minimum of K quantities, one for each arm, that each depends only on the rewards and pull counts of the respective arm (it does not change when pulling other arms). In particular, we consider algorithms for which the sampled arm is $i_{t+1} \in \arg\min_{k\in[K]} F(N_{k,t}, N_{k,t}\hat{\Delta}^2_{k,t}; a_k)$ for a function $F : \mathbb{N}\times\mathbb{R}_+\times\mathbb{R}_+^* \to \mathbb{R}$ that depends on the pull counts, the information about the sign and the weight of the arm.

Algorithm 3.1: Index-based algorithm for thresholding bandit

Inputs: an index function $F : \mathbb{N}\times\mathbb{R}_+\times\mathbb{R}_+^* \to \mathbb{R}$; $a_1,\dots,a_K \in \mathbb{R}_+^*$; $\sigma > 0$; and $\theta \in \mathbb{R}$

For $t = 1,\dots,T$ do

- for all $k \in [K]$ define

$$N_{k,t-1} = \sum_{s=1}^{t-1}\mathbb{I}\{k=i_s\},\ \hat{\mu}_{k,t-1} = \frac{1}{N_{k,t-1}}\sum_{s=1}^{t-1}\mathbb{I}\{k=i_s\}X_s,\ \hat{\Delta}^2_{k,t-1} = \frac{1}{2\sigma^2}\Big(\hat{\mu}_{k,t-1}-\theta\Big)^2$$

- pull $i_t \in \arg\min_{k\in[K]} F(N_{k,t-1}, N_{k,t-1}\hat{\Delta}^2_{k,t-1}; a_k)$.
- observe $X_t \sim \nu_{i_t}$

Define $t_{\max} = \max_{t\in[T]}\min_{k\in[K]} F(N_{k,t}, N_{k,t}\hat{\Delta}^2_{k,t}; a_k)$

Return for each $k \in [K]$ the sign $\hat{s}_k = \mathrm{sign}(\hat{\mu}_{k,t_{\max}} - \theta)$

After T rounds, the algorithm recommends the sign of the arms at the round $t_{\max} \in [T]$ at which $\min_{k\in[K]} F(N_{k,t}, N_{k,t}\hat{\Delta}_{k,t}^2; a_k)$ was maximal. This rule is used as opposed to returning the sign of all arms at time T to facilitate the analysis, which is based on the observation that there is a small probability of error when all arms have high index. The time $t_{\max}$ should be close to T: in particular, only one arm is sampled (possibly several times) between $t_{\max}$ and T (see Appendix A.1). In Section 3.4, we provide a generic analysis for index-based algorithms satisfying the assumption below.

Assumption 3.4. *The index function $F(n, x; a) : \mathbb{N} \times \mathbb{R}_+ \times \mathbb{R}_+^* \to \mathbb{R}$ is non-decreasing in n and x and $\lim_{n\to+\infty} F(n, ny; a) = +\infty$ for all $y > 0, a > 0$.*

Intuitively, algorithms that verify Assumption 3.4 prefer pulling arms that were pulled the least (smallest n) and whose quantity of information about the sign ($n\hat{\Delta}_{k,n}^2$) is small. This class includes several algorithms from the thresholding bandits literature: APT (Locatelli, Gutzeit, and Carpentier, 2016) for $F(n, x; a_k) = x$ and LSA (Tao et al., 2019) for $F(n, x; a_k) = x + \log(n)$ (these algorithms are only defined for $a_k = 1$). We now propose a generic method for designing an index-based algorithm.

3.3.2 Frank-Wolfe for Thresholding bandits

Our strategy to minimize the expected loss is inspired by the Frank-Wolfe algorithm (Frank and Wolfe, 1956) and aims at controlling an upper-bound on the loss, such as the right hand side of Inequality (3.4). Let's write that function as $B(N_T) = \sum_{k=1}^K a_k e^{-N_{k,T}\Delta_k^2}$. The high-level idea is to sequentially estimate its gradient and move to the minimizer of its linear approximation. If the gaps were known, we could compute at time $t+1$ the gradient of the bound with respect to N_t, $\nabla B(N_t) = (-a_k\Delta_k^2 e^{-N_{k,t}\Delta_k^2})_k$ and use the Frank-Wolfe algorithm. The algorithm would pull $i_{t+1} \in \arg\min_u u^\top \nabla B(N_t)$ for u in the simplex, which is simply $\arg\min_{k\in[K]}(-a_k\Delta_k^2 e^{-N_{k,t}\Delta_k^2})$. The gaps are however unknown. We therefore compute an estimate of the gaps $\hat{\Delta}_{k,t}$, with which we form the estimated gradient

$$\hat{\nabla} B(N_t)_k = -a_k\hat{\Delta}_k^2 e^{-N_{k,t}\hat{\Delta}_{k,t}^2} = -\exp\left(-\left(N_{k,t}\hat{\Delta}_{k,t}^2 - \log(N_{k,t}\hat{\Delta}_{k,t}^2) + \log\left(\frac{N_{k,t}}{a_k}\right)\right)\right).$$

This gives a natural choice for the index function of our algorithm $F(n, x; a_k) = x - \log x + \log(n/a_k)$. However, the latter is decreasing in x for $x \in (0, 1)$, which in addition to violating Assumption 3.4, may lead to instability in the initial phase when the gaps Δ_k are poorly estimated by $\hat{\Delta}_{k,n}$. We therefore propose a slight modification that preserves the asymptotic behavior of F and we call the resulting algorithm FWT (Frank-Wolfe for Thresholding bandits):

$$F(n, x; a_k) = \max\{x, 1\} - \log(\max\{x, 1\}) + \log(n/a_k)\,. \tag{FWT}$$

Recovering APT Using different upper-bounds B on the expected loss may lead to different algorithms. In particular, we highlight a link between our Frank-Wolfe inspired method and the APT algorithm of (Locatelli, Gutzeit, and Carpentier, 2016), which was designed to minimize the loss

$$L_T = \sum_{k=1}^{K} a_k \mathbb{I}\{\hat{s}_k \neq s_k\} = \sum_{k=1}^{K} a_k E_k \,.$$

Following our method with the choice $B(N_t) = \max_{k\in[K]} e^{-N_{k,t}\Delta_k^2}$ results in exactly the same sampling rule as the one of the APT algorithm (the recommendation rule differs slightly since we recommend the sign at $t_{\max}$ and not at T). Indeed, the derivative of B with respect to $N_{k,t}$ is nonzero (and negative) if and only if $N_{k,t}\Delta_k^2 = \arg\min_j N_{j,t}\Delta_j^2$ (ignoring the case in which there are several argmins, for which the tie breaking can be arbitrary). This leads to the choice $F(n, x; a_k) = x$ in Algorithm 3.1, which then pulls $i_{t+1} = \arg\min_{k\in[K]} N_{k,t}\hat{\Delta}_k^2$. This is the sampling rule of APT.

3.4 Loss upper bound

We provide a loss upper bound that is valid for all index-based algorithms verifying Assumption 3.4. We then give a summary of the analysis outline and the resulting loss bounds.

Theorem 3.5. *Let $K \geq 1$, $a_1, \ldots, a_K > 0$, $T \geq 1$, and $\sigma > 0$. Let $F : \mathbb{N} \times \mathbb{R} \times \mathbb{R}_+^* \to \mathbb{R}$ that satisfies Assumption 3.4. Let $C_1, \ldots, C_K > \max_k F(0, 0; a_k)$. For all $j, k \in [K]$, define*

- *$t_j(C_k)$ a solution of the equation $F(t, t\Delta_j^2; a_j) = C_k$,*
- *$S_k \subseteq [K]$ and $t_{j,0}(C_k) \in \mathbb{R}_+$, a set and values such that for $i \notin S_k$,*

$$\mathbb{P}\left(\exists n \leq t_{i,0}(C_k), F(n, n\hat{\Delta}_{n,i}^2; a_i) \geq C_k\right) = 1$$

Then the expected loss of Algorithm 3.1 is upper-bounded as

$$\mathbb{E}[L_T^{\mathbb{A}}] \leq \sum_{k=1}^{K} a_k \left(e \cdot \exp\left(-\frac{\frac{1}{2}\left(T - \sum_{j\notin S_k} t_{j,0}(C_k)\right) - \sum_{j\in S_k} t_j(C_k)}{\sum_{j\in S_k} 1/\Delta_j^2} \right) + T \cdot e^{-t_k(C_k)\Delta_k^2} \right) .$$

Before diving into the proof, we give a high-level intuition. The analysis is composed of two parts:

1. First we establish that for any arm $j \in [K]$, with large probability, there is a time $\tau_j(C_k)$ such that $F(\tau_j(C_k), \tau_j(C_k)\hat{\Delta}_{\tau_j(C_k),j}; a_j) \geq C_k$. We prove that for all $j, k \in [K], \tau_j(C_k)$ has

an exponential tail then use the fact that the algorithm pulls the minimal index to control the probability that the minimum never reaches C_k.

2. We show that if an arm's index is large, then the probability of mistake on it is small.

Remark 3.6. *The times $t_j(C_k)$ of Theorem 3.5 are the smallest numbers of samples such that $t_j(C_k) \geq \tau_j(C_k)$ with high enough probability. By determining those times, we derive explicit bounds for algorithms that verify Assumption 3.4.*

We note that our analysis is valid for a more general class than the one studied in this paper. Indeed, **1)** we analyze index functions slightly more general than $F(n, x; a_k)$. Each arm has a potentially different index function $F_k(n, x)$, and **2)** we consider algorithms which obey -a slightly more generic- Assumption 3.4, *i.e.* whose index can be written as $I_n^k = F_k(n, n\hat{\Delta}_{n,k}^2)$, where each F_k is non-decreasing in both variables, and $\lim_{n\to+\infty} F_k(n, ny) = +\infty$ for $y > 0$.

Proof roadmap: Set $(C_k)_{1\leq k\leq K} \in \mathbb{R}^+$, so that it immediately follows

$$\begin{aligned}
\mathbb{E}[L_T] &= \sum_k a_k \mathbb{P}(\hat{s}_k \neq s_k) \\
&\leq \sum_{k=1}^{K} a_k \mathbb{P}(\hat{s}_k \neq s_k \wedge \exists T' \in [T], I^k_{N_{T'}} \geq C_k) + a_k \mathbb{P}(\forall T' \in [T], I^k_{N_{T'}} < C_k) \\
&\leq \sum_{k=1}^{K} a_k \mathbb{P}(\hat{s}_k \neq s_k \wedge \exists T' \in [T], I^k_{N_{T'}} \geq C_k) + a_k \mathbb{P}(\overline{\mathcal{F}_{C_k}}) \,,
\end{aligned}$$

where $\overline{\mathcal{E}}$ stands for the complement of an event $\mathcal{E}$.

The proof proceeds in two steps, which are proved in Section 3.4.1 and Section 3.4.2, in order to control both probabilities introduced above:

1. Lemma 3.8: with large probability, there is some time t for which the index $I^k_{N_{k,t}}$ is large for all $k \in [K]$ (and all arms are well explored)
2. Theorem 3.10: if $I^k_{N_{k,t}}$ is large, then there is a small probability of mistake for arm k.

3.4.1 With large probability, all arms are well explored

We know from Lemma A.3 that any index based algorithm verifies $\mathcal{F}_C = \{\sum_k \tau_k(C) \leq T\}$. Hence, to prove that $\mathcal{F}_C$ happens with great probability, it suffices to show that $\sum_k \tau_k(C)$ has an exponential tail.

We derive a bound on $\mathbb{P}(\sum_k \tau_k(C) > T)$. To that end, we bound individually for each arm $\mathbb{P}(\tau_k(C) \geq t_k + x)$ for some t_k to be defined and $x \geq 0$, and conclude by Corollary A.7.

Theorem 3.7. *Under Assumption 3.4, the algorithm verifies, for all $x \geq 0$,*

$$\mathbb{P}(\sqrt{\tau_k(C)} > \sqrt{t_k(C)} + x) \leq \exp(-\Delta_k^2 x^2) \,,$$

with $t_k(C)$ solution to $F_k(t, t\Delta_k^2) = C$, if such a solution exists. Otherwise, if $C < F_k(0,0)$ and no solution exists, $\tau_k(C) = 0$ with probability 1.

Proof. We first bound $\mathbb{P}(\tau_k(C) > t_k + x)$, where t_k is chosen later, and $x \geq 0$.

$$\begin{aligned}\mathbb{P}(\tau_k(C) > t_k + x) = \mathbb{P}(\forall n \leq t_k + x,\ I_n^k < C) &\leq \mathbb{P}(I_{t_k+x}^k < C) \\ &= \mathbb{P}(F_k(t_k + x, (t_k + x)\hat{\Delta}_k^2) < C)\end{aligned}$$

First, by monotonicity of F_k, this probability equals zero if $C \leq F_k(t_k + x, 0)$. If $C > F_k(t_k + x, 0)$, we define $\delta_{t_k+x,k,C}$ such that $\Delta_{t_k+x,k}^{\delta_{t_k+x,k,C}} = \inf\{\Delta \geq 0 \mid C < F_k(t_k + x, (t_k + x)\Delta^2)\}$. In the following, we write $F_k(n, \cdot)^{-1}(C) := \inf\{x \mid C \leq F_k(n, x)\}$. If $F_k(n, \cdot)$ is increasing, this is its inverse, but we only suppose that $F_k(n, \cdot)$ is non-decreasing. Note that $x < F_k(n, \cdot)^{-1}(C)$ implies that $F_k(n, x) < C$. With that definition, $(t_k + x)(\Delta_{t_k+x,k}^{\delta_{t_k+x,k,C}})^2 = F_k(t_k + x, \cdot)^{-1}(C)$. As a consequence, we get

$$\begin{aligned}\mathbb{P}(\tau_k(C) > t_k + x) \leq \mathbb{P}(I_{t_k+x}^k < C) &= \mathbb{P}(F_k(t_k + x, (t_k + x)\hat{\Delta}_{t_k+x,k}^2) < C) \\ &\leq \mathbb{P}((t_k + x)\hat{\Delta}_{t_k+x,k}^2 < F_k(t_k + x, \cdot)^{-1}(C)) \\ &= \mathbb{P}(\hat{\Delta}_{t_k+x,k}^2 < (\Delta_{t_k+x,k}^{\delta_{t_k+x,k,C}})^2) \\ &\leq \delta_{t_k+x,k,C} \,,\end{aligned}$$

where by definition,

$$\begin{aligned}\delta_{t_k+x,k,C} &= \exp\left(-(t_k + x)(\Delta_k - \Delta_{t_k+x,k}^{\delta_{t_k+x,k,C}})\right) \\ &= \exp\left(-\left(\sqrt{\Delta_k^2(t_k + x)} - \sqrt{F_k(t_k + x, \cdot)^{-1}(C)}\right)^2\right) \,.\end{aligned}$$

We intend to prove an exponential decrease with x. In order to have it, we will set t_k such that the exponential is equal to 1 for $x = 0$, and then decreases as x grows.Let then t_k be such that $C \leq F_k(t_k, t_k\Delta_k^2)$. It exists as soon as $C \geq F_k(0,0)$ (where the later is non-positive for specific algorithms we will consider). For all $t \geq t_k$, $F_k(t, t\Delta_k^2) \geq F_k(t_k, t_k\Delta_k^2) \geq C$, which leads to $\sqrt{\Delta_k^2 t} - \sqrt{F_k(t, \cdot)^{-1}(C)} \geq 0$. Note that since F_k is non-decreasing in the first variable we have $F_k(t + x, \cdot)^{-1}(C) \leq F_k(t, \cdot)^{-1}(C)$ for all $t, x \geq 0$, and

$$\delta_{t_k+x,k,C} = \exp\left(-\left(\sqrt{\Delta_k^2(t_k + x)} - \sqrt{F_k(t_k + x, \cdot)^{-1}(C)}\right)^2\right)$$

$$\leq \exp\left(-\left(\sqrt{\Delta_k^2(t_k+x)} - \sqrt{F_k(t_k,\cdot)^{-1}(C)}\right)^2\right)$$
$$\leq \exp\left(-\Delta_k^2\left(\sqrt{t_k+x} - \sqrt{t_k}\right)^2\right) .$$

Let $Y_k = \max(\sqrt{\tau_k(C)}, \sqrt{t_k})$; we have proved that for all $x \geq 0$,

$$\mathbb{P}(Y_k > \sqrt{t_k+x}) \leq \exp\left(-\Delta_k^2\left(\sqrt{t_k+x} - \sqrt{t_k}\right)^2\right) .$$

By setting $x = 2\sqrt{\lambda t_k} + \lambda$ for $\lambda \geq 0$, we get $\mathbb{P}(Y_k > \sqrt{t_k} + \lambda) \leq \exp\left(-\Delta_k^2\lambda^2\right)$. □

Lemma 3.8. *For all $C > 0$, t_k such that $F(t_k, t_k\Delta_k^2) \geq C$.*

$$\mathbb{P}(\sum_k \tau_k(C) \geq T) \leq e \times \exp\left(-\frac{\frac{T}{2} - \sum_k t_k}{\sum_k 1/\Delta_k^2}\right).$$

Proof. Rewrite the event $\{\sum_k \tau_k(C) > T\}$ using $Y_k = \max(\sqrt{\tau_k(C)}, \sqrt{t_k})$, so that:

$$\begin{aligned}\mathbb{P}\left(\sum_k \tau_k(C) > T\right) &= \mathbb{P}\left(\sum_k ((Y_k - \sqrt{t_k}) + \sqrt{t_k})^2 > T\right) \\ &\leq \mathbb{P}\left(2\sum_k (Y_k - \sqrt{t_k})^2 + 2\sum_k t_k > T\right) \\ &\leq \mathbb{P}\left(\sum_k (Y_k - \sqrt{t_k})^2 > T/2 - \sum_k t_k\right) . \qquad (3.7)\end{aligned}$$

We now apply Lemma A.6 to $(Y_k - \sqrt{t_k})^2$, which verifies $\mathbb{P}((Y_k - \sqrt{t_k})^2 \geq x) \leq \exp(-\Delta_k^2 x)$. From Equation (3.7), we obtain

$$\mathbb{P}(\sum_k \tau_k(C) > T) \leq \mathbb{P}(\sum_k (Y_k - \sqrt{t_k})^2 > T/2 - \sum_k t_k) \leq e \times \exp\left(-\frac{T/2 - \sum_k t_k}{\sum_k 1/\Delta_k^2}\right) .$$

□

Remark We can actually derive a tighter bound than (3.7)

$$\begin{aligned}\mathbb{P}(\sum_k \tau_k(C) > T) &= \mathbb{P}(\sum_k ((Y_k - \sqrt{t_k}) + \sqrt{t_k})^2 > T) \\ &\leq \mathbb{P}\left(\left(\sqrt{\sum_k (Y_k - \sqrt{t_k})^2} + \sqrt{\sum_k t_k}\right)^2 > T\right) \\ &= \mathbb{P}\left(\sum_k (Y_k - \sqrt{t_k})^2 > (\sqrt{T} - \sqrt{\sum_k t_k})^2\right) .\end{aligned}$$

Roughly speaking, to get it, just write $\|(Y-\sqrt{t})+\sqrt{t}\|^2 \leq (\|Y-\sqrt{t}\|+\|\sqrt{t}\|)^2$. To get Equation (3.7), we further use $(\|Y-\sqrt{t}\|+\|\sqrt{t}\|)^2 \leq 2\|Y-\sqrt{t}\|^2+2\|\sqrt{t}\|^2$. In the case of APT (at least) it leads to the *same* final bound on the algorithm because when we optimize further down, we set $\sum_k t_k = T/4$ no matter which of these inequalities we use, value for which resulting exponents are equal.

Corollary 3.9. *Suppose now that there is a set S_C such that for $k \notin S_C$, $\mathbb{P}(\tau_k(C) > t_k) = 0$. Then we can refine Lemma 3.8 to*

$$\begin{aligned}\mathbb{P}(\sum_k \tau_k(C) > T) &\leq \mathbb{P}(\sum_{k\in S_C} \tau_k(C) > T - \sum_{k\notin S_C} t_k) \\ &\leq e \times \exp\left(-\frac{(T-\sum_{k\notin S_C} t_k)/2 - \sum_{k\in S_C} t_k}{\sum_{k\in S_C} 1/\Delta_k^2}\right).\end{aligned}$$

3.4.2 When an arm index is large, the probability of mistake is small

The goal of this section is to bound $\mathbb{P}(\hat{s}_k \neq s_k \wedge \exists T' \in [T], I^k_{N_{T'}} \geq C)$. We define the random variable $E_{k,t} = \mathbb{I}\{(\hat{\mu}_{k,t}-\theta)(\mu_k-\theta) < 0\}$; it is equal to 1 iff there is an error on the sign at time t. The algorithm makes a mistake on arm k is $E_{k,t_{\max}} = 1$ since it returns the sign at that time.

Theorem 3.10. *The algorithm using F_k for its index definition verifies*

$$\begin{aligned}\mathbb{P}(\hat{s}_k \neq s_k \wedge \exists T' \in [T], I^k_{N_{T'}} \geq C) &\leq \mathbb{P}(\exists T' \in [T], E_{k,T'} = 1 \wedge I^k_{N_{T'}} \geq C) \\ &\leq T \cdot \inf\{e^{-n_k\Delta_k^2} \mid F_k(n_k, n_k\Delta_k^2) < C\}.\end{aligned}$$

Proof. We use Lemma 3.13 (below): find n_k as large as possible such that $F_k(n_k, n_k\Delta_k^2) < C$. Then, since the algorithm returns the sign of the arm at the time at which its index was maximal, we get $\mathbb{P}(\hat{s}_k \neq s_k \wedge \exists T' \in [T], I^k_{N_{T'}} \geq C) \leq \mathbb{P}(\exists T' \in [T], E_{k,T'} = 1 \wedge I^k_{N_{T'}} \geq C) \leq T \cdot e^{-n_k\Delta_k^2}$. □

Lemma 3.11. *For any $\delta_k \in (0,1)$, with probability at least $1-\delta_k$, and for all $n \in [T]$, it holds*

$$\sqrt{N_{k,t}}(\hat{\mu}_t^k - \mu^k) \leq \sqrt{\log\left(\frac{T}{\delta_k}\right)}.$$

Proof. This is a direct implication of Hoeffding's inequality with a union bound for time-uniformity. □

Define $n_k = \frac{1}{\Delta_k^2}\log\left(\frac{T}{\delta_k}\right)$. Consider the following three facts (their definition will be useful for the following proofs):

1. If the concentration holds, then $\sqrt{N_{k,t}}(\hat{\mu}_t^k - \mu^k) \leq \sqrt{n_k}\Delta_k$.
2. If there is a mistake at time t, then we have

(a) $\hat{\mu}_t^k - \mu^k \geq \Delta_k$.

(b) $(\hat{\mu}_t^k - \mu^k)^2 \geq \hat{\Delta}^2_{N_{k,t},k} + \Delta_k^2$.

3. If $I^k_{N_{k,t}} > C$ then $F_k(N_{k,t}, N_{k,t}\hat{\Delta}^2_{t,k}) > C$ and $N_{k,t}\hat{\Delta}^2_{t,k} \geq F_k(N_{k,t}, \cdot)^{-1}(C)$.

Lemma 3.12. *If at time t, concentration holds and there is a mistake (1 and 2 are true), then $N_{k,t} \leq n_k$ and $F_k(N_{k,t}, (n_k - N_{k,t})\Delta_k^2) \geq I^k_{N_{k,t}}$.*

Proof. First point: combine 1 and 2(a). Second point: use 1, then 2(b), then the definition of $I^k_{N_{k,t}}$:

$$n_k\Delta_k^2 \geq N_{k,t}(\hat{\mu}_t^k - \mu^k)^2 \geq N_{k,t}\hat{\Delta}^2_{t,k} + N_{k,t}\Delta_k^2 \geq F_k(N_{k,t}, \cdot)^{-1}(I^k_{N_{k,t}}) + N_{k,t}\Delta_k^2 \,.$$

□

Lemma 3.13. *If at time t, all three "if" are true, then $F_k(n_k, n_k\Delta_k^2) \geq C$.*

Proof. Use the monotonicity of F_k in the inequality of Lemma 3.12.We have $N_{k,t} \leq n_k$ and $n_k - N_{k,t} \leq n_k$. □

3.5 Examples

Here we explicit the earlier theorem for some previously existing algorithms in this setting as well as our newly proposed one. Then, we provide a numerical comparison of the different bounds so as to provide an intuition to the reader interesting in comparing them.

3.5.1 Explicit bounds for FWT, LSA and APT

APT ((Locatelli, Gutzeit, and Carpentier, 2016)) We analyze the variant of this algorithm that returns the sign at the time $t_{\max}$ when the minimal index was maximal.

Corollary 3.14. *Suppose that for all $k \in [K]$, $a_k = 1$. For all $T \in \mathbb{N}^*$,*

$$\mathbb{E}[L_T^{APT}] \leq 2K\sqrt{e \cdot T} \cdot \exp\left(-\frac{1}{4}\frac{T}{\sum_{j=1}^K 1/\Delta_j^2}\right) .$$

Since $\max_k E_k \leq \sum_k E_k$, the bound of Corollary 3.14 is also a bound on the zero-one loss, which we can compare to the result of (Locatelli, Gutzeit, and Carpentier, 2016). Our result shows a $1/4$ factor in the exponential instead of the worse $1/32$ constant of the original paper.

Proof. This algorithm (in its variant that stops at $t_{\max}$) corresponds to $F(n,x) = x$. To apply Theorem 3.5 we find that $t_j(C_k) = \frac{C_k}{\Delta_j^2}$ is solution, then:

$$\mathbb{E}[L_T] \leq \sum_{k=1}^{K} e a_k \exp\left(-\left(\frac{T/2}{\sum_{j=1}^{K} 1/\Delta_j^2} - C_k\right)\right) + T \cdot a_k \exp\left(-C_k\right)$$

An optimal C_k is such that $e \cdot \exp\left(-\frac{\frac{T}{2}}{\sum_{j=1}^{K} 1/\Delta_j^2} + C_k\right) = T \cdot \exp(-C_k)$.

Then the bound becomes: $\mathbb{E}[L_T] \leq 2\sqrt{e \cdot T} \sum_{k=1}^{K} a_k \exp\left(-\frac{1}{4}\frac{T}{\sum_{k=1}^{K} 1/\Delta_k^2}\right)$. □

LSA ((Tao et al., 2019)) This algorithm corresponds to $F(n,x) = x + \log n$, we provide the following bound, which is significantly tighter than the one provided by its inventors.

Corollary 3.15. *For all $T \in \mathbb{N}^*$,*

$$\mathbb{E}[L_T^{LSA}] \leq \sum_{k=1}^{K} e \cdot a_k \exp\left(-\left(\frac{\frac{T}{2} - \sum_{j=1}^{K} \frac{1}{\Delta_j^2} W(\Delta_j^2 \exp(C_k))}{\sum_{j=1}^{K} 1/\Delta_j^2}\right)\right) + T a_k \exp\left(-W(\Delta_k^2 \exp(C_k))\right).$$

Furthermore, if $\forall k, j \quad C_k + \log \Delta_j^2 \geq 1$ we obtain:

$$\mathbb{E}[L_T^{LSA}] \leq \sum_{k=1}^{K} (1+e) \cdot a_k \exp\left(-\left(\frac{\frac{T}{2} + \sum_{j=1}^{K} \frac{1}{\Delta_j^2}(\log \frac{1}{\Delta_j^2} + \log(C_k + \log \Delta_j^2))}{\sum_{j=1}^{K} 1/\Delta_j^2}\right) + C_k\right)$$
$$+ \sum_{k=1}^{K} \frac{T(C_k + \log \Delta_k^2)}{\Delta_k^2} a_k \exp\left(-C_k\right). \tag{3.8}$$

Proof. The stopping time $t_j(C_k)$ is solution to $t\Delta_j^2 + \log t = C_k$. This equation has a closed form solution: $t_j(C_k) = \frac{1}{\Delta_j^2} W(\Delta_j^2 \exp(C_k))$, where W is the Lambert W function, and this entails the first bound.

Then, we can use an inequality on the Lambert function (*cf* Corollary 2.4 in (Hoorfar and Hassani, 2008)), For all $x \geq e$ we have

$$\log x - \log\log x \leq W(x) \leq \log x - \log\log x + \log\left(1 + e^{-1}\right),$$

this entails that if $\forall k, j \quad C_k + \log \Delta_j^2 \geq 1$ we obtain the second more specific bound. □

FWT (our algorithm) This algorithm corresponds to $F_k(n,x) = \max(1,x) - \log\max(1,x) + \log n - \log a_k$. Using our generic analysis, we are able to prove the following result about *FWT*.

Corollary 3.16. *Let S, S' be two sets with $S' \subseteq S \subseteq [K]$ and let $C \in \mathbb{R}$ be such that $C \geq 1 + \max_{k\in S} \log \frac{1}{a_k\Delta_k^2}$. Then, for all $T \geq 1$*

$$\mathbb{E}[L_T^{FWT}] \leq \sum_{k\notin S'} a_k + e\sum_{k\in S'} a_k \exp\left(-\frac{\frac{1}{2}(T - \sum_{j\notin S} a_j e^{C-1}) + \sum_{j\in S}\frac{1}{\Delta_j^2}\log\frac{1}{a_j\Delta_j^2}}{\sum_{j\in S} 1/\Delta_j^2} + C\right) + T\sum_{k\in S'} a_k \exp\left(-C + \log(1/(a_k\Delta_k^2))\right).$$

Moreover, if we have $T \geq 2\sum_{j=1}^K \frac{1}{\Delta_j^2}(2 + \log\frac{a_j\Delta_j^2 \max_i a_i\Delta_i^2}{(\min_k a_k\Delta_k^2)^2} - \log\frac{T}{e^3})$, we get the bound

$$\mathbb{E}[L_T] \leq 2\sqrt{eT}\sum_k a_k \exp\left(-\frac{1}{2}\frac{T/2 - \sum_j \frac{1}{\Delta_j^2}\log\frac{a_j\Delta_j^2}{a_k\Delta_k^2}}{\sum_j 1/\Delta_j^2}\right).$$

Up to a $1/4$ factor, this is the exponent of the optimal non-adaptive oracle (cf *Equation 3.4*)

Proof. In order to find the times $t_j(C_k)$ of Theorem 3.5 we solve the equation:

$$\max(1, n\Delta_k^2) - \log\max(1, n\Delta_k^2) + \log(n\Delta_k^2) = C + \log a_k + \log\Delta_k^2 .$$

Let $D_k = C + \log a_k + \log\Delta_k^2$. We want a solution to $\max(1,x) - \log\max(1,x) + \log x = D_k$. The function on the left, which we now denote by $\mathcal{I}$, is increasing and bijective from $\mathbb{R}^+$ to $\mathbb{R}$.

- If $D_k \geq 1$, $\mathcal{I}(D_k) = D_k$.
- If $D_k \leq 1$, $\mathcal{I}(e^{D_k-1}) = D_k$.
- For all $D_k > 0$, $\mathcal{I}(e^{D_k-1}) \geq D_k$.

Moreover, we have $F_k(a_k e^{C-1}) \geq C$, it comes $\mathbb{P}(\tau_k(C) > a_k e^{C-1}) = 0$.
Let $C_k > 0$ and $S_k \subseteq \{j \in [K] \mid C_k + \log a_j\Delta_j^2 \geq 1\}$. Let S' be a set such that for all $k \in S'$, $k \in S_k$.

$$\begin{aligned}\mathbb{E}[L_T] &\leq \sum_{k\notin S'} a_k + \sum_{k\in S'} a_k\mathbb{P}(\overline{\mathcal{F}_{C_k}}) + \sum_{k\in S'} a_k\mathbb{P}(\overline{\mathcal{E}_k(T)}\cap\mathcal{F}_{C_k}) \\ &\leq \sum_{k\notin S'} a_k + e\sum_{k\in S'} a_k \exp\left(-\frac{\frac{1}{2}(T - \sum_{j\notin S_k} a_j e^{C_k-1}) - \sum_{j\in S_k}\frac{1}{\Delta_j^2}\log a_j\Delta_j^2}{\sum_{j\in S_k} 1/\Delta_j^2} + C_k\right)\end{aligned}$$

$$+ T \sum_{k \in S'} a_k \exp\left(-C_k - \log a_k \Delta_k^2\right) .$$

Large T: The value of C_k which equalizes the two terms indexed by k verifies

$$C_k = \frac{1}{2} \frac{\frac{1}{2}(T - \sum_{j \notin S_k} a_j e^{C_k - 1}) - \sum_{j \in S_k} \frac{1}{\Delta_j^2} \log a_j \Delta_j^2}{\sum_{j \in S_k} 1/\Delta_j^2} - \frac{1}{2} \log a_k \Delta_k^2 + \frac{1}{2} \log \frac{T}{e} .$$

The latter can be chosen if T is big enough such that $S_k = [K]$ for all arms, this is the case if $T \geq 2 \sum_{j=1}^K \frac{1}{\Delta_j^2}(2 + \log \frac{a_j \Delta_j^2 \max_i a_i \Delta_i^2}{(\min_k a_k \Delta_k^2)^2} - \log \frac{T}{e^3})$, this entails the second bound of the corollary.

General T: We choose a set $S \subseteq [K]$ and set $C_k = C$, a common value still to be determined, for all $k \in S'$. Then for all $k \in S'$, we set $S_k = S$. We impose $C \geq 1 + \max_{j \in S} \log \frac{1}{a_j \Delta_j^2}$, such that the condition $S_k \subseteq \{j \in [K] \mid C_k + \log a_j \Delta_j^2 \geq 1\}$ is verified. This finished the proof for the first bound of the corollary. □

Remark 3.17. *Note that the bounds for LSA and FWT are actually very similar, indeed Theorem 3.5 applies to LSA and FWT with the following times:*

- *LSA:* $t_j(C_k) = W(e^{C_k} \Delta_j^2)/\Delta_j^2$ *and* $t_{j,0}(C_k) = e^{C_k}$,
- *FWT:* $t_j(C_k) = \log(e^{C_k} a_j \Delta_j^2)/\Delta_j^2$ *and* $t_{j,0}(C_k) = a_j e^{C_k - 1}$,

Therefore, for the two algorithms, the times $t_j(C_k)$ are close (equal up to the $\log \log$ terms in W), thus their bounds are close as well. Note that LSA is only defined for $a_j = 1$ for all j.

We highlight a notable property, in the regime where T is sufficiently large, *FWT* recovers the same exponent as in the non-adaptive oracle loss bound (3.5) (up to a factor $1/4$). In the same regime of large T, the bound that we obtain for LSA is of the same order, but less explicit due to the function W. The latter is still impressive since the original theorem of (Tao et al., 2019) for LSA exhibits an exponent significantly looser, of order $\exp\left(-\frac{1}{16020} \frac{T}{\sum_{j=1}^K 1/\Delta_j^2}\right)$, *i.e.* 4005-times worse than our bound. We finally derive a bound for our newly introduced algorithm.

3.5.2 Numerical comparison

Figure 3.4 compares the upper-bounds of Corollary 3.14 (APT), Theorem 3.5 (see also Equation (3.8)) (LSA), and Corollary 3.16 (FWT) for the particular case $\Delta_i = (i/K)^2$ and $a_i = 1$, for $i = 1, \ldots, K = 50$. We can see that while the bounds of LSA, APT, and FWT are asymptotically similar, that of FWT starts to be significant for much smaller T. On the right, we can see the importance of the set S' in Corollary 3.16: the bounds first ignores all the arms, and suffers a

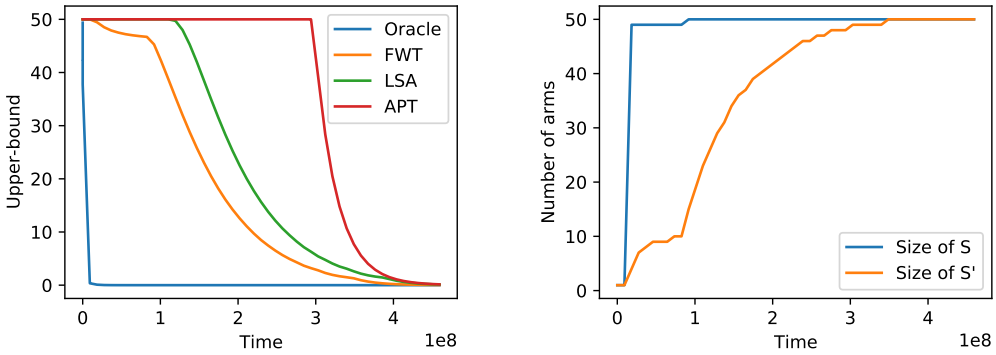


Figure 3.4 – [left] Comparison of the upper-bounds of Corollaries 3.14, 3.16 and that of the optimal non-adaptive oracle of Equation (3.5) (blue) when the gaps are of the form $\Delta_i = (i/K)^2$. [right] Evolution over time of the size of the optimal sets S (blue) and S' (orange) that minimize the bound of Corollary 3.16.

loss of 1 and then adds them one by one as soon as they can be classified. The bound derived in (Tao et al., 2019) for LSA is not represented on the figures, since it is still bigger than K for the considered range of T.

Figure 3.5 compares the upper-bounds of Corollary 3.14 (APT), Equation (3.8) (LSA), and Corollary 3.16 (FWT) for the particular case $\Delta_i = i/K$ and $a_i = 1$, for all $i = 1, \dots, K$ and $K = 50$. We observe a behavior similar to that of Figure 3.4.

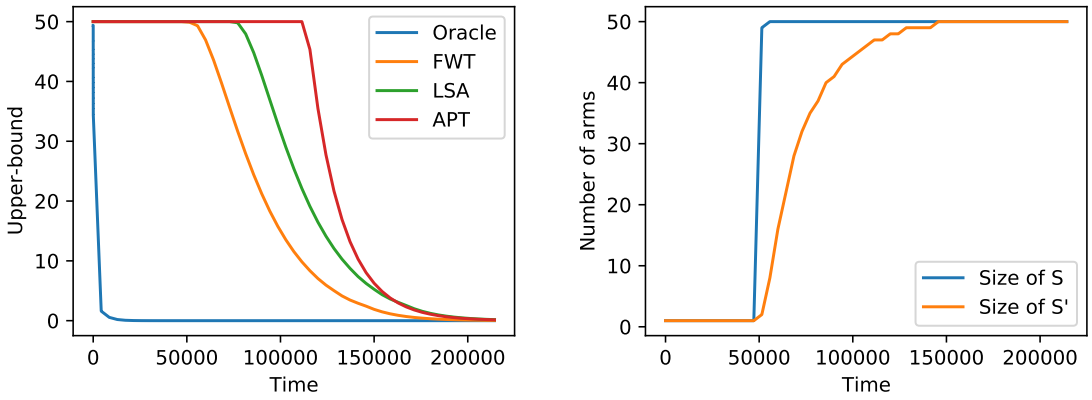


Figure 3.5 – [left] Comparison of the upper-bounds of Corollaries 3.14, 3.16 and that of the optimal non-adaptive oracle of Equation (3.5) (blue) when the gaps are of the form $\Delta_i = i/K$. [right] Evolution over time of the size of the optimal sets S (blue) and S' (orange) that minimize the bound of Corollary 3.16.

3.5.3 The sum-of-gaps objective

We show that our method applies for the sum-of-gaps objective $\sum_{k=1}^{K} \Delta_k E_k$. This is not a particular case of the setting discussed previously since a_k was known to the algorithm, while Δ_k is unknown. It serves as a proof of concept for the extensibility of our method. The index given by FWT in this setting is $F(n,x) = x' - \frac{3}{2}\log(x') + \frac{3}{2}\log(n)$, where $x' = \max\left(x, \frac{3}{2}\right)$. We can then bound the sum-of-gaps loss using our generic analysis by proceeding similarly to Theorem (3.5).

Corollary 3.18. (***FWT for the sum-of-gaps objective***) *In the large horizon regime, specifically when* $T \geq 2\sum_{j=1}^{k} \frac{1}{\Delta_j^2}\left(3 + 3\log\frac{\Delta_j \max_i \Delta_i}{(\min_i \Delta_i)^2} - \log\frac{T}{e}\right)$, *then we show that*

$$\mathbb{E}[\sum_{k=1}^{K} \Delta_k E_k] \leq 2\sqrt{eT}\sum_k \Delta_k \exp\left(-\frac{1}{2}\frac{\frac{T}{2} + \sum_j \frac{3}{2}\frac{1}{\Delta_j^2}\log\frac{\Delta_k^2}{\Delta_j^2}}{\sum_j 1/\Delta_j^2}\right).$$

This can be useful for applications in which errors are more tolerated for arms that are close to the threshold.

Proof. The global loss we investigate in this section is

$$L_T = \sum_k \Delta_k E_k.$$

First we write the Frank-Wolfe index: $\arg\min_k N_{k,t}\hat{\Delta}_{k,t}^2 - \frac{3}{2}\log\left(N_{k,t}\hat{\Delta}_{k,t}^2\right) + \frac{3}{2}\log(N_{k,t})$, then we slightly modify it to comply with Assumption 3.4 (see explanation above Equation FWT):

$$I_{N_{k,t}}^k = \max(\frac{3}{2}, N_{k,t}\hat{\Delta}_{k,t}^2) - \frac{3}{2}\log\max(\frac{3}{2}, N_{k,t}\hat{\Delta}_{k,t}^2) + \frac{3}{2}\log N_{k,t}.$$

This corresponds to the index $F(n,x) = \frac{3}{2}\log n + \max(\frac{3}{2}, x) - \frac{3}{2}\log\max(\frac{3}{2}, x)$.
Solving $F(n, n\Delta_k^2) = C$ gives rise to two cases:

- $n = \frac{1}{\Delta_k^2}(C + \frac{3}{2}\log\Delta_k^2)$ if $C + \frac{3}{2}\log\Delta_k^2 \geq \frac{3}{2}$,
- $n = \frac{3}{2}\exp(\frac{2}{3}C - 1)$ otherwise. In that case, $n \leq \frac{3}{2}\frac{1}{\Delta_k^2}$.

Also, $n = \frac{3}{2}\exp(\frac{2}{3}C - 1)$ is solution to $F(n,0) = C$. Consider $C_k > 0$, let $S_k = \{j \in [K] \mid C_k + \frac{3}{2}\log\Delta_j^2 \geq \frac{3}{2}\}$ and S' be a set such that for all $k \in S'$, $k \in S_k$, then

$$\mathbb{E}[L_T] \leq \sum_{k \notin S'} \Delta_k + \sum_{k \in S'} \Delta_k \mathbb{P}(\overline{\mathcal{F}_{C_k}}) + \sum_{k \in S'} \Delta_k \mathbb{P}(\overline{\mathcal{E}_k(T)} \cap \mathcal{F}_{C_k})$$

$$\leq \sum_{k \notin S'} \Delta_k + e \sum_{k \in S'} \Delta_k \exp\left(- \frac{\frac{1}{2}(T - \sum_{j \notin S_k} \frac{3}{2}\Delta_j e^{\frac{2}{3}C_k - 1}) + \sum_{j \in S_k} \frac{3}{2}\frac{1}{\Delta_j^2} \log \frac{1}{\Delta_j^2}}{\sum_{j \in S_k} 1/\Delta_j^2} + C_k \right)$$
$$+ \sum_{k \in S'} T\Delta_k \exp\left(-C_k - \frac{3}{2}\log \Delta_k^2\right)$$

Large T The values of C_k that optimize the r.h.s of the previous inequality verify:

$$C_k = \frac{1}{2} \frac{\frac{1}{2}(T - \sum_{j \notin S_k} \frac{3}{2}\Delta_j e^{\frac{2}{3}C_k - 1}) + \sum_{j \in S_k} \frac{3}{2}\frac{1}{\Delta_j^2} \log \frac{1}{\Delta_j^2}}{\sum_{j \in S_k} 1/\Delta_j^2} - \frac{1}{2}\log \Delta_k^3 + \frac{1}{2}\log \frac{T}{e}$$

If T is big enough such that $S_k = [K]$ for all arms, which is true in the case considered in the corollary, then we prove the bound of the latter. □

3.6 Additional Experiments

In this section, we illustrate on synthetic data the performance of APT, LSA, and FWT. The implemented algorithms respectively correspond to Algorithm 3.1 with the following choices:

$$F(n, x; 1) = x \qquad \text{(APT)}$$
$$F(n, x; 1) = x + \log(n) \qquad \text{(LSA)}$$
$$F(n, x; 1) = (1 + \sqrt{x})^2 - \log(1 + \sqrt{x})^2 + \log(n) \qquad \text{(FWT)}$$

Note that we used a slighly different version for APT than the one proposed in the analysis $F(n, x; 1) = \max\{1, x\} - \log \max\{1, x\} + \log(n)$. The analysis and experiments work similarly for both versions. But the $(1 + \sqrt{x})^2$ version performs slightly better empirically while the $\max\{1, x\}$ version provides cleaner theoretical results. The experiments are averaged over 500 runs and consider arm distributions of the form $\nu_k = \mathcal{N}(\mu_k, 1)$. The gaps are thus $\Delta_k = |\mu_k|$, for $k = 1, \ldots, K$. The performance criterion is the sum of errors defined in Equation 3.4 with $a_k = 1$. The experiments were run on a personal laptop with Intel Core i5, Dual Core, 3.1 GHz.

Since most of the tested experiments obtained similar performance, we only provide the results of a few experiments. Although our theoretical upper bounds are slighly better for FWT, LSA and FWT generally have similar performance, while APT underperforms. This last point is not surprising since, although we provide in Corollary 3.14 an upper bound for APT that appears asymptotically similar to those of LSA and FWT, APT was not designed to minimize the sum of errors. APT was made to minimize the probability of making at least one error and thus focuses too much on arms with very small gaps that are very difficult to classify.

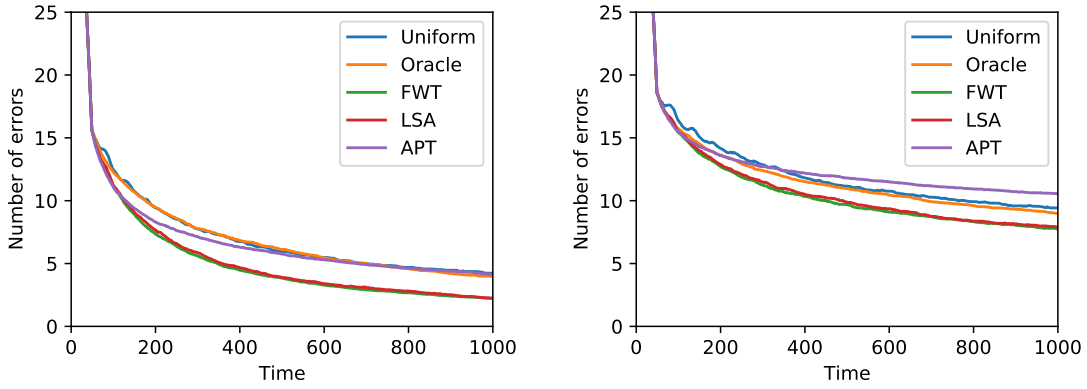


Figure 3.6 – Sum of errors over time of the different algorithms when $\mu_k = \frac{(-1)^k k}{K}$ [left] and $\mu_k = \frac{(-1)^k k^2}{K^2}$ [right] for $K = 50$ arms.

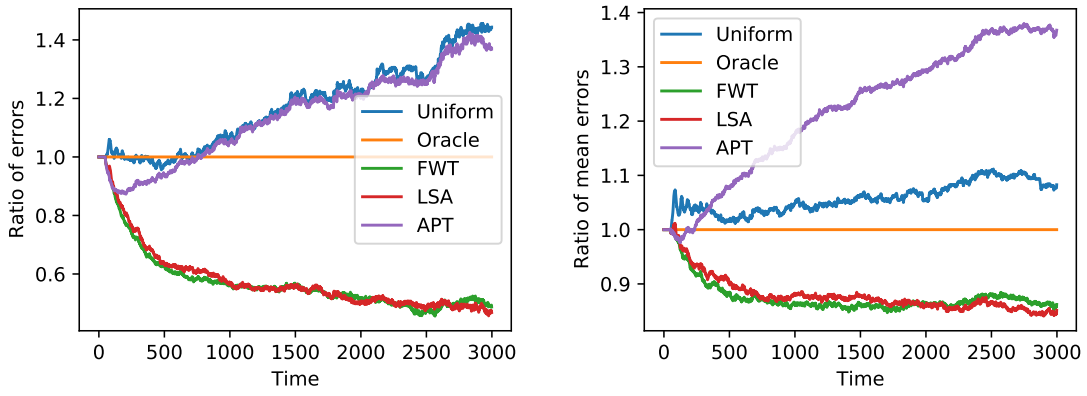


Figure 3.7 – Ratio of improvement with respect to the non adaptive oracle sampling when $\mu_k = \frac{(-1)^k k}{K}$ [left] and $\mu_k = \frac{(-1)^k k^2}{K^2}$ [right] for $K = 50$ arms.

Figure 3.6 shows the performance of the algorithms together with the non-adaptive oracle of Section 3.7. Figure 3.7 plots the ratio of error with respect to the non-adaptive oracle. Interestingly, in all of our experiments, APT and FWT perform better than it.

3.7 Beating the oracle? The benefits of adaptivity.

We argue that in some situations adaptive algorithms can greatly outperform the non-adaptive oracle of Section 3.2.2, i.e., the cost of non-adaptivity can be much higher than the cost of learning. The algorithms in the family we considered are all adaptive in the sense that they adapt their drawing strategy as more information is observed, at the cost of learning the parameter μ_k. We illustrate the benefits of adaptivity in the following toy example.

The "optimal" non-adaptive algorithm may be worse than adaptive algorithms. Consider the following parametric problem. An arm distribution is parametrized by $x \in \mathbb{R}$ and is supported on $\{0, x\}$; a sample of that distribution is equal to 0 or x, each with probability $1/2$. We assume that all arms have non-zero parameter and we will compute the optimal non-adaptive allocation.

We make the convention that if an algorithm sees only zeros for one arm, it returns any sign with probability $1/2$. The error probability of a non-adaptive allocation N_T^k for arm k is half of the probability of seeing only zeros (since if anything else is observed, the arm can be classified with perfect accuracy). Hence the total error is

$$\mathbb{E}[L_T] = \frac{1}{2}\sum_{k=1}^{K} \frac{1}{2^{N_{k,T}}} \geq \frac{K}{2^{(T/K)+1}},$$

which is minimized with the uniform allocation: $N_{k,T} = \frac{T}{K}$ for all $k \in [K]$.

Consider now an adaptive procedure that sample each arm in turn, but stops sampling an arm as soon at it sees a non-zero value. We crudely prove an upper bound for its number of errors, by remarking that it is zero if the algorithm classifies all arms correctly and smaller than K otherwise. The number of samples required to perfectly classify an arm follows a geometric distribution with parameter $1/2$. As a consequence, the number of required samples to classify all arms correctly follows a negative binomial $\mathrm{NB}(K, 1/2)$. Let Z be such a negative binomial random variable. The expected number of errors of the adaptive procedure is up to $K\mathbb{P}(Z > T)$. It then verifies

$$\mathbb{E}[L_T] \leq K\mathbb{P}(Z > T) \leq Ke^{-(\log(2)/2)T}\mathbb{E}e^{(\log(2)/2)Z} = \frac{K}{2^{T/2}}\left(1 + \frac{1}{\sqrt{2}}\right)^K,$$

where the value $\log(2)/2$ is chosen for simplicity (in $[0, \log 2)$). In the regime where T is large, this is of order $1/2^{T/2}$, which for $K > 2$ is much smaller than $1/2^{T/K}$ for the uniform allocation.

This toy example differs drastically from more realistic situations, as one non-zero sample for an arm is sufficient to know the sign of the expectation perfectly. We therefore consider empirically more reasonable frameworks, closer to those analyzed in the paper: the distributions of K arms are either $\mathcal{N}(1, 1)$ or $\mathcal{N}(-1, 1)$. Since all gaps Δ_i are equal, the optimal non-adaptive oracle is also the uniform sampling. The results are illustrated on the left part of Figure 3.8 and highlight the fact that all the adaptive algorithms considered (APT, LSA or FWT) drastically outperform the oracle. The right part of the figure shows the same phenomenon on another example in which the gaps are not constant. In particular, we can see that FWT and LSA have similar performance while APT (not designed for this purpose) generally suffers from a larger error. This result was corroborated by most of our experiments. We refer to Appendix 3.6 for more details.

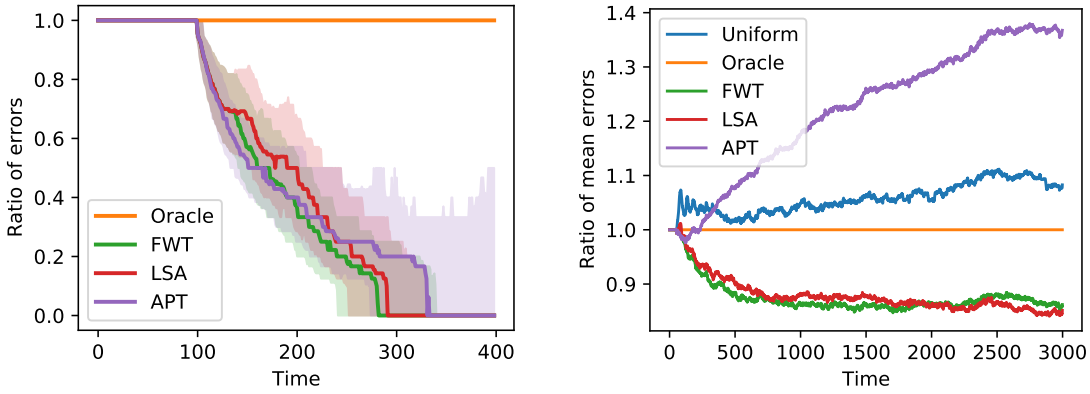


Figure 3.8 – [left] Median (and $0.25, 0.75$ empirical quantiles obtained on 500 runs) of the ratio between the error suffered by each algorithm and that of the optimal non-adaptive oracle ($\mu_k = (-1)^k, k = 1, \dots, 100$). [right] Ratio of the averaged errors (over 500 runs) of each algorithm with that of the oracle ($\mu_k = (-1)^k (k/K)^2, k = 1, \dots, 50$).

3.8 Discussion

An interesting research direction is to consider objective functions more general than (3.1). In particular, we believe that our approach (algorithm and analysis) can be generalized to loss functions that have the form $L_T = \sum_{k=1}^{K} f(\Delta_k, E_k)$ under certain regularity assumptions on f. Moreover, we focused on separable losses (hence linear wlog) and the index based algorithms we analyze reflect that separability. An obvious and intriguing direction for further work is to replace that assumption. One might for example want to design an algorithm that minimizes the probability of making more than a given number of mistakes.

The fact that adaptive algorithms can beat non-adaptive oracles has already been observed empirically for fixed confidence identification (Simchowitz, Jamieson, and Recht, 2017; Degenne, Koolen, and Ménard, 2019), although only in cases where the non-adaptive oracle was worse only for small times and was still asymptotically optimal. The phenomenon we observe for fixed budget thresholding is much more significant and remains to be explained by theoretical arguments. Currently, the best theoretical bound for adaptive algorithms is still a factor $1/4$ away in the exponent from the non-adaptive oracle bound.

Chapter 4

Linear regression: an improved algorithm & application to linear bandits

We consider the problem of online linear regression in the stochastic setting. We derive high probability regret bounds for online *ridge* regression and the *forward* algorithm. This enables us to compare online regression algorithms more accurately and eliminate assumptions of bounded observations and predictions. Our study advocates for the use of the forward algorithm in lieu of ridge due to its enhanced bounds and robustness to the regularization parameter. Moreover, we explain how to integrate it in algorithms involving linear function approximation to remove a boundedness assumption without deteriorating theoretical bounds. We showcase this modification in linear bandit settings where it yields improved regret bounds. Last, we provide numerical experiments to illustrate our results and endorse our intuitions.[1]

Contents

[1]This chapter is based on a collaboration with Odalric Maillard and Vianney Perchet (Ouhamma, Maillard, and Perchet, 2021). It was accepted for publication at the *32nd conference on advances in Neural Information Processing Systems* (*NeurIPS*).

4.1 Introduction and preliminaries

The *forward regression* algorithm, popularized in (Vovk, 2001; Azoury and Warmuth, 2001), shows competitive performance bounds in the challenging setup of online regression with *adversarial bounded* observations. We revisit the analysis of this strategy in the practically relevant alternative situation of *stochastic* linear regression with sub-Gaussian noise, hence possibly *unbounded* observations. When compared to the classical ridge regression strategy - its natural competitor - the existing analysis in the adversarial bounded case suggests the forward algorithm has higher performances. It is then natural to ask whether this conclusion holds for the stochastic setup. However, we show that in the stochastic setup, the existing adversarial analysis does not seem sufficient to draw conclusions, as it does not capture some important phenomena, such as the concentration of the parameter estimate around the regression parameter. It may further lead the practitioner to use an improper tuning of the regularization parameter. In order to overcome these issues, we revisit the analysis of the forward algorithm in the case of unbounded sub-Gaussian linear regression and provide a high probability regret bound on the performance of the forward and ridge regression strategies. Owing to this refined analysis, we show that the forward algorithm is superior in this scenario as well, but for different reasons than what is suggested by the adversarial analysis. We discuss the implications of this result in a practical application: stochastic linear bandits, both from theoretical and experimental perspectives.

Setup: In the classical setting of online regression with the square loss, an environment initially chooses a sequence of feature vectors $\{x_t\}_t \in \mathbb{R}^d$ together with corresponding observations $\{y_t\}_t \in \mathbb{R}$. Then, at each decision step t, the learner receives feature vector x_t and must output a prediction $\hat{y}_t \in \mathbb{R}$. Afterwards, the environment reveals the true label y_t and iteration $t+1$ begins. In this article, we focus on the case when the data generating process is a *stochastic* linear model:

$$\exists \theta_* \in \mathbb{R}^d \text{ such that } \forall t \in \mathbb{N}^* : \quad y_t = x_t^\top \theta_* + \varepsilon_t,$$

where $\{\varepsilon_t\}_t$ is a noise sequence. At iteration t, strategy $\mathcal{A}$ computes a parameter $\theta_{t-1}^{\mathcal{A}}$ to predict $\hat{y}_t^{\mathcal{A}} = x_t^\top \theta_{t-1}^{\mathcal{A}}$. In the sequel, we omit the subscript $\mathcal{A}$ when the algorithm is clear from context. The learner's prediction incurs the loss: $\ell_t^{\mathcal{A}} \triangleq \ell(x_t^\top \theta_{t-1}, y_t) = (\hat{y}_t - y_t)^2$, the learner then updates its prediction θ_{t-1} to θ_t and so on. The total cumulative loss at horizon T is denoted $L_T^{\mathcal{A}} = \sum_{t=1}^T \ell_t^{\mathcal{A}}$. We also let $\ell_t(\theta) = \ell(x_t^\top \theta, y_t)$ (resp. $L_T(\theta) = \sum_{t=1}^T \ell_t(\theta)$) be the instantaneous (resp. cumulative) loss incurred by predicting θ at time t (resp. $\forall t = 1, \dots, T$). Online regression algorithms are evaluated using different regret definitions, in the form of a relative cumulative loss to a batch loss; The quantity of interest in this paper is:

$$R_T^{\mathcal{A}} = L_T^{\mathcal{A}} - \min_\theta L_T(\theta). \tag{4.1}$$

From the perspective of online learning theory, online regression algorithms are usually designed for an *adversarial* setting, assuming an arbitrary bounded response variable $|y_t| \leq Y$ at each time step. While the mere existence of algorithms with tight guarantees in this general setting is remarkable, a practitioner may also consider alternative settings, in which analysis for the adversarial setup may be overly conservative. For illustration, we focus on the practical setting of bounded parameter $\|\theta_*\|_2 \leq S$ and i.i.d zero-mean σ-sub-Gaussian noise sequences:

$$\forall t \geq 1, \gamma \in \mathbb{R}: \quad \mathbb{E}\left[\exp(\gamma\varepsilon)\right] \leq \exp(\sigma^2\gamma^2/2).$$

We emphasize that while previous results in literature are valid for the adversarial bounded setting, we will still shed new light on the performance of these strategies in a stochastic unbounded setup, which is neither more general nor more restrictive than the adversarial one, and discuss their implications for the practitioner. Let us recall the two popular online regression algorithms considered.

Online ridge regression [Algorithm 4.1]: This folklore algorithm is defined in the online setting as the greedy version of batch ridge regression:

$$\theta_t^{\mathtt{r}} \in \arg\min_{\theta} L_t(\theta) + \lambda||\theta||_2^2, \tag{4.2}$$

where λ is a parameter and $\lambda||\theta||_2^2$ is a regularization used to penalize model complexity.

Algorithm 4.1: Online ridge regression

Given $\theta_0 \in \mathbb{R}^d$
for $t = 1, \ldots, T$ **do**
  observe $\mathrm{x}_t \in \mathbb{R}^d$ and predict $\hat{y}_t = x_t^\top \theta_{t-1}^{\mathtt{r}} \in \mathbb{R}$
  observe y_t and incur loss $\ell_t \in \mathbb{R}$
  update parameter: $\theta_t^{\mathtt{r}} \in \arg\min_\theta L_t(\theta) + \lambda||\theta||_2^2$
end

A solution to the quadratic optimization problem of Equation 4.2 is given in closed form, by $\theta_t^{\mathtt{r}} = G_t(\lambda)^{-1} b_t$, where $G_t(\lambda) = \lambda I + \sum_{q=1}^t x_q x_q^\top$ and $b_t = \sum_{q=1}^t x_q y_q$. We may further denote G_t instead of $G_t(\lambda)$ when λ is clear from context.

The forward algorithm [Algorithm 4.2]: A subtle change to the ridge regression takes advantage of the next feature x_{t+1} to better adapt to the next loss:

$$\theta_t^{\mathtt{f}} \in \arg\min_{\theta} L_t(\theta) + (x_{t+1}^\top \theta)^2 + \lambda||\theta||_2^2. \tag{4.3}$$

Algorithm 4.2: The forward algorithm

Given $\theta_0 \in \mathbb{R}^d$
for $t = 1, \dots, T$ **do**
  observe $\mathrm{x}_t \in \mathbb{R}^d$
  update parameter: $\theta^{\mathtt{f}}_{t-1} \in \arg\min_\theta L_{t-1}(\theta) + (x_t^\top \theta)^2 + \lambda ||\theta||_2^2$
  predict $\hat{y}_t = x_t^\top \theta^{\mathtt{f}}_{t-1} \in \mathbb{R}$
  observe y_t and incur loss $\ell_t \in \mathbb{R}$
end

Equivalently, the update step can be written: $\theta^{\mathtt{f}}_t = G^{-1}_{t+1} b_t$, where G_t is still defined same as before. Intuitively, the term $(x_{t+1}^\top \theta)^2$ in Equation 4.3 is a "predictive loss", a penalty on the parameter θ in the direction of the new feature vector x_{t+1}. This approach can be linked to transductive methods for regression (Cortes and Mohri, 2007; Tripuraneni and Mackey, 2019). (Tripuraneni and Mackey, 2019) describe two algorithms for linear prediction in supervised settings, and leverage the knowledge of the next test point to improve the prediction accuracy. However, these algorithms have significant computational complexities and are not adapted to online settings.

Related work Linear regression is perhaps one of the most known algorithms in machine learning, due to is simplicity and explicit solution. In contrast with the *batch* setting (when all observations are provided), *online* linear regression started receiving interest relatively recently. The first theoretical analyses date back to (Foster, 1991; Littlestone, Long, and Warmuth, 1991; Cesa-Bianchi, Long, and Warmuth, 1996; Kivinen and Warmuth, 1997). Under the assumption that the response variable is bounded $|y_t| \leq Y$, it has been shown that the forward algorithm (Vovk, 2001; Azoury and Warmuth, 2001) achieves a relative cumulative online error of $dY^2 \log(T)$ compared to the best batch regression strategy. This bound holds *uniformly* over bounded response variables and competitor vectors, and is 4 times better than the corresponding bound derived for online *ridge* regression.

Bartlett et al. (2015) studies minimax regret bounds for online regression, and ingeniously removed a dependence on the scale of features in existing bounds by considering the beforehand-known features setting, where all feature points $\{x_t\}_{1 \leq t \leq T}$ are known before the learning starts. Moreover, they derive a "backward algorithm" that is optimal under certain intricate assumptions on observations and features. Later on, (Malek and Bartlett, 2018) proves that under new (tricky) assumptions on observed features and labels the *backward algorithm* is not only optimal but applicable in sequential settings as well. More recently, (Gaillard et al., 2019) provides an optimal algorithm in the setting of beforehand known features without imposing stringent conditions as in (Bartlett et al., 2015; Malek and Bartlett, 2018). The latter shows that the forward algorithm with $\lambda = 0$ yields a first-order *optimal* asymptotic regret bound

uniform over bounded observations. However, due to the lack of regularization, their bound (*cf* Theorem 11 in (Gaillard et al., 2019)) may blow up if the design matrix $G_t(0)$ is not full rank. It is hence not uniform over all bounded feature sequences $\{x_t\}_t$.

Paper outline and contributions: In this paper, we continue the line of work initiated on the *forward* algorithm and advocate for its use in the stochastic setting with possibly unbounded response variables, in replacement for the ridge regression (whenever possible). To this end, we consider an online *stochastic* linear regression setup where the noise is assumed to be i.i.d σ-sub-Gaussian.

In Section 4.2 we recall the online performance bounds established for ridge regression and the forward algorithm in the *adversarial* case with bounded observations. Next, in subsection 4.2.3, we discuss some limitations of the adversarial results when comparing regression algorithms in the stochastic setting. For instance, these bounds compare the cumulative loss of a strategy to the value of the batch optimization problem, which may not be indicative of the real performance of the strategy (*cf* Corollary 4.5) and may encourage a sub-optimal tuning of the regularization parameter.

In Section 4.3, we study the performance of these algorithms using the cumulative regret with respect to the true parameter (*cf* Equation 4.6), which we believe is more practitioner-friendly than comparing to the batch optimization problem. We show in Theorem 4.6 how these two measures of performance are related. We provide in Theorems 4.7 and 4.9 a novel analysis of online regression algorithms without assuming bounded observations. This key result is made possible by considering high probability bounds instead of bounded individual sequences. We show that the regret upper-bound for ridge regression is inversely proportional to the regularization parameter. Consequently, we argue that following these results, forward regression should be used *in lieu* of ridge regression.

In Section 4.5, we revisit the linear bandit setup previously analyzed assuming *bounded* rewards: we relax this assumption and provide an -optimism in the face of uncertainty- style algorithm with the forward algorithm instead of ridge, which is especially well-suited for the bandit setup, and provide novel regret analysis in Theorem 4.15. We proceed similarly by revisiting a setup of non-stationary (abruptly changing) linear bandits.

4.2 Adversarial bounds and limitations

In this section, we recall existing results regarding the aforementioned ridge and forward algorithms. We then discuss their limits and benefits when considered from a stochastic perspective.

4.2.1 Adversarial regret bounds (known results)

One of the first theoretical analyses of online regression dates back to (Vovk, 2001) and (Azoury and Warmuth, 2001), and is recalled in the theorem below. It is stated in the form of an "online-to-offline conversion" performance bound.

Theorem 4.1. (*Theorem 4.6*[2] *of* (*Azoury and Warmuth, 2001*)) *The online ridge regression algorithm satisfies:*

$$L_T^r - \min_\theta \left(L_T(\theta) + \lambda \|\theta\|_2^2 \right) \leq 4(Y^r)^2 d \log \left(1 + \frac{TX^2}{\lambda d} \right),$$

where $X = \max_{1 \leq t \leq T} \|x_t\|_2$, *and* $Y^r = \max_{1 \leq t \leq T} \left\{ |y_t|, \left| \boldsymbol{x}_t^\top \boldsymbol{\theta}_{t-1} \right| \right\}$.

The reader should note that this result compares the learner's online cumulative loss to the regularized batch ridge regression loss. As such, it is an online-to-offline conversion regret. This is different from the sequential regret that would compare to the minimum achievable loss. This theorem highlights a dependence on the *range* of predictions of the algorithm, as $Y^r \geq \max_{1 \leq t \leq T} \left| \boldsymbol{x}_t^\top \boldsymbol{\theta}_{t-1} \right|$.

Remark 4.2. (*Small losses*) *The regret bound for ridge regression can be improved if the learner knows that the loss is small for the best expert, see Orabona, Cesa-Bianchi, and Gentile* (2012). *Note however that such techniques require prior knowledge of all the best expert loss* L_T^*, *their optimal bound is* $\sim O(\sqrt{L_T^* \log T})$.

The forward algorithm has an enhanced performance in this setup according to this next result.

Theorem 4.3. (*Theorem 5.6 of* (*Azoury and Warmuth, 2001*)) *The forward algorithm satisfies:*

$$L_T^f - \min_\theta \left(L_T(\theta) + \lambda \|\theta\|_2^2 \right) \leq (Y^f)^2 d \log \left(1 + \frac{TX^2}{\lambda d} \right),$$

where $X = \max_{1 \leq t \leq T} \|x_t\|_2$, *and* $Y^f = \max_{1 \leq t \leq T} |y_t|$.

Notice that in this result Y is different than in Theorem 4.1 and is independent from the algorithm's predictions. Moreover, Theorem 4.3 exhibits a bound that is at least 4 times better

than Theorem 4.1. More precisely, Theorem 4.1 suggests that, in order to compare the two bounds, prior knowledge of $Y^{\mathtt{f}}$ is required to further clip the predictions of online ridge regression in $[-Y^{\mathtt{f}}, Y^{\mathtt{f}}]$; and that even with such knowledge the forward algorithm may be 4-times better than ridge regression. We believe that this unfortunately led researchers to turn away from analyzing more deeply what may happen.

4.2.2 Limitation in the adversarial setup: rigid regularization

To evaluate online regression strategies, a tight *lower* bound was derived in (Gaillard et al., 2019). The latter studied uniform minimax lower bounds in the setting of beforehand-known features (that is when $(x_t)_{1\leq t\leq T}$ known in advance), which is very challenging for a lower bound. They show that, the minimax uniform regret bound is controlled as follows.

Theorem 4.4. (*Gaillard et al. (2019)*) *For all $T \geq 8, Y > 0$ we have:*

$$R^{\star}_{T,[-Y,Y]} \geq dY^2(\log(T) - (3 + \log(d)) - \log(\log(d)))\,.$$

where $R^{\star}_{T,[-Y,Y]} \overset{def}{=} \inf_{\mathcal{A}} \sup_{x_{1,\dots,T}\in[0,1]^d} \sup_{|y_t|\leq Y} \left\{ \sum_{t=1}^T (y_t - \hat{y}_t^{\mathcal{A}})^2 - \inf_{u\in\mathbb{R}^d} \sum_{t=1}^T (y_t - x_t^\top u)^2 \right\}$

We will use this result to evaluate the optimality of ridge and forward regressions. First, we need to convert Theorems 4.1 and 4.3 to sequential *regret* bounds. Indeed, in their current form, they compare the cumulative loss of the learner to the value of a regularized batch optimization. This next result transforms them, and is a corollary of Theorems 11.7 and 11.8 of Cesa-Bianchi and Lugosi (2006).

Corollary 4.5. (*Of Theorems 11.7 and 11.8 of (Cesa-Bianchi and Lugosi, 2006)*) *For all $T \geq 1, (x_t)_{1\leq t\leq T} \in \mathbb{R}^d, (y_t)_{1\leq t\leq T} \in [-Y, Y]$ such that $\|x_t\|_2 \leq X$,*

$$\text{for } \mathcal{A} \in \{\mathtt{r}, \mathtt{f}\} \qquad R_T^{\mathcal{A}} \leq c^{\mathcal{A}}(Y^{\mathcal{A}})^2 d \log\left(1 + \frac{TX^2}{\lambda d}\right) + \frac{\lambda\,(Y^{\mathcal{A}})^2 T}{\lambda_{r_T}(G_T(0))},$$

where $r_T = \operatorname{rank}(G_T(0))$ and λ_{r_T} is its smallest positive eigenvalue, $c^{\mathtt{r}} = 4$ and $c^{\mathtt{f}} = 1$.

This bound suggests that to obtain a $\log(T)$ bound, λ should not be chosen larger than about $\log(T)/T$, due to the second term, this is the *stringent regularization limitation*.

Proof. Consider (w.l.o.g) ridge regression, denote X_T the design matrix and y_T the labels, then:

$$\|\theta_T\|_2 = \left\|G_T(0)^\dagger \mathbf{b}_T\right\|_2 = \sqrt{y_T^\top X_T^\top G_T(0)^\dagger G_T(0)^\dagger X_T y_T} \leq \sqrt{\frac{y_T^\top X_T^\top G_T(0)^\dagger X_T y_T}{\lambda_{r_T}(G_T)}}$$
$$\leq Y^{\mathtt{r}} \sqrt{\frac{T}{\lambda_{r_T}(G_T)}},$$

where $G_T(0)^\dagger$ is the pseudo-inverse of $G_T(0)$, the last inequality is because $X_T^\top G_T(0)^\dagger X_T$ is an orthogonal projection on $\mathrm{Im}(X^\top)$. Injecting in the previous theorem finishes the proof, these bounds hold for arbitrary bounded sequences. The proof for the forward algorithm proceeds in the same way by replacing G_T by G_{T+1} and $Y^{\mathtt{r}}$ by $Y^{\mathtt{f}}$. □

Choosing $\lambda = 1/T$ yields a first order regret of $2dY^2$ for the forward algorithm and $8dY^2$ for ridge regression (with clipping and prior knowledge of Y), which is at best twice the first order term from the lower bound. This suggests the presence of an optimality gap. Strikingly, Gaillard et al. (2019) shows that a non-regularized version of the forward algorithm achieves the optimal first order of dY^2. However, it also suffers from an important weakness: Indeed, the $(Y^{\mathcal{A}})^2/\lambda_{r_T}(G_T(0))$ term in Corollary 4.5 is not uniformly bounded over feature sequences, but only on specific "well-behaved" features. In fact, double uniformity over features and observations is still an open question (see (Gaillard et al., 2019)).

4.2.3 Limitations in the stochastic setting

Now that we have recalled the main properties of the forward and ridge algorithms in the adversarial setup, we advocate for the need of a complementary analysis of the previous algorithms in the stochastic unbounded setting by unveiling some key limitations.

Too unconstrained The existing analysis being for a different setting, it naturally ignores crucial aspects of the stochastic setup. For instance, the quantity Y is uninformative and may be substantial. Let us look at how the term Y appears in the proofs of Azoury and Warmuth (2001). For ridge regression, the penultimate step to prove Theorem 4.1 writes:

$$L_T^{\mathtt{r}} - \min_\theta \left(L_T(\theta) + \lambda\|\theta\|_2^2\right) \leq \sum_{t=1}^T \underbrace{\left(x_t^\top \theta_{t-1} - y_t\right)^2 x_t^\top G_t^{-1} x_t}_{\text{first term}} \leq 4(Y^{\mathtt{r}})^2 \sum_{t=1}^T x_t^\top G_t^{-1} x_t. \tag{4.4}$$

In an adversarial setting, the "first term" cannot be controlled without assuming bounded predictions $|x_t^\top \theta_{t-1}| \leq Y^{\mathtt{r}}$, and doing so yields a bound $\left(x_t^\top \theta_{t-1} - y_t\right)^2 \leq 4(Y^{\mathtt{r}})^2$. In a stochastic setup however, we expect the term $\left(x_t^\top \theta_{t-1} - y_t\right)^2$ to reduce and stabilize around $\left(x_t^\top \theta_* - y_t\right)^2$, owing to the convergence properties of the estimate towards θ_*.

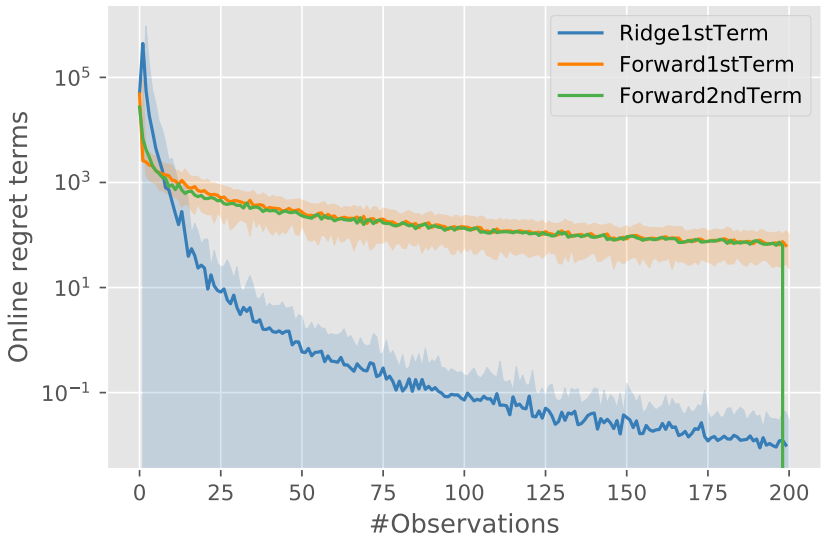


Figure 4.1 – Online regret. y-axis is logarithmic.

For the forward algorithm, the final step in the proof of Theorem 4.3 writes:

$$L_T^{\mathtt{f}} - \min_{\theta}\left(L_T(\theta) + \lambda\|\theta\|_2^2\right) \leq \sum_{t=1}^{T} \underbrace{y_t^2 x_t^\top G_t^{-1} x_t}_{\text{first term}} - \sum_{t=1}^{T-1} \underbrace{\boldsymbol{x}_{t+1}^\top \boldsymbol{G}_t^{-1} \boldsymbol{x}_{t+1} \left(\boldsymbol{x}_{t+1}^\top \boldsymbol{\theta}_t\right)^2}_{\text{second term}}. \tag{4.5}$$

Then, the analysis uses that $|y_t| \leq Y^{\mathtt{f}}$ and disregards the negative contribution of the "second term". *Illustrative example:* Let us analyze these terms in an practice: consider $d = 5$, $\theta_* \in \mathbb{R}^5$, we sample 200 features uniformly in $[0,1]^5$ and Gaussian noises ($\sigma = 0.1$). Figure 4.1 displays the instantaneous first regret term of both algorithms (with $\lambda = 1$) and the second regret term of the forward algorithm, averaged over 100 replicates. We remark that the first terms vanish quickly for ridge regression and are quite stable for the forward algorithm. On the other hand, they are essentially cancelled out by the second term. Overall, the two strategies perform on par on this example. This suggests that Theorems 4.1 and 4.3 can be misleading in this stochastic setup: for ridge regression they introduce a conservative $4(Y^{\mathtt{r}})^2$ bound on $(x_t^\top(\theta_{t-1} - \theta_*))^2$, while in practice we observe that this term decreases rapidly to zero; for the forward algorithm, the bound ignores the effect of a negative term, which, as we see in Figure 4.1, is essential to explain why this algorithm may outperform ridge regression.

Time dependence: In the stochastic case, it can be confusing to introduce $Y^{\mathcal{A}}$. Indeed, the latter hides a significant implicit time-dependence. For instance, for the forward algorithm $Y^{\mathtt{f}} = \max_{1\leq t\leq T} |x_t^\top \theta_* + \varepsilon_t|$. Considering the tractable setting of Gaussian i.i.d noise with variance σ^2 and using the classical Sudakov minoration (Sudakov, 1969), we deduce that there

exists $C > 0$ such that:

$$\forall T \geq 1 : \mathbb{E}[Y^{\mathtt{f}}] \geq \mathbb{E}\left[\max_{1\leq t\leq T} \varepsilon_t\right] - X\|\theta_*\|_2 \geq \sigma C\sqrt{2\log(T)} - X\|\theta_*\|_2.$$

Since $(Y^{\mathtt{f}})^2$ appears in the previous performance bounds, this suggests that $Y^{\mathtt{f}}$ actually increases the order of the regret bound to $\log(T)^2$ in this setting.

By focusing on the *unbounded stochastic* scenario, we hope in this paper to shed novel light on the practical performance of these strategies and better explain these phenomena.

4.3 High probability bounds

In this section, we analyze online ridge regression and the forward algorithm in the *stochastic* setting. We present our results in terms of the following intuitive regret definition:

$$\bar{R}_T^{\mathcal{A}} = L_T^{\mathcal{A}} - L_T(\theta_*). \tag{4.6}$$

This regret directly compares the cumulative loss of the learner to the cumulative loss of the oracle knowing the true parameter θ_*. This contrasts with the online-to-batch conversion result that compares the loss of the learner to the value of a batch regularized optimization problem. Since we are in a stochastic setup, we further state results in high probability. More precisely, we state Theorems 4.6,4.7, 4.9 below holding with high probability uniformly over all T, and not simply for each T. As a first step, we prove that for T great enough, we can choose this definition instead of R_T defined in Equation 4.1 without altering the bounds.

Theorem 4.6. (*Regret equivalence*) *In the stochastic setting with sub-Gaussian noise, for all $\delta > 0$ with probability at least $1 - \delta$, for all $T > 0, (x_t)_{1\leq t\leq T} \in \mathbb{R}^d$ such that $\|x_t\|_2 \leq X, |G_T(0)| > 0$*

$$L_T(\theta_*) - \min_{\theta\in\mathbb{R}^d} L_T(\theta) = o\left(\log(T)^2\right),$$

in particular, it comes

$$R_T^{\mathcal{A}} = \bar{R}_T^{\mathcal{A}} + o\left(\log(T)^2\right)$$

Theorem 4.6 justifies choosing $\bar{R}_T$ to provide identical first order guarantees as R_T. Indeed, in the following sections we prove high probability upper bounds of order $O(\log(T)^2)$.

Proof. Denote $\forall T \geq 1 : \theta_T = \arg\min_{\theta \in \mathbb{R}^d} L_T(\theta)$, then:

$$R_T - \bar{R}_T = L_T(\theta_*) - L_T(\theta_T) = 2\sum_{t=1}^{T} \varepsilon_t (\theta_T - \theta_*)^\top x_t - \sum_{t=1}^{T} \left((\theta_T - \theta_*)^\top x_t\right)^2. \tag{4.7}$$

Denote $S_T = \sum_{t=1}^{T} \varepsilon_t (\theta_T - \theta_*)^\top x_t$ and $A_T = \sum_{t=1}^{T} \left((\theta_T - \theta_*)^\top x_t\right)^2$. Using Lemma B.1 we can prove that $S_T = o(A_T)$, this means that we can focus on bounding A_T to obtain the desired result.

Using Lemma B.1 and Equation (4.7) we get that for all $\sigma', \delta > 0$ with probability at least $1 - \delta$:

$$\begin{aligned} R_T - \bar{R}_T &\leq \sqrt{2(A_T + 1/\sigma'^2) \log\left(\sqrt{\sigma'^2 A_T + 1}/\delta\right)} - A_T \\ &\leq \sqrt{(A_T + 1/\sigma'^2)\left(\log(\sigma'^2 A_T + 1) + 2\log(1/\delta)\right)} - A_T \\ &\leq \sqrt{A_T + 1/\sigma'^2}\left(\sqrt{\log(\sigma'^2 A_T + 1)} + \sqrt{2\log(1/\delta)}\right) - A_T \\ &\leq \frac{1}{\sigma'^2} + \sqrt{2(A_T + 1/\sigma'^2)\log(1/\delta)} \end{aligned} \tag{4.8}$$

The next step is to the use confidence intervals of Maillard (2016) (Theorem 3.3 therein) which hold once the design matrix is singular, see Theorem B.1 for its statement.

The latter entails, that for bounded features $\|x\| \leq X$, we obtain $\lambda_{max}(G_T(0)) \leq TX^2$. Denote $T_0 = \inf_{t \geq 1}\{|G_t| > 0\}$, and for $t \geq T_0 : \beta_t = 2(1+\kappa)(1+\alpha)\sigma^2 \log \frac{\kappa_d(e^2 \lambda_{max}(G_T))}{\delta}$, then for all $\delta > 0$ with probability at least $1 - \delta$:

$$\begin{aligned} A_T &= \sum_{t=1}^{T} \left((\theta_T - \theta_*)^\top x_t\right)^2 \leq A_{T_0} + \sum_{t=T_0}^{T} \left((\theta_T - \theta_*)^\top x_t\right)^2 \\ &\leq A_{T_0} + \sum_{t=T_0}^{T} \beta_t \|x_t\|^2_{G_t(0)^{-1}} \leq A_{T_0} + \beta_T \sum_{t=T_0}^{T} \|x_t\|^2_{G_t(0)^{-1}} \end{aligned} \tag{4.9}$$

Then we bound the sum of features using Lemma B.2.

From equation (4.8), using $A_T \leq \sum_{t=1}^{T} \|\theta_T - \theta_*\|^2_{G_t} \|x_t\|^2_{G_t^{-1}} \leq \sum_{t=1}^{T} \|\theta_T - \theta_*\|^2_{G_T} \|x_t\|^2_{G_t^{-1}}$, then injecting Lemma B.2 with $\lambda = 0$, we find that for all $\delta > 0$, with probability at least $1 - \delta$:

$$A_T \leq A_{T_0} + \beta_T d \log\left(1 + TX^2/\lambda_{min}(G_{T_0}(0))d\right)$$

Then injecting this last inequality in equation (4.7) gives, for all $\delta, \sigma' > 0$, with probability at least $1-\delta$:

$$R_T - \bar{R}_T \leq \frac{1}{\sigma'^2} + \sigma' \sqrt{2\log(1/\delta)\left(\beta_T d \log\left(1 + TX^2/\lambda_{min}(G_{T_0})d\right) + 1\right)}.$$

Finally, we know -by definition- that $R_T \geq \bar{R}_T$, this concludes the proof for the equivalence of the two regret definitions. □

4.3.1 Online ridge regression

We start our results by stating a new high probability regret bound for online ridge regression.

Theorem 4.7. *In the stochastic setting with sub-Gaussian noise, for all $\delta > 0$ with probability at least $1-\delta$, for all $T \geq 0$:*

$$\bar{R}_T^r \leq \frac{2d\sigma^2 X^2}{\lambda \log(1 + X^2/\lambda)} \log\left(1 + TX^2/\lambda d\right) \log\left(\frac{(1 + TX^2/\lambda d)^{d/2}}{\delta/2}\right) + o(\log(T)^2),$$

where $X = \max_{1 \leq t \leq T} \|x_t\|_2$.

This result is interesting because the *ranges* of both predictions and observations do not appear, hence predictions clipping and/or a prior knowledge assumption on $Y^{\mathfrak{f}}$ are not required. On the other hand, a factor $1/\lambda$ appears in the worst case of a singular design matrix. This seems to be the price for no longer assuming bounded predictions. Another notable improvement is that this bound no longer involves $\lambda\|\theta\|_2^2$ terms. In particular, it is uniform over bounded sequences of observations.

Proof. See Equation 4.12 for an explicit bound. In particular, the $o(\log(T)^2)$ term is $O(\log(T)^{3/2})$.

Let's write the instantaneous regret:

$$\bar{r}_t = \ell_t(\theta_{t-1}) - \ell_t(\theta_*) = (\theta_{t-1}^\top x_t - \theta_*^\top x_t)^2 + 2\varepsilon_t(\theta_{t-1}^\top x_t - \theta_*^\top x_t) \tag{4.10}$$

The proof proceeds in three steps, that we detail hereafter and then we explain how to combine them for the final result.

First step: Confidence bound to control the concentration of θ_{t-1} around θ_*. For this we use the confidence ellipsoid from Abbasi-Yadkori, Pál, and Szepesvári (2011), which states that for

any $\delta > 0$, with probability at least $1 - \delta$, for all $t > 0$:

$$\|\theta_t^{\mathrm{r}} - \theta_*\|_{G_t} \leq \sqrt{\beta_t(\delta)} = \sigma\sqrt{d\log\left(\frac{1 + tX^2/\lambda d}{\delta}\right)} + \lambda^{1/2}S.$$

It comes, with probability at least $1 - \delta$:

$$(\theta_{t-1} - \theta_*)^\top x_t \leq ||x_t||_{G_{t-1}^{-1}} ||\theta_{t-1} - \theta_*||_{G_{t-1}} \leq \sqrt{\beta_{t-1}(\delta)}||x_t||_{G_{t-1}^{-1}}.$$

Then, since β_t is non-decreasing:

$$L_t - L_t^* \leq \beta_{T-1}\sum_{t=1}^{T}\|x_t\|^2_{\eta_{t-1}} + 2\sum_{t=1}^{T}\varepsilon_t(\theta_{t-1} - \theta_*)^\top x_t. \tag{4.11}$$

Second step: Next we bound the sum of feature norms. The main idea here is to use linear algebra techniques to obtain a telescopic sum.
Lemma B.2 doesn't apply here because we have $||x_t||_{G_{t-1}^{-1}}$ instead of $||x_t||_{G_t^{-1}}$. We derive a similar result for this sum of feature norms in Lemma B.3.

Third step: To control the second term in the r.h.s of Equation 4.10, we use Martingale inequalities similar to the ones used for the confidence intervals to derive a uniform high probability bound. Indeed, Lemma B.4 proves that the second term in Equation 4.10 is a of order $\sim O(\log(T)\log\log T)$. In fact, with high probability $\sum_{s=1}^{t}\left((\theta_{t-1} - \theta_*)^\top x_t\right)^2 = O(\log(T)^2)$ therefore, with high probability S_T is of order $\sim O(\log(T)\log(\log T)/\delta)$. Consequently, with high probability, S_T is second order.

Proof aggregation: By combining earlier results we find for any $\delta, \sigma' > 0$, with probability at least $1 - \delta$, for all $T \geq 0$:

$$\bar{R}_T^{\mathrm{r}} \leq \left(\sigma\sqrt{d\log\left(\frac{1 + TX^2/\lambda d}{\delta/2}\right)} + \lambda^{1/2}S\right)^2 \frac{X^2/\lambda}{\log(1 + X^2/\lambda)} d\log\left(1 + TX^2/\lambda d\right)$$
$$+ \sigma\sqrt{2\left(1/\sigma'^2 + \sum_{s=1}^{t}\left((\theta_{s-1} - \theta_*)^\top x_s\right)^2\right)\log\left(2\sqrt{1 + \sigma'^2\sum_{s=1}^{t}((\theta_{s-1} - \theta_*)^\top x_s)^2}/\delta\right)}. \tag{4.12}$$

□

Remark 4.8. (*Regularization in ridge*) *Note that the bound holds with high probability, uniformly over T, and not only for each individual time horizon. In the proof of this result, $1/\lambda$ emerges from bounding $\lambda_{min}(G_t(0))$ in the worst case. When the collected features ensure the design matrix $G_t(0)$ is invertible, $1/\lambda$ virtually disappears. We highlight this experimentally in Section 4.4.2.*

4.3.2 The forward algorithm

We analyze the forward algorithm and derive a high probability regret bound for it using similar techniques up to minor modifications.

Theorem 4.9. *Assuming sub-Gaussian noise, with probability at least* $1-\delta$*, for all* $T \geq 0$*:*

$$\bar{R}_T^f \leq 2d\sigma^2 \log\left(1 + TX^2/\lambda d\right) \log\left(\frac{(1+TX^2/\lambda d)^{d/2}}{\delta/2}\right) + o(\log(T)^2),$$

where $X = \max_{1\leq t\leq T} \|x_t\|_2$ *and the* $o(\log(T)^2)$ *depends on* λ *(see Equation 4.13).*

Proof. The proof proceeds similarly to that of Theorem 4.7. First, we need to bound the instantaneous regret.

$$\bar{r}_t = \ell_t(\theta_{t-1}) - \ell_t(\theta_*) = (\theta_{t-1}^\top x_t - \theta_*^\top x_t)^2 + 2\varepsilon_t(\theta_{t-1}^\top x_t - \theta_*^\top x_t)$$

We proceed in three steps like before.

First step: We start by deriving a confidence ellipsoid for this new parameter estimate. This is the novel result of Theorem B.5 where we show that for any $\delta > 0$, with probability at least $1-\delta$, for all$t > 0$:

$$\|\theta_t - \theta_*\|_{G_t} \leq \sqrt{\beta_t(\delta)} = \sigma\sqrt{d\log\left(\frac{1+tX^2/\lambda d}{\delta}\right)} + (\lambda^{1/2} + X)S.$$

For the first term, with probability at least $1-\delta$ for all $t \geq 0$:

$$\begin{aligned}(\theta_{t-1} - \theta_*)^\top x_t &\leq ||x_t||_{G_t^{-1}} ||\theta_{t-1} - \theta_*||_{G_t} \\ &\leq \sqrt{\beta_{t-1}(\delta)}||x_t||_{G_t^{-1}} \leq \sqrt{\beta_{T-1}(\delta)}||x_t||_{G_t^{-1}}.\end{aligned}$$

Second step: We can use Lemma B.2 to bound the sum of feature norms. It comes

$$\sum_{t=1}^{T}(\theta_{t-1}^\top x_t - \theta_*^\top x_t)^2 \leq \beta_T(\delta) d\log\left(1 + TX^2/\lambda d\right)$$

Third step: Again, we derive Lemma B.6, a high probability bound To control the second term in the r.h.s of (4.10).

Proof aggregation: We combine previous results to finish the proof of the forward algorithm regret bound. For any $\delta, \sigma' > 0$, with probability at least $1-\delta$, for all $T \geq 0$:

$$\bar{R}_T^{\mathbf{f}} \leq \left(\sigma\sqrt{d\log\left(\frac{1+TX^2/\lambda d}{\delta/2}\right)} + (\sqrt{\lambda}+X)S\right)^2 \frac{X^2/\lambda}{\log(1+X^2/\lambda)} d\log\left(1+TX^2/\lambda d\right)$$
$$+ \sigma\sqrt{2\left(1/\sigma'^2 + \sum_{s=1}^{t}\left((\theta_{t-1}-\theta_*)^\top x_t\right)^2\right)\log\left(\sqrt{1+\sigma'^2\sum_{s=1}^{t}((\theta_{t-1}-\theta_*)^\top x_t)^2}/\delta\right)}. \quad (4.13)$$

□

Theorem 4.9 exhibits a better bound than Theorem 4.7. In fact, the coefficient of the first order term for the forward algorithm only depends on the dimensionality and the noise variance, whilst for ridge regression, it also depends on the features' scale and on the regularization parameter λ.

Remark 4.10. (*Unrestrained regularization*) *Compared to existing results, this analysis lifts the "stringent regularization" that requires $\lambda = 1/T$ or data-dependent regularization* (cf (*Malek and Bartlett, 2018*)) *to obtain uniform bounds. Therefore, Theorems 4.7 and 4.9 are not a mere consequence of bounding Y^2 with high probability in previous deterministic theorems. For completeness, we also derive a high probability regret bound for a non-regularized version of the forward algorithm in Appendix 4.4; this algorithm was proven to be asymptotically first order minimax optimal in the adversarial bounded setting* (*Gaillard et al., 2019*).

4.4 The unregularized-forward algorithm

For the sake of completeness, we propose a high probability bound on the regret of a non-regularized forward algorithm -studied in the adversarial bounded case in Gaillard et al. (2019)- which achieves the optimal asymptotic first order deterministic minimax bound of $dY^2\log(T)$. This algorithm is a simple yet elegant modification of forward regression, it avoids the exploding $\lambda\|\theta_T\|_2^2$ term by setting $\lambda = 0$. Consequently $\theta_t = G_{t+1}^\dagger b_t$, where $G_t^\dagger$ is the pseudo-inverse of G_t.

Theorem 4.11. (*Regret of the unregularized forward*) *The unregularized forward regression achieves, for any $\delta > 0$, with probability at least $1-\delta$ for all $T > 0$:*

$$\bar{R}_T^{u-f} \leq 2(1+\kappa)(1+\alpha)\sigma^2\log\left(\frac{\kappa_d(1+TX^2/\gamma d)}{\delta/4}\right)\log\left(\frac{|G_T^\dagger|}{|G_{T_1}^\dagger|}\right)$$
$$+ 2\sigma^2\log\left(\frac{4T_1}{\delta}\right)\left(d + \sum_{1\leq t\leq T_1, t\in\mathcal{T}}\log\left(\frac{X^2}{\lambda_{r_t}(\sum_{s=1}^{t}x_t x_t^\top)}\right)\right),$$

where $\kappa, \alpha \in \mathbb{R}_+^*$ *are peeling parameters (can be chosen),* $\gamma = \min_{1\leq t\leq T} \|x_t\|_2$*, and* $\kappa_d(x) \propto x^d$ *up to logarithmic factors and depends on* κ *and* α (cf *Theorem 5.4 in Maillard (2016)*)*.* $T_1 = \min\{t \geq 1, |G_t| > 0\}$ *is, if it exists, the first time the design matrix is non-singular, otherwise* $T_1 = T$*, and* $\mathcal{T}$ *is the set of indices* t *such that rank*(G_t) *>rank*(G_{t-1})*. The last term accounts for when the design matrix is singular, and is naturally unbounded (this was also the case in the adversarial case).*

Asymptotically, with probability at least $1-\delta$ the first regret term is bounded as:

$$\bar{R}_T^{u-f} \leq 2(1+\kappa)(1+\alpha)\log\left(\frac{C(\kappa,\alpha)(TX^2/\lambda d)^d}{\delta}\right)\log\left((T-T_1)X^2/\lambda d\right),$$

where $C(\kappa, \alpha)$ is a function of the peeling parameters.
We don't seek a more involved analysis to explicit this bound or improve on it, but we see that vaguely it leads to a bound similar to Theorems 4.7 and 4.9 provided that the term accounting for the singularity of the design matrix is controlled. The latter empowers the intuition that in the high probability analysis, the forward algorithm is *first order minimax optimal* even though concretely we cant be sure because we don't have access to uniform lower bounds.

Proof. The proof consists of two mains steps: the first is to use the following bound while the design matrix is singular:

Theorem 4.12. (*Theorem 11 Gaillard et al. (2019)*) *For all* $T \geqslant 1$*, for all sequences* $x_1, \ldots, x_T \in \mathbb{R}^d$ *and all* $y_1, \ldots, y_T \in [-Y, Y]$*, the unregularized forward algorithm achieves the regret bound*

$$R_T(\mathbf{u}) \leq Y^2 \sum_{t=1}^{T} \mathbf{x}_t^{\mathrm{T}} \eta_t^{\dagger} \mathbf{x}_t \leqslant dY^2 \log T + dY^2 + Y^2 \sum_{t\in[1,T]\cap\mathcal{T}} \log\left(\frac{X^2}{\lambda_{r_t}(\sum_{s=1}^{t} x_s x_s^{\top})}\right)$$

where $\forall M \in \mathcal{M}_d(\mathbb{R}), \lambda_1(M) \geq \ldots \geq \lambda_d$ *are* M*'s eigenvalues and* $r_t = \mathrm{rank}(\sum_{s=1}^{t} x_s x_s^{\top})$*) and where the set* $\mathcal{T}$ *contains* r_T *rounds, given by the smallest* $s \geqslant 1$ *such that* x_s *is not* mull*, and all the* $s \geqslant 2$ *for which rank* $(\mathrm{G}_{s-1}) \neq \mathrm{rank}\,(\mathrm{G}_s)$*.*

The second step is a bound when the design matrix is invertible, using Theorem B.1. Denote $T_1 = \inf_{t\geq 1}\{|G_t| > 0\}$, using Theorem 4.12:

$$\bar{R}_{T_1} \leq Y^2\left(d\log(T_1) + d + \sum_{1\leq t\leq T_1, t\in\mathcal{T}} \log\left(\frac{X^2}{\lambda_{r_t}(G_t)}\right)\right)$$

From standard results on sub-Gaussian noise, we also know that $\mathbb{E}[\max_{1\leq t\leq T}\varepsilon_t] \leq \sigma\sqrt{2\log(T)}$ (see *e.g.* Kamath (2015)), then using the transformation of Laplace along with Markov's

inequality, $\forall \delta > 0$ P $(\forall T \geq 1, Y^2 \leq 2\sigma^2 \log(T/\delta)) \geq 1 - \delta$, hence with probability at least $1 - \delta$:

$$\bar{R}_{T_1} \leq 2d\sigma^2 \log \frac{T_1}{\delta} \log(T_1) + 2d\sigma^2 \log \frac{T_1}{\delta} + 2\sigma^2 \frac{\log T_1}{\delta} \sum_{1 \leq t \leq T_1, t \in \mathcal{T}} \log \left(\frac{X^2}{\lambda_{r_t}(\sum_{s=1}^{t} x_s x_s^\top)} \right). \tag{4.14}$$

And for $T > T_1$, we bound $R_T - R_{T_1}$ using the same methodology as the proofs of Theorem 4.7 and Theorem 4.9. In fact, using the confidence bounds of Theorem B.1 we find, for all $\delta > 0$, with probability at least $1 - \delta$:

$$\forall t > T_1 : (\theta_{t-1}^\top x_t - \theta_*^\top x_t)^2 \leq \sqrt{\beta_{t-1}(\delta)} ||x_t||_{G_t^\dagger}.$$

We use the tail inequality of Lemma (B.4) to get, $\forall \delta > 0$, with probability at least $1 - \delta, \forall T > 0$:

$$\bar{R}_T - \bar{R}_{T_1} \leq 2(1+\kappa)(1+\alpha)\sigma^2 \log \left(\frac{\kappa_d(1 + TX^2/\lambda d)}{\delta/2} \right) \log \left(\frac{|G_T^\dagger|}{|G_{T_1}^\dagger|} \right) \tag{4.15}$$

From (4.14) and (4.15) we obtain for all $\delta > 0$, with probability at least $1 - \delta$:

$$\bar{R}_T \lesssim 2(1+\kappa)(1+\alpha)\sigma^2 \log \left(\frac{\kappa_d(1 + TX^2/\lambda d)}{\delta/4} \right) \log \left(\frac{|G_T^\dagger|}{|G_{T_1}^\dagger|} \right) + 2\sigma^2 \frac{\log(T_1)}{\delta/4} \left(d + \sum_{1 \leq t \leq T_1, t \in \mathcal{T}} \log \left(\frac{X^2}{\lambda_{r_t}(\sum_{s=1}^{t} x_t x_t^\top)} \right) \right).$$

□

4.4.1 Tightness of the bounds

Here we clarify the impact of a tighter confidence width for regularized least squares that was proved concurrently with the writing of this paper. First we state the result then we discuss its implications.

Theorem 4.13. (*Theorem 1 of Tirinzoni, Pirotta, et al. (2020)*) *Let $\delta \in (0,1), n \geq 3$, and $\widehat{\theta}_t$ be a regularized least-square estimator obtained using $t \in [n]$ samples collected using an arbitrary bandit strategy $\pi := \{\pi_t\}_{t \geq 1}$. Then,*

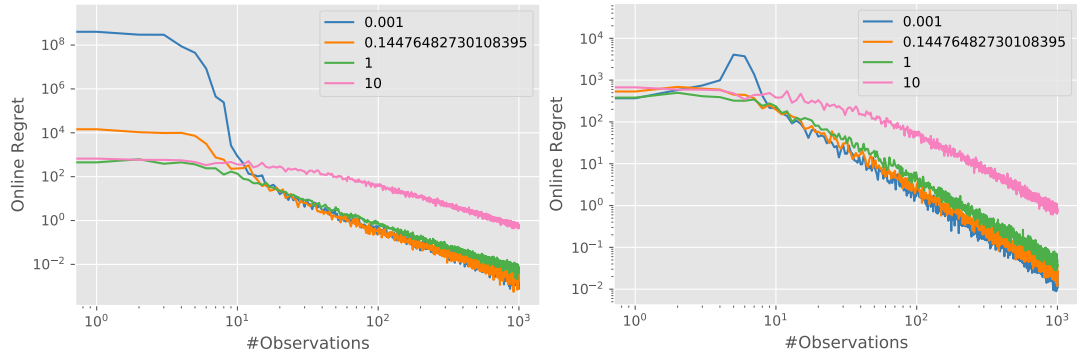
$$\mathbb{P}\left\{ \exists t \in [n] : \left\| \widehat{\theta}_t - \theta_* \right\|_{\bar{V}_t} \geq \sqrt{c_{n,\delta}} \right\} \leq \delta$$

where $c_{n,\delta}$ is of order $\mathcal{O}(\log(1/\delta) + d\log\log n)$.

This has important implications for Theorems 4.6, 4.7 and 4.9: in short it re-scales their regret upper-bounds from $R_T = O\left((d\sigma)^2 \log(T)^2\right)$ to $R_T = O\left(d\sigma^2 \log(T)\log(1/\delta)\right)$. The first order $(d\sigma)^2 \log(T)^2$ in our results is the product of **1)** $d\log T$ from the elliptical lemma, for bounding the sum of feature norms and **2)** $\sigma^2 \log(T^d/\delta)$ the confidence ellipsoid width in the estimation of the regression parameter. It is the second term that is altered following the new result from Tirinzoni, Pirotta, et al. (2020). These tighter confidence intervals change the upper bounds to $O(d\sigma^2 \log(T)\log\log(T))$. The latter matches the popular lower bounds in excess risk literature (see *e.g.* Theorem 1 in Mourtada (2019)) up to sub-logarithmic terms suggesting *the optimality of the forward algorithm in the stochastic setting*.

4.4.2 Experiment

We provide experimental evidence supporting the fact that our novel high probability analysis better reflects the influence of regularization than results its adversarial counterpart.



(a) Ridge regression **(b)** The forward algorithm

Figure 4.2 – Online regret's (Instantaneous loss difference) dependence on λ. All axes are logarithmic. Lines are averages over 100 repetitions and shaded areas represent one standard deviation.

In Figures 4.2a and 4.2b we observe the effect of regularization on the performance of ridge regressions and on the Forward algorithm in a 5-dimensional regression setting. We vary $\lambda \in \{1/T, 1/\log(T), 1, 10\}$, sample a zero mean Gaussian noise with $\sigma = 0.1$ and draw features uniformly from the unit ball. The results clearly highlight the robustness of the forward algorithm to λ, contrarily to ridge. In particular, for ridge regression, we observe the exact dependence on λ described by Theorem 4.7 in the first rounds of learning; as explained in Remark 4.8, once the collected features are enough for the design matrix $G_t(0)$ to become non-singular, the $1/\lambda$ virtually disappears from the first order regret bound and is replaced by the smallest eigenvalue of $G_t(0)$, making the regret significantly more stable.

4.5 Application: linear bandits

The proposed analysis of forward regression in the stochastic setting suggests that using it could be useful for revisiting several popular setups that include linear function approximation. We apply this change for stochastic linear bandits hereafter and derive the novel regret bound obtained when using forward regression instead of the standard ridge regression.

4.5.1 Stationary bandits

Consider the setting of *stochastic linear bandits*, where at round t the reward of an action x_t (from the action space $\mathcal{X} \subset \mathbb{R}^d$) is $y_t = \langle x_t, \theta_* \rangle + \varepsilon_t$, where $\theta_* \in \mathbb{R}^d$ is an unknown parameter and ε_t is, conditionally on the past, a σ-sub-Gaussian noise. An upper bound S on the unknown parameter's norm is provided: $\|\theta_*\|_2 \leq S$. The (pseudo) regret in this setting is defined:

$$R_T = \sum_{t=1}^{T} \langle x_t^*, \theta_* \rangle - \sum_{t=1}^{T} \langle x_t, \theta_* \rangle = \sum_{t=1}^{T} \langle x_t^* - x_t, \theta_* \rangle, \tag{4.16}$$

where $x_t^* = \arg\max_{x \in \mathcal{X}} \langle x, \theta_* \rangle$. Traditionally, the following additional assumption is made.

Assumption 4.14. *for all* $x_t \in \mathcal{X} \quad \langle x_t, \theta_* \rangle \in [-1, 1]$.

The "optimism in the face of uncertainty linear bandit" (OFUL) algorithm was introduced in (Abbasi-Yadkori, Pál, and Szepesvári, 2011). OFUL resorts to ridge regression, constructs a confidence ellipsoid for the parameter estimate, and chooses the action that maximizes the upper-confidence bound on the reward. Under Assumption 4.14, (Abbasi-Yadkori, Pál, and Szepesvári, 2011) we prove that the cumulative regret of OFUL satisfies, for $\delta > 0$ with probability at least $1 - \delta$, $\forall T > 0 \quad R_T^{\mathtt{r}} \leq 4\sqrt{Td \log(\lambda + TX^2/d)} \left(\lambda^{1/2} S + \sigma \sqrt{2 \log(1/\delta) + d \log(1 + TX^2/(\lambda d))} \right)$, where $X = \max_{1 \leq t \leq T} \|x_t\|_2$.

Forward variant [Algorithm 4.3]: In a second phase, we propose the variant OFUL$^{\mathtt{f}}$ in which we replace ridge regression by the forward algorithm. What this means is that the parameter estimate is a function of actions:

$$\theta_t^{\mathtt{f}}(x) = \arg\min_{\theta \in \mathbb{R}^d} \sum_{s=1}^{t} (y_s - \langle x_s, \theta \rangle)^2 + \lambda \|\theta\|_2^2 + \langle x, \theta \rangle^2.$$

This fits perfectly since actions can be chosen. Implementation details are in Algorithm 4.3.

Algorithm 4.3: OFUL$^{\mathtt{f}}$ algorithm

Given $\lambda, \delta, S > 0$
for $t = 1, \dots, T$ **do**
  $x_t = \arg\max_{x \in \mathcal{X}} \langle x, \theta_t^{\mathtt{f}}(x) \rangle + \|x\|_{G_{t-1,x}^{-1}} (\sqrt{\lambda} + \|x\|_2) S + \sigma \sqrt{2 \log\left(\frac{(1+tX_t^2(x)/\lambda d)^{d/2}}{\delta} \right)}$,
  where $X_t(x) = \max\{\|x\|_2, \max_{1 \le s \le t-1} \|x_s\|_2\}$, $G_{t-1,x} = G_{t-1} + xx^\top$ and $\theta_t^{\mathtt{f}}(x) = \arg\min_{\theta \in \mathbb{R}^d} \sum_{s=1}^{t-1} (y_s - \langle x_s, \theta \rangle)^2 + \lambda \|\theta\|_2^2 + \langle x, \theta \rangle^2$
  play x_t and observe y_t.
end

Note that OFUL$^{\mathtt{f}}$ only requires an upper bound S on $\|\theta_*\|_2$. We prove that OFUL$^{\mathtt{f}}$ enjoys the same regret bound as OFUL and doesn't require Assumption 4.14. In stark contrast, we cannot show a similar bound for the standard OFUL without said assumption, it actually suffers a λ-dependent scaling factor in this case.

Theorem 4.15. (*Bandits with unbounded rewards*) *Without Assumption 4.14, for all $\delta > 0$, OFUL^r achieves with probability at least $1 - \delta$, for all $T \ge 1$,*

$$R_T^r \le 4 \sqrt{\frac{\boldsymbol{X}^2 T d \log(\lambda + TX^2/d)}{\boldsymbol{\lambda} \log(1 + \boldsymbol{X}^2/\boldsymbol{\lambda})}} \left(\lambda^{1/2} S + \sigma \sqrt{2 \log(1/\delta) + d \log(1 + TX^2/(\lambda d))} \right),$$

also, we show that for all $\delta > 0$, OFUL^f achieves with probability at least $1 - \delta$, for all $T \ge 1$:

$$R_T^f \le 4 \sqrt{T d \log(\lambda + TX^2/d)} \left((\lambda^{1/2} + X) S + \sigma \sqrt{2 \log(1/\delta) + d \log(1 + TX^2/(\lambda d))} \right).$$

Proof. OFUL with forward regression: Lets decompose the instantaneous regret as follows:

$$\begin{aligned}
r_t &= \langle \theta_*, x_* \rangle - \langle \theta_*, x_t \rangle \le \left\langle \tilde{\theta}_t, x_t \right\rangle - \langle \theta_*, x_t \rangle = \left\langle \tilde{\theta}_t - \theta_*, x_t \right\rangle \\
&= \left\langle \widehat{\theta}_{t-1} - \theta_*, x_t \right\rangle + \left\langle \tilde{\theta}_t - \widehat{\theta}_{t-1}, x_t \right\rangle \\
&= \left\| \widehat{\theta}_{t-1} - \theta_* \right\|_{(G_{t-1} + x_t x_t^\top)} \|X_t\|_{(G_{t-1} + x_t x_t^\top)^{-1}} + \left\| \tilde{\theta}_t - \widehat{\theta}_{t-1} \right\|_{(G_{t-1} + x_t x_t^\top)} \|x_t\|_{(G_{t-1} + x_t x_t^\top)^{-1}} \\
&\le 2 \sqrt{\beta_{t-1}(x_t, \delta)} \, \|x_t\|_{(G_{t-1} + x_t x_t^\top)^{-1}}, \qquad (4.17)
\end{aligned}$$

where $\widetilde{\theta}_t$ is the optimistic parameter estimate, *i.e.* the $\theta \in C_t(x_t)$ that maximizes the upper confidence bound on the reward of action x_t. The first inequality is since $\left(X_t, \widetilde{\theta}_t\right)$ is optimistic, and the last step holds by Cauchy-Schwarz. Using Inequality (4.17) and the expression of the

confidence interval $(C_t(x))$ for the forward algorithm at the action x:

$$\left\{\theta \in \mathbb{R}^d : ||\theta_t^{\mathtt{f}} - \theta||_{G_t + xx^\top} \leq \sqrt{\beta_t(x,\delta)} = (\sqrt{\lambda} + ||x||_2)S + \sigma\sqrt{2\log\left(\frac{(1+tX^2/\lambda d)^{d/2}}{\delta}\right)}\right\}$$

we get that, with probability at least $1-\delta$, for all $n \geq 0$

$$\begin{aligned} R_n &\leq \sqrt{n\sum_{t=1}^n r_t^2} \leq \sqrt{8\beta_n(\delta) n \sum_{t=1}^n \|x_t\|_{(G_{t-1}+x_t x_t^\top)^{-1}}} \\ &\leq 4\sqrt{nd\log(\lambda + nL/d)}\left((\lambda^{1/2}+X)S + \sigma\sqrt{2\log(1/\delta) + d\log(1+nL/(\lambda d))}\right) \end{aligned}$$

where the last step follow from Lemma B.2.

OFUL with ridge regression: Now we derive a novel regret bound for online ridge regression, one that doesn't require the bounded rewards, *i.e.* Assumption 4.14.
The proof follows exactly like the above one except the last step (control of the norm of actions) that now proceeds using Lemma B.3. The first step is to use the confidence ellipsoid for the ridge regression parameter (*cf* Theorem 2 of Abbasi-Yadkori, Pál, and Szepesvári (2011)). With probability at least $1-\delta$, for all $t \geq 0, \theta_*$ lies in the set

$$C_t = \left\{\theta \in \mathbb{R}^d : \|\theta_t^{\mathtt{r}} - \theta\|_{G_t} \leq \sqrt{\beta_t(\delta)} = \sigma\sqrt{d\log\left(\frac{1+tX^2/\lambda d}{\delta}\right)} + \lambda^{1/2}S\right\}.$$

Then

$$\begin{aligned} r_t &= \langle\theta_*, x_*\rangle - \langle\theta_*, x_t\rangle \leq \left\langle\tilde{\theta}_t, x_t\right\rangle - \langle\theta_*, x_t\rangle = \left\langle\tilde{\theta}_t - \theta_*, x_t\right\rangle \\ &= \left\langle\widehat{\theta}_{t-1} - \theta_*, x_t\right\rangle + \left\langle\tilde{\theta}_t - \widehat{\theta}_{t-1}, x_t\right\rangle \\ &= \left\|\widehat{\theta}_{t-1} - \theta_*\right\|_{G_{t-1}} \|X_t\|_{G_{t-1}^{-1}} + \left\|\tilde{\theta}_t - \widehat{\theta}_{t-1}\right\|_{G_{t-1}} \|x_t\|_{G_{t-1}^{-1}} \\ &\leq 2\sqrt{\beta_{t-1}(\delta)}\,\|x_t\|_{G_{t-1}^{-1}} \end{aligned} \tag{4.18}$$

where $\widetilde{\theta}_t$ is the optimistic parameter estimate, *i.e.* the $\theta \in C_t$ that maximizes the upper confidence bound on the reward of action x_t. The first inequality is since $\left(X_t, \widetilde{\theta}_t\right)$ is optimistic, and the last step holds by Cauchy-Schwarz. Using Inequality (4.18) we get that, with probability at least $1-\delta$, for all $n \geq 0$

$$\begin{aligned} R_n &\leq \sqrt{n\sum_{t=1}^n r_t^2} \leq \sqrt{8\beta_n(\delta) n \sum_{t=1}^n \|x_t\|_{G_{t-1}^{-1}}} \\ &\leq 4\sqrt{\frac{ndX^2\log(1+nX^2/\lambda d)}{\lambda\log(1+X^2/\lambda)}}\left(\lambda^{1/2}S + \sigma\sqrt{2\log(1/\delta) + d\log(1+nX^2/(\lambda d))}\right) \end{aligned}$$

where the last step follow from Lemma B.3.

□

Remark 4.16. *we can drop the dependence on X and S by bounding the second term in the index of OFUL and OFUL$^{\mathtt{f}}$ (see line 285) by $XS(1 + X/\sqrt{\lambda})$ and then dropping this -constant- term at the expense of a looser index. Therefore, knowing the bounds x and S is not crucial. Furthermore, while we choose to adopt the pseudo-regret definition like in (Abbasi-Yadkori, Pál, and Szepesvári, 2011), we could also derive similar bounds for the regret involving rewards $y_t = \langle x_t, \theta_* \rangle$ instead of their expected value, $(y_t)_{t \geq 1}$ are unbounded.*

Experiment We provide experimental evidence that the OFUL$^{\mathtt{f}}$ variant improves OFUL for linear bandits; we find that it is generally as good as the standard OFUL$^{\mathtt{r}}$, and in some cases it can prove to be significantly more robust to aberrant regularization parameters. We consider a 100-dimensional linear bandit with 10 arms, the parameter vector is drawn from the unit ball, actions are such that $\|x_t\| \leq 200$. Noise $\varepsilon_t \overset{\mathcal{L}}{=} \mathcal{N}(0, 10^{-1})$, $\lambda = 10^{-5}$, $\delta = 10^{-3}$.

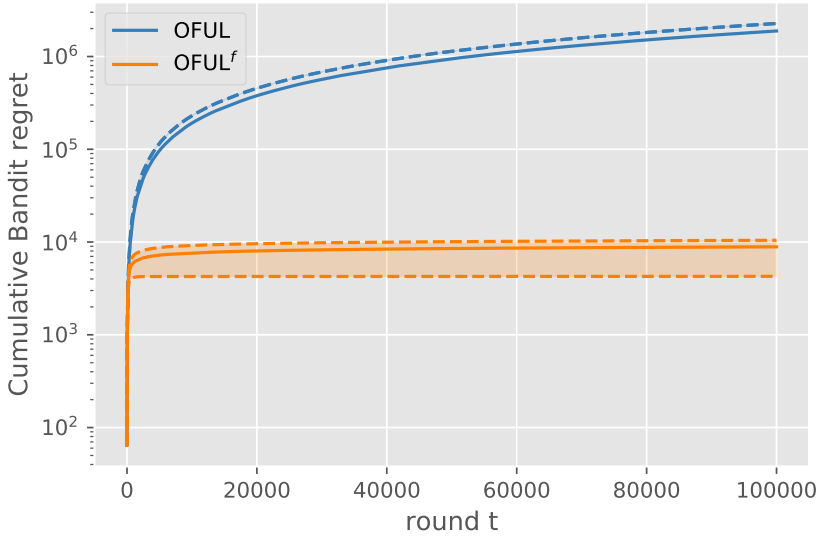


Figure 4.3 – Cumulative regret. y-axis is logarithmic.

In Figure 4.3, lines are average regret over 100 repetitions and shaded areas cover the region between dashed-lines that are the first and third quartiles. We observe that -as predicted by Theorem 4.15: OFUL$^{\mathtt{f}}$ is particularly robust and choosing $\lambda = 1/T$ incurs substantial regret for OFUL. Because of this phenomena, and for the same observations in the online stochastic regression setting, we advocate for the use of the forward algorithm instead of ridge regression whenever possible, to take advantage of its increased robustness to λ.

Remark 4.17. *Regarding the choice $\lambda = 1/T$: we use this specific regularization for two reasons: 1) to demonstrate the benefits of our stochastic analysis, since previous deterministic bounds suggest this λ is best, 2) to showcase the increased robustness of OFUL^f compared to OFUL. In fact, more often than not, OFUL performs as good as OFUL, except when λ is small or X is large.*

4.5.2 Non-stationary bandits

In this section, we study linear stochastic bandits in the non-stationary setting. We analyze then provide an experimental study of this setup. Consider *non-stationary stochastic linear bandits*, where the target parameter is varying with time: $\theta_* = \theta_*(t) \in \mathbb{R}^d$, assuming that $\sum_{s=1}^{T-1} \|\theta_*(s) - \theta_*(s+1)\|_2 \leq B_T$.

One of the optimal algorithms in this setting is `D-LinUCB` of (Russac, Vernade, and Cappé, 2019), it defines θ_t as

$$\theta_t = \arg\min_{\theta \in \mathbb{R}^d} \sum_{s=1}^{t} \gamma^{t-s}(y_s - \langle x_s, \theta \rangle)^2 + \lambda/2 \|\theta\|_2^2.$$

`D-LinUCB` proceeds as follows:

Algorithm 4.4: `D-LinUCB`

Input: $\delta, \sigma, \lambda, X, S, \gamma > 0$, dimension $d \in \mathbb{N}^*$.
Initialization: $b = 0_{\mathbb{R}^d}$, $V = \lambda I_d$, $\widetilde{V} = \lambda I_d$, $\theta = 0_{\mathbb{R}^d}$
for $t \geq 1$ **do**
  Receive $\mathcal{X}$, compute $\beta_{t-1} = \sqrt{\lambda} S + \sigma \sqrt{2 \log\left(\frac{1}{\delta}\right) + d \log\left(1 + \frac{X^2(1-\gamma^{2(t-1)})}{\lambda d (1-\gamma^2)}\right)}$
  for $a \in \mathcal{X}$ **do**
    Compute $\text{UCB}(a) = a^\top \theta + \beta_{t-1} \sqrt{a^\top V^{-1} \widetilde{V} V^{-1} a}$
  $A_t = \arg\max_a(\text{UCB}(a))$
  Play action A_t and **receive reward** X_t
  Updating phase: $V = \gamma V + x_t x_t^\top + (1-\gamma)\lambda I_d$, $\widetilde{V} = \gamma^2 \widetilde{V} + x_t x_t^\top + (1-\gamma^2)\lambda I_d$
    $b = \gamma b + Y_t X_t$, $\theta = V^{-1} b$

We recall the regret bound of standard `D-LinUCB` .

Theorem 4.18. (*Theorem 3 of Russac, Vernade, and Cappé (2019)*) *Assuming that*

$$\sum_{s=1}^{T-1} \|\theta_*(s) - \theta_*(s+1)\|_2 \leq B_T$$

and $\forall x \in \mathcal{X}, t \geq 1 : \langle x, \theta_t \rangle \leq 1$*, the regret of the* `D-LinUCB` *algorithm is bounded for all* $\gamma, \delta \in (0,1)$ *and integer* $D \geq 1$*, with probability at least* $1-\delta$*, by:*

$$R_T^r \leq 2XDB_T + \frac{4X^3S}{\lambda}\frac{\gamma^D}{1-\gamma}T + 2\sqrt{2}\beta_T\sqrt{dT} \times \sqrt{T\log(1/\gamma) + \log\left(1 + \frac{X^2}{d\lambda(1-\gamma)}\right)},$$

where β_T is the width of the confidence interval for $\theta_*(T)$.

Now we introduce `D-LinUCB` $^{\mathtt{f}}$, which uses the forward algorithm and defines an action dependent θ_t as:

$$\arg\min_{\theta \in \mathbb{R}^d} \sum_{s=1}^{t} \gamma^{t-s}(y_s - \langle x_s, \theta \rangle)^2 + \lambda/2\|\theta\|_2^2 + \langle x, \theta \rangle^2. \tag{4.19}$$

Theorem 4.19. *Assuming that* $\sum_{s=1}^{T-1} \|\theta_*(s) - \theta_*(s+1)\|_2 \leq B_T$*, the regret of the* `D-LinUCB` $^{\mathtt{f}}$ *is bounded for all* $\gamma, \delta \in (0,1)$ *and integer* $D \geq 1$*, with probability at least* $1-\delta$*, by*

$$R_T^f \leq 2XDB_T + \frac{4X^3S}{\lambda}\frac{\gamma^D}{1-\gamma}T + 2\beta_T\sqrt{dT}\sqrt{T\log(1/\gamma) + \log\left(1 + \frac{(2-\gamma)X^2}{d\lambda(1-\gamma)}\right)}.$$

Proof. This result is again a modification of the original proof consisting in bounding the sum of the actions' norms differently. Let us recall the notations $V_t = \sum_{s=1}^{t} w_s x_s x_s^\top + \lambda_t I_d + xx^\top$ and $\tilde{V}_t = \sum_{s=1}^{t} w_s^2 x_s x_s^\top + \mu_t I_d + xx^\top$. To summarize the difference of this analysis -that no longer requires a bounded rewards assumption- at the step where we bound the sum of actions' norms, we replace Proposition 4 of Russac, Vernade, and Cappé (2019):

$$\sum_{t=1}^{T} \min\left(1, \|x_t\|^2_{V_{t-1}^{-1}\tilde{V}_{t-1}V_{t-1}^{-1}}\right) \leq 2\sum_{t=1}^{T} \log\left(1 + \gamma^{-t}\|x_t\|^2_{V_{t-1}^{-1}}\right) \leq 2\log\left(\frac{\det(V_T)}{\lambda^d}\right),$$

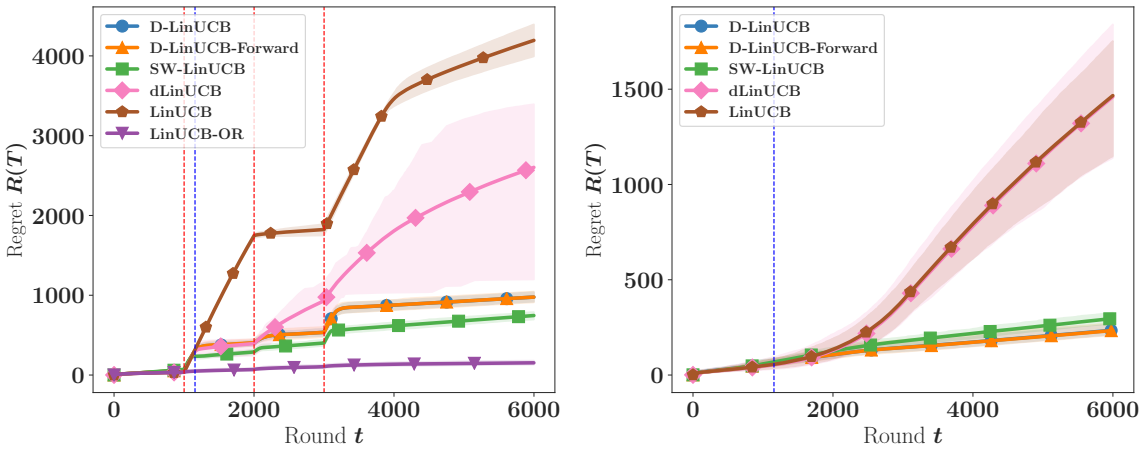
that requires the predictions to lie in the same range as the rewards with this inequality for `D-LinUCB` $^{\mathtt{f}}$

$$\sum_{t=1}^{T} \|x_t\|^2_{V_t^{-1}\tilde{V}_t V_t^{-1}} \leq \sum_{t=1}^{T} \log\left(1 + \gamma^{-t}\|x_t\|^2_{V_t^{-1}}\right) \leq \log\left(\frac{\det(V_T)}{\lambda^d}\right).$$

We don't provide the full proof of this result as it is cumbersome and not of special interest for our purposes since it is similar to the analysis for `D-LinUCB` except for the inequality above. □

Remark 4.20. *This result is fascinating as it first allows to remove an unnecessary assumption, and further yields a better bound than* `D-LinUCB` $^{\mathtt{r}}$ *which suffers the factor* $\frac{X\sqrt{2}}{\lambda\log(1+X/\lambda)}$ *in its last regret term without Assumption 4.14.*

Experiments for non-stationary linear bandits: We now reproduce the experiments of (Russac, Vernade, and Cappé, 2019) for non-stationary linear bandits, and add `D-LinUCB` $^{\mathtt{f}}$ to the



(**a**) abruptly changing environment

(**b**) slowly varying environment

Figure 4.4 – Performance of several algorithms in an non-stationary environments, averaged over 100 runs, shaded areas represent one standard deviation.

pool of algorithms. We first simulate an *abruptly* changing environment of dimension 2 with 3 changes: for $t < 10^3 : \theta_* = (1, 0)$; for $10^3 \leq t \leq 2.10^3 : \theta_* = (-1, 0)$; for $2.10^3 < t < 3.10^3 : \theta_* = (0, 1)$; for $t > 3.10^3 : \theta_* = (0, -1)$. We observe in Figure 4.4a that both variants of `D-LinUCB` compare on par. Here `LinUCB-OR` denotes an oracle knowing the change points.

Second, we simulate a slowly changing environment where the parameter θ_* starts at $(1, 0)$ and moves counter-clockwise on the unit-circle up to the position $(0, 1)$ in 3.10^3 steps then remains there, $B_T = 1.57$. We see the results in Figure 4.4b, where we notice that in this setting as well, `D-LinUCB` [f] has very similar performance to standard `D-LinUCB` .

Remark 4.21. *In both experiments, we also reported the performances of `SW-LinUCB`, that is alternative version to `D-LinUCB`. `SW-LinUCB` is better suited for abrupt changes while `D-LinUCB` is better suited for slow changes.*

Note that we added these final experiments to demonstrate the competitiveness of algorithms that use forward regression against their ridge counterparts in the same settings that were used by previous works. While we could have specified specific parameters to illustrate the robustness to regularization of algorithms that incorporate the forward algorithm; we estimate that the experiments presented in the main text already fulfilled this objective. Again, the purpose here is to show that using the forward algorithm improves the theoretical guarantees without deteriorating the performance.

4.6 Discussion

We revisited the analysis of online linear regression algorithms in the setup of stochastic, possibly unbounded observations. We proved high probability regret bounds for three popular online regression algorithms (*cf* Theorems 4.7, 4.9 and 4.11). These bounds provide novel understanding of online regression. In particular, Theorem 4.7 seems to be the first regret bound for ridge regression that does not require bounded predictions or prior knowledge of a bound on observations. Our novel bounds seem to correctly capture the nature of dependence with regularization, as indicated by Figure 4.2. Moreover, a new results from Tirinzoni, Pirotta, et al. (2020) can be incorporated in the proof mechanism to bring the high probability upper bounds to $O(d\sigma^2 \log(T) \log\log(T))$, which matches the optimal achievable bounds from the excess risk literature up to sub-logarithmic factors.

Furthermore, we argue that replacing ridge regression by the forward algorithm whenever possible in algorithms that require linear approximations can be beneficial, we depict this in a case study involving linear bandits: First from a theoretical standpoint our results show that the OFUL$^{\mathsf{f}}$ algorithm enjoys the classic first order regret bound while dropping Assumption 4.14; Second, we find that empirically, implementing OFUL with the forward algorithm makes the algorithm significantly more robust to extreme values of regularization, which is of practical interest.

More broadly, we believe that the improvement resulting from replacing ridge regression with the forward algorithm could be extended to several other settings. For instance, Graph bandits are of interest as well: they consider linear function approximations using ridge regression, and make Assumption 4.14, see for example Theorem 1 of (Valko et al., 2014); Meta-learning with linear bandits can also be enhanced using forward regression: see for example Lemma 1 and consequent results in (Cella, Lazaric, and Pontil, 2020).

Chapter 5

Continuous MDPs: the Bilinear Exponential Family representation

We study the problem of episodic reinforcement learning in continuous state-action spaces with unknown rewards and transitions. Specifically, we consider the setting where the rewards and transitions are modeled using parametric bilinear exponential families. We propose an algorithm, `BEF-RLSVI`, that a) uses penalized maximum likelihood estimators to learn the unknown parameters, b) injects a calibrated Gaussian noise in the parameter of rewards to ensure exploration, and c) leverages linearity of the bilinear exponential family transitions with respect to an underlying RKHS to perform tractable planning. We further provide a frequentist regret analysis of `BEF-RLSVI` that yields an upper bound of $\tilde{\mathcal{O}}\left(\sqrt{d^3H^3K}\right)$, where d is the dimension of the parameters, H is the episode length, and K is the number of episodes. Our analysis improves the existing bounds for the bilinear exponential family of MDPs by $\sqrt{H}$ and removes the handcrafted clipping deployed in existing `RLSVI`-type algorithms. Our regret bound is order-optimal with respect to H and K.[1]

Contents

[1]This chapter is based on a collaboration with Debabrota Basu and Odalric Maillard (Ouhamma, Basu, and Maillard, 2023). It was accepted for publication as an oral at the *37th AAAI Conference on Artificial Intelligence*.

5.1 Introduction

Reinforcement Learning (RL) is a well-studied and popular framework for sequential decision making, where an agent aims to compute a *policy* that allows her to maximize the accumulated reward over a horizon by interacting with an *unknown* environment (Sutton and Barto, 2018).

Episodic RL. In this paper, we consider the episodic finite-horizon MDP formulation of RL, in short *Episodic RL* (Osband, Russo, and Van Roy, 2013; Azar, Osband, and Munos, 2017; Dann, Lattimore, and Brunskill, 2017). Episodic RL is a tuple $\mathcal{P} = \langle \mathcal{S}, \mathcal{A}, \mathbf{P}, r, K, H \rangle$, where the state (resp. action) space $\mathcal{S}$ (resp. $\mathcal{A}$) might be continuous. In episodic RL, the agent interacts with the environment in episodes consisting of H steps. Episode k starts by observing state s_1^k. Then, for $t = 1, \dots H$, the agent draws action a_t^k from a (possibly time-dependent) policy $\pi_t(s_t^k)$, observes the reward $r(s_t^k, a_t^k) \in [0, 1]$, and transits to a state $s_{t+1}^k \sim \mathbf{P}(. \mid s_t^k, a_t^k)$ according to the transition function $\mathbf{P}$. The performance of a policy π is measured by the total expected reward V_1^π starting from a state $s \in \mathcal{S}$, the value function and the state-action value functions at step $h \in [H]$ are defined as

$$V_h^\pi(s) \triangleq \mathbf{E}\left[\sum_{t=h}^{H} r(s_t, a_t) \mid s_h = s\right], \quad Q_h^\pi(s, a) \triangleq \mathbf{E}\left[\sum_{t=h}^{H} r(s_t, a_t) \mid s_h = s, a_h = a\right].$$

Here, computing the policy leading to maximization of cumulative reward requires the agent to strategically control the actions in order to learn the transition functions and reward functions as precisely as required. This tension between learning the unknown environment and reward maximization is quantified as *regret*: the typical performance measure of an episodic RL algorithm. *Regret* is defined as the difference between the *expected cumulative reward* or *value* collected by the optimal agent that knows the environment and the expected cumulative reward or value obtained by an agent that has to learn about the unknown environment. Formally, the regret over K episodes is

$$\mathcal{R}(K) \triangleq \sum_{k=1}^{K} \left(V_1^{\pi^\star}(s_1^k) - V_1^{\pi_t}(s_1^k) \right).$$

Key Challenges. *The first challenge in episodic RL is to tackle the exploration–exploitation trade-off.* This is traditionally addressed with the *optimism principle* that either carefully crafts optimistic upper bounds on the value functions (Azar, Osband, and Munos, 2017), or maintains a posterior on the parameters to perform posterior sampling (Osband, Russo, and Van Roy, 2013), or perturbs the value function estimates with calibrated noise (Osband, Van Roy, and Wen, 2016). Though the first two approaches induce theoretically optimal exploration, they might not yield tractable algorithms for large/continuous state-action spaces as they either involve optimization in the optimistic set or maintaining a high-dimensional posterior. Thus, *we focus on extending*

the third approach of Randomized Least-Square Value Iteration (RLSVI) *framework, and inject noise only in rewards to perform tractable exploration.*

The second challenge, which emerges *for continuous state-action spaces, is to learn a parametric functional approximation of either the value function or the rewards and transitions* in order to perform planning and exploration. Different functional representations (or models), such as linear (Jin et al., 2020), bilinear (Du, Kakade, Lee, et al., 2021), and bilinear exponential families (Chowdhury, Gopalan, and Maillard, 2021), are studied in literature to develop optimal algorithms for episodic RL with continuous state-action spaces. Since the linear assumption is restrictive in real-life -where non-linear structures are abundant-, generalized representations have obtained more attention recently (Chowdhury, Gopalan, and Maillard, 2021; Li, Li, et al., 2021; Du, Kakade, Lee, et al., 2021; Foster et al., 2021). The BEF model is of special interest as it is expressive enough to represent tabular MDPs (discrete state-action), factored MDPs (Kearns and Koller, 1999), and linearly controlled dynamical systems (such as Linear Quadratic Regulators (Abbasi-Yadkori and Szepesvári, 2011)) as special cases (Chowdhury, Gopalan, and Maillard, 2021). Thus, in this paper, *we study the BEF of MDPs, i.e. the episodic RL setting where the rewards and transition functions can be modeled with bilinear exponential families.*

The third challenge is to perform tractable planning[2] *given the perturbation for exploration and the model class.* Existing work (Osband and Van Roy, 2014; Chowdhury, Gopalan, and Maillard, 2021) assumes an oracle to perform planning and yield policies that aren't explicit. The main difficulty in such planning approaches is calculating $\int \mathbf{P}(s' \mid s, a)V_h(s)$ for all (s, a) pairs. This is not trivial unless the transition is assumed to be linear and decouples s' from (s, a), which is not known to hold except for tabular MDPs. This challenge received attention recently, *e.g.* (Du, Kakade, Wang, et al., 2019) asks when misspecified linear representations are enough for a polynomial sample complexity in several settings. (Shariff and Szepesvári, 2020; Lattimore, Szepesvari, and Weisz, 2020; Van Roy and Dong, 2019) provide positive answers for certain linear settings. In this paper, *we aim to design a tractable planner for the BEF representation.*

In this paper, we aim to address the following question that encompasses the three challenges:

Can we design an algorithm with **tractable exploration** and **planning** for the *bilinear exponential family of MDPs* yielding a **near-optimal frequentist regret bound**?

Contributions. We address this question in three folds.

1. *Formalism:* We assume that neither rewards nor transitions are known, previous efforts on the bilinear exponential family of MDPs assumed knowledge of rewards. This makes the addressed problem harder, practical, and more general. We also observe that though the

[2]By tractable planning, we mean having a planner with (pseudo-)polynomial complexity in the problem parameters, i.e. in the dimension of features, the horizon, and the number of episodes.

Table 5.1 – A comparison of RL Algorithms for MDPs with functional representations.

Algorithm	Regret	Tractable exploration	Tractable planning	Free of clipping	Model, assumptions
Thompson sampling (Ren et al., 2021)	$\sqrt{d^2H^3K}$ (Bayesian)	✗	✓	N.A	Gaussian $\mathbf{P}$ Known rewards
`EXP-UCRL` (Chowdhury, Gopalan, and Maillard, 2021)	$\sqrt{d^2H^4K}$ (Frequentist)	✗	✗	N.A	Bilinear Exp Family (BEF) known rewards
`SMRL` (Li, Li, et al., 2021)	$\sqrt{d^2H^4K}$	✗	✗	N.A	BEF, known rewards
`UCRL-VTR` (**pmlr-v119-ayoub20a**)	$\sqrt{d^2H^4K}$	✗	✗	N.A	Linear mixture model
$\mathcal{F}-$`PHE-LSVI` (Ishfaq et al., 2021) `PHE-LSVI` (linear-RL)	$\text{poly}(d_E H)\sqrt{KH}$ $\sqrt{d^3H^4K}$	✓	✗	✗	Eluder dimension, Tabular Anti-concentration
`UC-MatrixRL` (Yang and Wang, 2020)	$\sqrt{d^2H^5K}$	✗	✗	N.A	Linear factor MDP
`OPT-RLSVI` (Zanette, Brandfonbrener, et al., 2020)	$\sqrt{d^4H^5K}$	✓	✓	✗	Linear V
`BEF-RLSVI` (this work)	$\sqrt{d^3H^3K}$	✓	✓	✓	Bilinear Exp Family

transition model can represent non-linear dynamics, it implies a linear behavior (see Section 5.2) in a Reproducible Kernel Hilbert Space (RKHS). This observation contributes to the tractability of planning.

2. *Algorithm:* We propose an algorithm `BEF-RLSVI` that extends the `RLSVI` framework to bilinear exponential families (*cf* Section 5.3). `BEF-RLSVI` a) injects calibrated Gaussian noise in the rewards to perform exploration, b) leverages linearity of the transitions with respect to an underlying RKHS to perform tractable planning and c) uses penalized maximum likelihood to learn the parameters corresponding to rewards and transitions (*cf* Section 5.4). To the best of our knowledge, *`BEF-RLSVI` is the first algorithm for the bilinear exponential family of MDPs with tractable exploration and planning under unknown rewards and transitions.*

3. *Analysis:* We carefully develop an analysis of `BEF-RLSVI` that yields $\tilde{\mathcal{O}}\left(\sqrt{d^3H^3K}\right)$ regret which improves the existing regret bound for the BEF of MDPs with known rewards by a factor of $\sqrt{H}$ (Section 5.3.2). Our analysis builds on existing analyses of RLSVI-type algorithms (Osband, Van Roy, and Wen, 2016), but contrary to them, we remove the need to handcraft a clipping of the value functions (Zanette, Brandfonbrener, et al., 2020). We also do not need to *assume* anti-concentration bounds as we can explicitly control it by the injected noise. This was not done previously except for the linear MDPs. We illustrate this comparison in Table 5.1. We highlight three technical tools that we used to improve the previous analyses: 1) Using transportation inequalities instead of the simulation lemma reduces a $\sqrt{H}$ factor compared to (Ren et al., 2021), 2) Leveraging the observation that true value functions are bounded enables using an improved elliptical lemma (compared to (Chowdhury, Gopalan, and Maillard, 2021)), and 3) Noticing that the norm of features can only be large for a finite amount of time allows us to forgo clipping and reduce a $\sqrt{d}$ factor from the regret compared to (Zanette, Brandfonbrener, et al., 2020).

5.2 Bilinear exponential family of MDPs

We introduce the BEF model (Chowdhury, Gopalan, and Maillard, 2021) and extend it to parametric rewards. Then, we make an important observation of linearity.

Definition 5.1 (Bilinear exponential family model)**.** *We consider both transition and reward kernels to be unknown and modeled with bilinear exponential families. Specifically,*

$$\mathbb{P}\left(\tilde{s} \mid s,a\right) = \exp\left(\psi(\tilde{s})^\top M_{\theta^{\mathtt{p}}}\varphi(s,a) - Z^{\mathtt{p}}_{s,a}(\theta^{\mathtt{p}})\right), \tag{5.1}$$

$$\mathbb{P}\left(r \mid s,a\right) = \exp\left(r\, B^\top M_{\theta^{\mathtt{r}}}\varphi(s,a) - Z^{\mathtt{r}}_{s,a}(\theta^{\mathtt{r}})\right), \tag{5.2}$$

where $\varphi \in (\mathbf{R}_+^q)^{\mathcal{S}\times\mathcal{A}}$ and $\psi \in (\mathbf{R}_+^p)^{\mathcal{S}}$ are known feature functions, and $B \in \mathbf{R}^p$ is a known scaling factor. The unknown reward and transition parameters are $\theta^{\mathtt{p}}, \theta^{\mathtt{r}} \in \mathbf{R}^d$. $M_{\theta^\cdot} \triangleq \sum_{i=1}^d \theta^\cdot_i A_i$, where A_i is a known $p \times q$ matrix for each i. Finally, Z denotes the log partition function:

$$Z^{\mathtt{p}}_{s,a}(\theta^{\mathtt{p}}) \triangleq \log \int_{\mathcal{S}} \exp\left(\psi(\tilde{s})^\top M_{\theta^{\mathtt{p}}}\varphi(s,a)\right) d\tilde{s},$$

$Z^{\mathtt{r}}$ is defined similarly.

A minor difference with the original BEF model and the one stated here is that, like (Li, Li, et al., 2021), we omit a base measure of the form $h(s,\tilde{s},a)$, all the BEF examples provided in (Chowdhury, Gopalan, and Maillard, 2021) still hold with this slight restriction. We denote $V^{\pi}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}},h}$, (resp. $Q^{\pi}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}},h}$) the value (resp. state-action) value function for policy π in the MDP parameterized by $(\theta^{\mathtt{p}},\theta^{\mathtt{r}})$ at time h. A policy $\pi^\star$ is *optimal* if for all $s \in \mathcal{S}$, $V^{\pi^\star}_{\theta,h}(s) = \max_{\pi\in\Pi} V^{\pi}_{\theta,h}(s)$. A learning algorithm minimizes the (pseudo) regret:

$$\mathcal{R}(K) \triangleq \sum_{k=1}^{K}\left(V^{\pi^\star}_{\theta,1}(s_1^k) - V^{\pi^t}_{\theta,1}(s_1^k)\right). \tag{5.3}$$

Linearity of transitions. Now, we state an observation about the bilinear exponential family and discuss how it helps with the challenge of planning in episodic RL. Specifically, the popular assumption of linearity of the transition kernel is a direct consequence of our model. Indeed,

$$2\psi\left(s'\right)^\top M_{\theta^{\mathtt{p}}}\varphi(s,a) = -\|(\psi(s') - M_{\theta^{\mathtt{p}}}\varphi(s,a)\|^2 + \|\psi(s')\|^2 + \|M_{\theta^{\mathtt{p}}}\varphi(s,a)\|^2.$$

Notice that the quadratic term is the Radial Basis Function (RBF) kernel. More precisely, for an RBF kernel with covariance $\Sigma = I_p$ and $k(x,y) \triangleq \exp\left(-\|x-y\|^2/2\right)$, we find

$$\mathbb{P}\left(s' \mid s,a\right) = \langle \phi^{\mathtt{p}}(s,a), \mu^{\mathtt{p}}(s')\rangle_{\mathcal{H}}, \tag{5.4}$$

where $\mathcal{H}$ is the RKHS associated with the $k(.,.)$, and

$$\mu^{\mathtt{p}}(s') = (2\pi)^{-p/2}\, k\left(\psi(s'),.\right)\, \exp\left(\|\psi\left(s'\right)\|^2/2\right)$$
$$\phi^{\mathtt{p}}(s,a) = k\left(M_{\theta^{\mathtt{p}}}^{\top}\varphi(s,a),.\right) \exp\left(\frac{\|M_{\theta^{\mathtt{p}}}^{\top}\varphi(s,a)\|^2}{2} - Z_{s,a}(\theta^{\mathtt{p}})\right)$$

In Equation (5.4), s' is decoupled from (s,a), we see hereafter why this is crucial to reducing the complexity of planning.

Remark 5.2. *Up to our knowledge, (Ren et al., 2021) is the only work providing an example of linear transitions for RL with continuous state-actions. They consider Gaussian transitions with an unknown mean $(f^{\star}(s,a))$ and known variance. It is a special case of the BEF model, where $\psi(s') = (s', \|s'\|^2)$ and $M_{\theta}\varphi(s,a) = (f_{\theta}(s,a)/\sigma^2, -1/\sigma^2)$.*

Importance of linearity. To understand the planning challenge in RL, recall the Bellman equation:

$$Q_h^{\pi}(s,a) = r(s,a) + \int_{\tilde{s}\in\mathcal{S}} P(s' \mid s,a) V_{h+1}^{\pi}(\tilde{s}) d\tilde{s},$$

We must approximate the integral at the R.H.S.for $(s,a) \in \mathcal{S}\times\mathcal{A}$. For a tabular MDP with $|S|$ states and $|A|$ actions, we need to evaluate $(Q_h^{\pi})_{h\in[H]}$, i.e. to approximate $|S|\times|A|\times H$ integrals per episode, which can be very expensive. However, if the transition model is linear (Equation (5.4)), then

$$Q_{\theta,h}^{\pi}(s,a) = r(s,a) + \left\langle \phi^{\mathtt{p}}(s,a), \int_{\mathcal{S}} \mu^{\mathtt{p}}(\tilde{s}) V_{\theta,h+1}^{\pi}(\tilde{s}) d\tilde{s} \right\rangle. \tag{5.5}$$

When $\phi^{\mathtt{p}}, \mu^{\mathtt{p}} \in \mathbf{R}^{\tau}$, we can obtain Qh by computing τ integrals per timestep, reducing the state-action space complexity to τ only. For our model, although $\phi^{\mathtt{p}}$ and $\mu^{\mathtt{p}}$ are infinite dimensional, we show in Section 5.4 (§ planning) that the planning complexity is still significantly reduced.

5.3 `BEF-RLSVI`: algorithm design and frequentist regret bound

We formally introduce and the Bilinear Exponential Family Randomized Least Squares Value Iteration algorithm (`BEF-RLSVI`) and provide a high probability regret bound.

5.3.1 `BEF-RLSVI`: algorithm design

`BEF-RLSVI` is based on `RLSVI` (Osband, Van Roy, and Wen, 2016) except it perturb the reward parameter only. The latter is reminiscent of Thompson Sampling, yet more explicit and with a better control of the optimism probability.

Algorithm 5.1: BEF-RLSVI

1: **Input:** failure rate δ, constants $\alpha^{\mathtt{p}}, \eta$ and $(x_k)_{k\in[K]} \in \mathbf{R}^+$
2: **for** episode $k = 1, 2, \dots$ **do**
3: Observe initial state s_1^k
4: Sample noise $\xi_k \sim \mathcal{N}\left(0, x_k(\bar{G}_k^{\mathtt{p}})^{-1}\right)$ such that

$$\bar{G}_k^{\mathtt{p}} = \tfrac{\eta}{\alpha^{\mathtt{p}}}\mathbf{A} + \textstyle\sum_{\tau=1}^{k-1}\sum_{h=1}^{H}(\varphi(s_h^\tau, a_h^\tau)^\top A_i^\top A_j \varphi(s_h^\tau, a_h^\tau))_{i,j\in[d]}$$

5: Perturb reward parameter: $\tilde{\theta}^{\mathtt{r}}(k) = \hat{\theta}^{\mathtt{r}}(k) + \xi_k$
6: Compute $(Q^k_{\hat{\theta}^{\mathtt{p}},\tilde{\theta}^{\mathtt{r}},h})_{h\in[H]}$ via Bellman-backtracking, see Algorithm 5.2
7: **for** $h = 1, \dots, H$ **do**
8: Pull action $a_h^k = \arg\max_a Q_{\hat{\theta}^{\mathtt{p}},\tilde{\theta}^{\mathtt{r}},h}(s_h^k, a)$
9: Observe reward $r(s_h^k, a_h^k)$ and state s_{h+1}^k.
10: **end for**
11: Update the penalized ML estimators $\hat{\theta}^{\mathtt{p}}(k), \hat{\theta}^{\mathtt{r}}(k)$, see Equation (5.6) and Equation (5.7)
12: **end for**

In line 4, BEF-RLSVI performs exploration by a Gaussian perturbation of the reward parameter. Contrary to optimistic approaches, this method is explicit and more efficient since it does not a involve high-dimensional optimization.

Algorithm 5.2: Bellman Backtracking

1: **Input** Parameters $\hat{\theta}^{\mathtt{p}}, \tilde{\theta}^{\mathtt{r}}$, initialize $\tilde{\theta} = (\tilde{\theta}^{\mathtt{r}}, \hat{\theta}^{\mathtt{p}})$ and for all $s \in \mathcal{S}, V_{H+1}(s) = 0$
2: **for** steps $h = H-1, H-2, \cdots, 0$ **do**
3: Calculate $Q_{\tilde{\theta},h}(s,a) = \mathbf{E}^{\tilde{\theta}^{\mathtt{r}}}_{s,a}[r] + \langle \phi^{\mathtt{p}}(s,a), \int V_{\tilde{\theta},h+1}(s')\mu^{\mathtt{p}}(s')ds' \rangle_{\mathcal{H}}$
4: **end for**

Line 3 can be approximated with $\mathcal{O}(pH^3K\log(HK))$ complexity without damaging the sample complexity (*cf* § planning, Section 5.4). Therefore, planning is tractable.

Remark 5.3. *The observation of linearity* (cf *Equation* (5.5) *and Line* 3) *does not reduce BEF MDPs to linear MDPs because the former holds in an RKHS. Also, linearity is not in the representation parameter. Therefore, linear RL algorithms do not readily solve the BEF MDPs.*

5.3.2 BEF-RLSVI: regret upper-bound

We state the standard smoothness assumptions on the model (Chowdhury, Gopalan, and Maillard, 2021; Jun et al., 2017; Lu, Meisami, and Tewari, 2021).

Assumption 5.4. *There exist constants $\alpha^{p}, \alpha^{r}, \beta^{p}, \beta^{r} > 0$, such that the representation model satisfies, for all $(s,a) \in \mathcal{S} \times \mathcal{A}$, and for all $\theta, x \in \mathbf{R}^d$*

$$\alpha^{p} \leq x^\top C^{\theta}_{s,a}[\psi] x \leq \beta^{p}, \alpha^{r} \leq \mathbb{V}\mathrm{ar}^{\theta}_{s,a}(r)\, x^\top B^\top B x \leq \beta^{r},$$

where $\mathbb{V}\mathrm{ar}^{\theta}_{s,a}(r) \triangleq \left(\mathbb{E}^{\theta}_{s,a}[r^2] - \mathbb{E}^{\theta}_{s,a}[r]^2\right)$, *and* $\mathbb{C}^{\theta}_{s,a}[\psi(s')] \triangleq \mathbb{E}_{\mathbf{P}_\theta|s,a}[\|\psi(s')\|^2] - \left\|\mathbb{E}_{\mathbf{P}_\theta|s,a}\left[\psi(s')\right]\right\|^2$

These inequalities imply a control over the eigenvalues of the Hessian matrices of the log-normalizers (*cf* Appendix C.2.3). We now state our main result.

Theorem 5.5. [*Regret bound*] *Let* $\mathbf{A} \triangleq ({}^\top(A_i A_j^\top))_{i,j\in[d]}$ *and* $G_{s,a} \triangleq (\varphi(s,a)^\top A_i^\top A_j \varphi(s,a))_{i,j\in[d]}$. *Under Assumption 5.4 and further considering that*

1. $\max\{\|\theta^{r}\|_{\mathbf{A}}, \|\theta^{p}\|_{\mathbf{A}}\} \leq B_{\mathbf{A}}$, $\|\mathbf{A}^{-1} G_{s,a}\| \leq B_{\varphi,\mathbf{A}}$ *and* $\mathbf{E}_{\theta^r}[r(s,a)] \in [0,1]$ *for all* (s,a).
2. *noise* $\xi_k \sim \mathcal{N}(0, x_k(\bar{G}_k^{p})^{-1})$ *satisfies* $x_k \geq \left(H\sqrt{\frac{\beta^{p}\beta^{p}(K,\delta)}{\alpha^{p}\alpha^{r}}} + \frac{\sqrt{\beta^{r}\beta^{r}(K,\delta)\min\{1, \frac{\alpha^{p}}{\alpha^{r}}\}}}{2\alpha^{r}}\right)^2 \propto dH^2$,

   then for all $\delta \in (0,1]$, with probability at least $1-\delta$,

$$\begin{aligned}
\mathcal{R}(K) \leq \sqrt{H\gamma_K^{r}} \Bigg[& \underbrace{\beta^{r} C_d \left(\sqrt{\frac{\beta^{r}(K,\delta)}{2\alpha^{r}}} + c\sqrt{x_K d \log(dK/\delta)} \right)}_{\text{Estimation error for no clipping} \approx dH} \\
& + \underbrace{\frac{\beta^{r} d\sqrt{x_K}}{\Phi(-1)} (1 + \sqrt{\log(d/\delta)}) \sqrt{C_d \left(1 + \frac{\alpha^{r} B_{\varphi,A} H}{\eta}\right)}}_{\text{Learning error for no clipping} \approx (dH)^{3/2}} \Bigg] \\
& + \Bigg[\underbrace{c\,\beta^{r}\sqrt{x_K d \gamma_K^{r} \log(dK/\delta)} + \frac{\beta^{r}\sqrt{x_K d \gamma_K^{r} \log(e/\delta^2)}}{\Phi(-1)} \left(1 + \sqrt{\log(d/\delta)}\right)}_{\text{Noise concentration} \approx d^{3/2}H} \\
& + \underbrace{\beta^{r}\sqrt{\frac{\beta^{r}(n,\delta)\gamma_K^{r}}{2\alpha^{r}}}}_{\text{Reward concentration} \approx d} + \underbrace{2H\left(\sqrt{\frac{2\beta^{p}}{\alpha^{p}}\beta^{p}(K,\delta)\gamma_K^{p}} + (1+\sqrt{\gamma_K^{r}})\sqrt{\log(1/\delta^2)}\right)}_{\text{Transition concentration} \approx dH} \Bigg] \sqrt{KH}
\end{aligned}$$

where for $i \in [p, r]$, $\beta^{i}(K,\delta) \triangleq \frac{\eta}{2} B_{\mathbf{A}}^2 + \gamma_K^{i} + \log(1/\delta)$, and $\gamma_K^{i} \triangleq d\log(1 + \frac{\beta^{i}}{\eta} B_{\varphi,\mathbf{A}} HK)$. Also, $C_d \triangleq \frac{3d}{\log(2)} \log\left(1 + \frac{\alpha^{r}\|\mathbf{A}\|_2^2 B_{\varphi,\mathbf{A}}^2}{\eta \log(2)}\right)$, Φ is the Gaussian CDF, and c is a universal constant.

Theorem 5.5 entails a regret $\mathcal{R}(K) = \mathcal{O}\left(\sqrt{d^3H^3K}\right)$ for `BEF-RLSVI`, where d is the number of parameters of the bilinear exponential family model, K is the number of episodes, and H is the horizon of an episode. We now clarify how this contrasts with related literature.

Comparison with other bounds. The closest result is from (Chowdhury, Gopalan, and Maillard, 2021), it considers the same model for transitions but with known rewards. They propose a `UCRL`-type and `PSRL`-type algorithm, which achieve a $\widetilde{O}(\sqrt{d^2H^4K})$ regret. There are two notable algorithmic differences with `BEF-RLSVI`. First, they use intractable-optimistic upper bounds or high-dimensional posteriors, while we do explore with explicit perturbations. The second difference is in planning: while they assume access to a planning oracle, we do it explicitly with pseudo-polynomial complexity (Section 5.4.1). Moreover, we improve the regret bound by $\sqrt{H}$ thanks to an improved analysis, (*cf* Lemma C.12). But similar to all `RLSVI`-type algorithms, we pick up an extra $\sqrt{d}$ (*cf* (Abeille and Lazaric, 2017)).

(Zanette, Brandfonbrener, et al., 2020) proposes a variant of `RLSVI` for continuous state-action spaces, where there are low-rank models of transitions and rewards. They show a regret bound $R(K) = \widetilde{O}(\sqrt{d^4H^5K})$, which is larger than that of `BEF-RLSVI` by $O(\sqrt{dH^2})$. In algorithm design, we improve on their work by removing the need to carefully clip the value function. Analytically, our model allows us to use transportation inequalities (*cf* Lemma C.7) instead of the simulation lemma, which saves us a $\sqrt{H}$ factor.

(Ren et al., 2021) considers Gaussian transitions, i.e. $s' = f^*(s,a) + \varepsilon$ such that $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. This is a particular case of our model. They propose to use Thompson Sampling, and have the merit of being the first to have observed linearity of the value function from this transition structure. But they do not connect it to the finite dimensional approximation of (Rahimi and Recht, 2007) unlike us (Section 5.4). Finally, they show a Bayesian regret bound of $O(\sqrt{d^2H^3K})$. This notion of regret is weaker than frequentist regret, hence this result is not directly comparable with Theorem 5.5.

Tightness of regret bound. A lower bound for episodic RL with continuous state-action spaces is still missing. However, for tabular RL, (Domingues et al., 2021) proves a lower bound of order $\Omega(\sqrt{H^3SAK})$. If we represent a tabular MDP in our model, we would need $d = S^2 \times A$ parameters (Section 4.3, (Chowdhury, Gopalan, and Maillard, 2021)). In this case, our bound becomes $R(K) = O(\sqrt{(S^2A)^3H^3K})$, which is clearly not tight is S and A. This is understandable due to the relative generality of our setting. We are however positively surprised that **our bound is tight in terms of its dependence on H and K.**

5.4 Algorithm design

We discuss our choices for the design of `BEF-RLSVI` as well as the justification of tractability of some essential sub-procedures.

5.4.1 Building blocks of BEF-RLSVI

We present necessary details about BEF-RLSVI and discuss the key algorithm design techniques.

Estimation of parameters. We estimate transitions and rewards from observations similar to EXP-UCRL (Chowdhury, Gopalan, and Maillard, 2021), *i.e.* by using a penalized maximum likelihood estimator

$$\hat{\theta}^{\mathtt{p}}(k) \in \operatorname*{arg\,min}_{\theta \in \mathbf{R}^d} \sum_{\substack{1 \le t \le k \\ 1 \le h \le H}} -\log \mathbf{P}_\theta \left(s_{h+1}^t \mid s_h^t, a_h^t \right) + \eta \operatorname{pen}(\theta).$$

Here, $\operatorname{pen}(\theta)$ is the trace-norm penalty: $\operatorname{pen}(\theta) = \frac{1}{2}\|\theta\|_{\mathbb{A}}$ where $\mathbf{A} = (\operatorname{tr}(A_i A_j^\top))_{i,j}$. By properties of the exponential family, the penalized ML estimator verifies, for $i \le d$:

$$\sum_{\substack{1 \le t \le k \\ 1 \le h \le H}} \left(\psi\left(s_{h+1}^t\right) - \mathbb{E}_{s_h^t, a_h^t}^{\hat{\theta}_k^{\mathtt{p}}} \left[\psi\left(s'\right) \right] \right)^\top A_i \varphi\left(s_h^t, a_h^t\right) = \eta \nabla_i \operatorname{pen}\left(\hat{\theta}_k^{\mathtt{p}}\right). \tag{5.6}$$

The above can be solved in closed form for simple distributions, like Gaussian, but it can be involved for other distribution (*cf* Appendix 5.4.2). For the reward, θ_r is defined similarly:

$$\hat{\theta}^{\mathtt{r}}(k) \in \operatorname*{arg\,min}_{\theta \in \mathbf{R}^d} \sum_{\substack{1 \le t \le k \\ 1 \le h \le H}} -\log \mathbf{P}_\theta \left(r_t \mid s_h^t, a_h^t \right) + \eta \operatorname{pen}(\theta),$$

Then, for all $i \in [d]$:

$$\sum_{\substack{1 \le t \le k \\ 1 \le h \le H}} \left(r_t - \mathbb{E}_{s_h^t, a_h^t}^{\hat{\theta}_k^{\mathtt{r}}} [r] \right) B^\top A_i \varphi\left(s_h^t, a_h^t\right) = \eta \nabla_i \operatorname{pen}\left(\hat{\theta}_k^{\mathtt{r}}\right) \tag{5.7}$$

Exploration. A significant challenge in RL is handling exploration in continuous spaces. The majority of the literature is split between intractable, upper confidence bound-style optimism or Thompson sampling algorithms with high-dimensional posterior and guarantees only in terms of Bayesian regret. In BEF-RLSVI, we adopt the approach of reward perturbation motivated by the RLSVI-framework (Zanette, Brandfonbrener, et al., 2020; Osband, Van Roy, and Wen, 2016). We show that perturbing the reward estimation can guarantee optimism with a constant probability, *i.e.* there exists $\nu \in (0, 1]$ such that for all $k \in [K]$ and $s_1^k \in \mathcal{S}$,

$$\mathbb{P}\left(\tilde{V}_1(s_1^k) - V_1^\star(s_1^k) \ge 0 \right) \ge \nu.$$

(Zanette, Brandfonbrener, et al., 2020) proves that this suffices to bound the learning error. However, their method clashes with not clipping the value function, as it modifies the probability of optimism. Thus, (Zanette, Brandfonbrener, et al., 2020) proposes an involved clipping

procedure to handle the issue of unstable values. Instead, by careful geometric analysis (*cf* Lemma C.14), we bound the occurrences of the unstable values, and in turn, upper bound the regret without clipping. Note that unlike (Ishfaq et al., 2021), `BEF-RLSVI` does not guarantee that the estimated value function is optimistic but still is able to control the learning error (*cf* Section 5.5).

Planning. Recall that with our model assumptions, we can write the state-action value function linearly (Equation (5.5)). Using `BEF-RLSVI`, we have at step h:

$$Q^{\pi}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},h}(s,a) = \mathbf{E}_{\tilde{\theta}^{\mathrm{r}}}[r(s,a)] + \left\langle \phi^{\mathrm{p}}(s,a), \int_{\mathcal{S}} \mu^{\mathrm{p}}(\tilde{s}) V^{\pi}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},h+1}(\tilde{s}) d\tilde{s} \right\rangle.$$

Then, we select the best action greedily to compute $Q_h(s,a)$. Although ϕ^{p} and ψ^{p} are infinite-dimensional, we show (next paragraph) that an approximation with dimensionality of order $\mathcal{O}(pH^2K\log(HK))$ is possible, and that it doesn't increase the regret. Thus, the planning can be done in $\mathcal{O}(pH^3K\log(HK)))$, which is pseudo-polynomial in p, H and K, ergo tractable.

For details about the finite-dimensional approximation of our transition kernel, refer to Subsection 5.4.2. Now, we highlight the schematic of a finite-dimensional approximation of ϕ^{p} and ψ^{p}. We proceed in three steps. **1)** We have with high probability $\mathbf{S}(V_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},h}) \leq dH^{3/2}$ (Section 5.5). **2)** If we have a uniform ε-approximation of $\mathbf{P}_{\theta^{\mathrm{p}}}$, we show that using it incurs at most an extra $\mathcal{O}\left(\varepsilon dH^{5/2}K\right)$ regret. **3)** Finally, following (Rahimi and Recht, 2007), we approximate uniformly the shift invariant kernels, here the RBF in Equation (5.4), within ε error and with features of dimensions $\mathcal{O}\left(p\varepsilon^{-2}\log\frac{1}{\varepsilon^2})\right)$, where p is dimension of ψ. Associating these three elements and choosing $\varepsilon = 1/\sqrt{(H^2K)}$, we establish our claim.

5.4.2 Tractable Planning and Maximum likelihood estimation

A Primer on random Fourier transforms. We start by defining the Random Fourier Transform and its most relevant property. Let us consider the transition model of Equation (5.1), we have

$$\mathbf{P}(s' \mid s,a,\theta) = \exp\left(\psi(s')M_\theta\varphi(s,a) - Z_\theta(s,a)\right) = \mathbf{E}_{p(w,b)}\left[f\left(\psi(s'),w,b\right) f\left(M_\theta\varphi(s,a),w,b\right)\right],$$

where $f(x,w,b) = \sqrt{2}\cos(w^\top x + b)$ are the random Fourier bases. $p(w,b) = \mathcal{N}(0,\sigma^{-2}I) \times \mathcal{U}([0,2\pi])$, such that $\mathcal{N}$ is the Gaussian distribution, $\mathcal{U}$ is the Uniform distribution, and $p(w,b)$ is a coupling among them.

Notice that this provides an alternative approach to decompose the transition kernel and obtain linearity of the value function. Moreover, since $\forall x,w \in \mathbf{R}^d, b \in \mathbf{R}, |f(x,w,b)| \leq \sqrt{2}$, we can use Hoeffding's inequality to prove that a Monte-Carlo approximation of $\mathbf{P}(s' \mid s,a,\theta)$ using N sample pairs of (w,b) guarantees an error smaller than ε with probability at least

$1-2\exp(-N\varepsilon^2/4)$. (Rahimi and Recht, 2007) proves a stronger result: it provides an algorithm approximating the Gaussian kernel for which the following uniform convergence bound holds.

Lemma 5.6. *Let $\mathcal{M}$ be a compact subset of $\mathcal{R}^p$ with diameter $\operatorname{diam}(\mathcal{M})$. Then, using the explicit mapping $\mathbf{z}$ defined in Algorithm 1 in (Rahimi and Recht, 2007) with N samples, we have*

$$\Pr\left[\sup_{x,y\in\mathcal{M}} \left|\mathbf{z}(\mathbf{x})'\mathbf{z}(\mathbf{y}) - k(\mathbf{y},\mathbf{x})\right| \geq \varepsilon\right] \leq 2^8 \left(\frac{\sigma_p \operatorname{diam}(\mathcal{M})}{\varepsilon}\right)^2 \exp\left(-\frac{N\varepsilon^2}{4(p+2)}\right)$$

where $\sigma_p^2 \equiv E_p\left[\omega'\omega\right]$ is the second moment of the Fourier transform of k.

Further, it implies that if $N = \Omega\left(\frac{p}{\varepsilon^2}\log\frac{\sigma_p \operatorname{diam}(\mathcal{M})}{\varepsilon}\right)$, then $\sup_{x,y\in\mathcal{M}} |\mathbf{z}(\mathbf{x})'\mathbf{z}(\mathbf{y}) - k(\mathbf{y},\mathbf{x})| \leq \varepsilon$ with constant probability.

Application to planning in `BEF-RLSVI`. Since our regret analysis is done under the high probability event of bounded estimation parameters, we know that the spaces of $\psi(s')$ and $M_\theta\varphi(s,a)$ are bounded and the diameter depends on the dimensions. We abstain from explicating the exact diameter as it only influences the number of samples logarithmically. Using $N \approx p/\varepsilon^2$ samples, we can construct a uniform ε-approximation of $\mathbf{P}(s' \mid s,a,\theta)$.

Let's call $\hat{V}_h$ the estimated value function using Algorithm 3 with the above approximation of transition. Here, we elucidate the span of this estimation of value function. First we have:

$$\hat{V}_H^\pi - V_H^\pi = \int_{s'} (\hat{P}-P)(s' \mid s,a) r(s',\pi(s'))\mathrm{d}s' \leq \varepsilon d H^{3/2}$$

Here, we use the facts that $\mathbb{S}\left(V_{\hat{\theta},\tilde{\theta}^\times,h}\right) \leq dH^{3/2}$ (*cf* Section 5.5.2) and the error in approximating P is bounded by ε, i.e. $\sup_{s',s,a}|(\hat{P}-P)(s'|s,a)| \leq \varepsilon$.

Assume that at step $h+1$, we have $\hat{V}_{h+1}^\pi - V_{h+1}^\pi \leq \sum_{j=1}^{h+1}\varepsilon^j\alpha_{h+1,j}$. Then, we obtain

$$\begin{aligned}
\hat{V}_h^\pi - V_h^\pi &\leq \int_{s'} (\hat{P}-P)(s' \mid s,a)\hat{V}_{h+1}^\pi(s')\mathrm{d}s' + \int_{s'} P(s' \mid s,a)(\hat{V}_{h+1}^\pi - V_{h+1}^\pi)(s')\mathrm{d}s' \\
&= \int_{s'} (\hat{P}-P)(s' \mid s,a)(V_{h+1}^\pi + \hat{V}_{h+1}^\pi - V_{h+1}^\pi)\mathrm{d}s' + \int_{s'} P(s' \mid s,a)(\hat{V}_{h+1}^\pi - V_{h+1}^\pi)(s')\mathrm{d}s' \\
&\leq \varepsilon(dH^{3/2} + \sum_{j=1}^{h+1}\varepsilon^j\alpha_{h+1,j}) + \sum_{j=1}^{h+1}\varepsilon^j\alpha_{h+1,j} \\
&\leq \varepsilon(dH^{3/2} + \alpha_{h+1,1}) + \sum_{j=2}^{h+1}\varepsilon^j(\alpha_{h+1,j-1} + \alpha_{h+1,j}) + \varepsilon^{h+2}\alpha_{h+1,h+1}
\end{aligned}$$

Using the fact that $\alpha_{1,1} = dH^{3/2}$ and with a proper induction, we find that:

$$\hat{V}_1^\pi - V_1^\pi \leq \varepsilon dH^{5/2} \frac{1-\varepsilon^{H-h}}{1-\varepsilon} \underset{H\to\infty}{\leq} \varepsilon dH^{5/2}$$

This concludes the proof of the arguments provided in § Planning of Section 5.4.1. This means that the extra regret due to planning with the approximation by RFT features is of order $\mathcal{O}\Big(\varepsilon dH^{5/2}K\Big)$. By choosing an ε of order $1/(H\sqrt{K})$, we deduce that approximating the probability kernel with $\mathcal{O}(pH^2K)$ samples induces a tractable planning procedure without harming the regret.

Remark 5.7. *The reader might be tempted to combine the finite approximation using RFT with algorithms from the linear reinforcement learning literature (Jin et al., 2020). However, note that the dimensionality of the linear space induced by RFT is polynomial in H and K. Consequently, applying algorithms designed with the assumption of linear value function would incur a linear regret.*

Maximum likelihood estimation The ML estimation is explicit for simple distributions like the Gaussian (Rogers and Young, 1977) and for Linearly controlled dynamical systems. But it requires integral approximations for generic transitions as mentioned in (Chowdhury, Gopalan, and Maillard, 2021). However, we believe that this estimation problem is simpler than the planning problem since the latter traditionally involves approximating an integral for all pairs of (s', a).

One of the popular solutions to this estimation are the **Integral approximation techniques.** (Neal, 2001) proposes to handle the MLe using simulated annealing, a method consisting in starting from a tractable distribution and updating it to resemble the distribution at hand. (Vembu, Gartner, and Boley, 2012) proposes MCMC techniques for approximating the partition function. (Carreira-Perpinan and Hinton, 2005) shows that optimizing a different objective, called the contrastive divergence leads to a good approximation of the ML.

Another popular solution to the ML estimation consists in **Score matching.** This is a technique that avoids approximating the partition function and is well studied in literature, see (Jørgensen, 1983). More recently, (Li, Li, et al., 2021) proposed an adaptation of this technique to the exact setting we consider. The latter shows that under certain conditions, that we are unable to verify, the estimation can be solved in $\mathcal{O}(d^3)$ time.

Other works (Shah, Shah, and Wornell, 2021) assume **Bounded distribution support and natural parameter**, and show tha, for a minimally represented k-parameter Exponential family, under boundedness of the support of the distribution and of the natural parameter,

an α-approximation of the MLE can be derived in $\mathcal{O}(\text{poly}(k/\alpha))$ time. The latter assumes a specific definition of compactness of the representation as well as knowledge of the support and shows how to re-parameterize the density to a specific class of exponential families that are easier to study.

Finally, (Dai, Dai, et al., 2019) studies exponential families such that the natural parameter belongs to some RKHS, moreover, it proposes a method that learns the **Kernel parameters** and improves over score matching in time and in memory complexity.

5.5 Regret analysis

We provide a high probability analysis of the regret of `BEF-RLSVI` under standard regularity assumptions of the representation. First we recall the regret definition then we separate the perturbation error from the statistical estimation:

$$\mathcal{R}(K) = \sum_{k=1}^{K} (V^{\star}_{\theta^{\mathrm{p}},\theta^{\mathrm{r}},1} - V^{\pi_k}_{\theta^{\mathrm{p}},\theta^{\mathrm{r}},1})(s_1^k) = \sum_{k=1}^{K} \Big(\underbrace{V^{\star}_{\theta^{\mathrm{p}},\theta^{\mathrm{r}},1} - V^{\pi_k}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},1}}_{learning} + \underbrace{V^{\pi_k}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},1} - V^{\pi_k}_{\theta^{\mathrm{p}},\theta^{\mathrm{r}},1}}_{Estimation} \Big)(s_1^k)$$

For the **estimation error**, we use smoothness arguments with concentrations of parameters up to some novelties. Regarding the **learning error**, we show that the injected noise ensures a constant probability of anti-concentration. Applying Assumption 5.4 and Lemma C.12 leads to the upper-bound.

5.5.1 Estimation error

To show that the estimation error $(\sum_{k=1}^{K} V_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},1} - V^{\pi_k}_{\theta^{\mathrm{p}},\theta^{\mathrm{r}},1})$ can be controlled, we decompose it to an error that comes from the estimation of the transition parameter and one that comes from the estimation of the reward parameter:

$$V^{\pi}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}}}(s_1^k) - V^{\pi}_{\theta^{\mathrm{p}},\theta^{\mathrm{r}}}(s_1^k) = \underbrace{V^{\pi}_{\hat{\theta}^{\mathrm{p}},\theta^{\mathrm{r}}}(s_1^k) - V^{\pi}_{\theta^{\mathrm{p}},\theta^{\mathrm{r}}}(s_1^k)}_{\text{transition estimation}} + \underbrace{V^{\pi}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}}}(s_1^k) - V^{\pi}_{\hat{\theta}^{\mathrm{p}},\theta^{\mathrm{r}}}(s_1^k)}_{\text{reward estimation}}, \tag{5.8}$$

we control each term separately in Section 5.5.1 and Section 5.5.1. Therefore, we obtain the following lemma controlling the estimation error.

Lemma 5.8. *The estimation error satisfies, with probability at least* $1 - 5\delta$

$$\sum_{k=1}^{K} V_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},1}(s_1^k) - V^{\pi}_{\theta^{\mathrm{p}},\theta^{r},1}(s_1^k) \leq 2H\sqrt{\frac{2\beta^{\mathrm{p}}}{\alpha^{\mathrm{p}}}\beta^{\mathrm{p}}(N,\delta)N\gamma_K^{\mathrm{p}}} + 2H\sqrt{2N\log(1/\delta)}$$

$$+\left[\sqrt{KHd\log\left(1+\alpha^{r}\eta^{-1}B_{\varphi,\mathbb{A}}n\right)}+C_d\sqrt{Hd\log(1+\alpha\eta^{-1}B_{\varphi,\mathbf{A}}H)}\right]\times\left(\sqrt{\frac{\beta^{r}(n,\delta)}{2\alpha^{r}}}\right.$$

$$\left.+c\sqrt{(\max_k x_k)d\log(dK/\delta)}\right)\beta^{r}+\sqrt{2KHd\log\left(1+\alpha^{r}\eta^{-1}B_{\varphi,\mathbb{A}}n\right)\log(1/\delta)}$$

where for $i\in[p,r]$, $\beta^{i}(K,\delta)\triangleq\frac{\eta}{2}B_{\mathbf{A}}^2+\gamma_K^{i}+\log(1/\delta)$, *and* $\gamma_K^{i}\triangleq d\log(1+\frac{\beta^{i}}{\eta}B_{\varphi,\mathbf{A}}HK)$. *Also,* $C_d\triangleq\frac{3d}{\log(2)}\log\left(1+\frac{\alpha^{r}\|\mathbf{A}\|_2^2B_{\varphi,\mathbf{A}}^2}{\eta\log(2)}\right)$, *and* c *is a universal constant.*

Proof. It follows directly by combining Lemma 5.9 and Lemma 5.12 using a union bound. □

Transition estimation

The goal of this section is to prove the following lemma which bounds the regret due to transition estimation.

Lemma 5.9. *We have, with probability at least* $1-2\delta$

$$\sum_{k=1}^{K}V_{\hat{\theta}^{p},\theta^{r}}(s_1^k)-V_{\theta^{p},\theta^{r}}^{\pi}(s_1^k)\leq 2H\sqrt{\frac{2\beta^{p}}{\alpha^{p}}\beta^{p}(N,\delta)N\gamma_K^{p}}+2H\sqrt{2N\log(1/\delta)}$$

where $\gamma_K^{p}:=d\log\left(1+\beta^{p}\eta^{-1}B_{\varphi,\mathbb{A}}HK\right)$, *and* $\beta^{p}(K,\delta)\triangleq\frac{\eta}{2}B_{\mathbf{A}}^2+\gamma_K^{p}+\log(1/\delta)$.

Note that in this error term, the reward is bounded since its parameter is exact, the value function is therefore smaller than H. Using the transportation of Lemma C.7 we obtain a bound in terms of kl divergences. We then notice that since the reward parameter is exact, the bound can be improved using Lemma C.12 under Assumption 5.4. We win a $\sqrt{H}$ factor compared to the analysis of (Chowdhury and Gopalan, 2019).

Proof. The proof proceeds in two parts. First, we will reveal a bound in terms of the induced local geometry, *i.e.* a bound in terms of KL-divergence. Second, we explicit the bound by transferring the induced local geometry to the euclidean one.

1) *Bound in terms of local geometry.* We provide a bound on the estimation error of the transition in terms of KL divergences, for that end we show that the estimation error can be decomposed and well controlled. We start by writing the one-step decomposition:

$$
\begin{aligned}
V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},1}(s_1^k) - V^{\pi}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}},1}(s_1^k) & \\
&= \mathbf{E}^{\hat{\theta}^{\mathtt{p}}}_{s_1^k,a_1^k}\left[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},2}\right] - \mathbf{E}^{\theta^{\mathtt{p}}}_{s_1^k,a_1^k}\left[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},2}\right] + \mathbf{E}^{\theta^{\mathtt{p}}}_{s_1^k,a_1^k}[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},2} - V^{\pi}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}},2}] \\
&= \mathbf{E}^{\hat{\theta}^{\mathtt{p}}}_{s_1^k,a_1^k}\left[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},2}\right] - \mathbf{E}^{\theta^{\mathtt{p}}}_{s_1^k,a_1^k}\left[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},2}\right] + V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},2}(s_{2k}) - V^{\pi}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}},2}(s_{2k}) + \zeta_1^k \\
&= \sum_{h=1}^{H} \mathbf{E}^{\hat{\theta}^{\mathtt{p}}}_{s_{hk},a_{hk}}\left[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},h+1}\right] - \mathbf{E}^{\theta^{\mathtt{p}}}_{s_{hk},a_{hk}}\left[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},h+1}\right] + \zeta_{hk}
\end{aligned}
$$

where $\zeta_{hk} = \mathbf{E}^{\theta^{\mathtt{p}}}_{s_{hk},a_{hk}}[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},h+1} - V^{\pi}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}},h+1}] - \left(V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},h+1}(s_{h+1k}) - V^{\pi}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}},h+1}(s_{h+1k})\right)$ is a martingale sequence, and the last equality comes by induction. Here we consider the true reward parameter which verifies $|\mathbf{E}_{\theta^{\mathtt{r}}}[r(s,a)]| \leq 1$ by assumption, therefore $|\zeta_{hk}| \leq 2H$. Using the Azuma-Hoeffding inequality (Boucheron, Lugosi, and Massart, 2013), with probability at least $1-\delta$

$$
\sum_{k=1}^{K}\sum_{h=1}^{H} \zeta_{hk} \leq 2H\sqrt{2KH\log(1/\delta)}
$$

We finish bounding the first term using Lemma C.7, indeed

$$
\begin{aligned}
\mathbf{E}^{\hat{\theta}^{\mathtt{p}}}_{s_{hk},a_{hk}}\left[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},h+1}\right] - \mathbf{E}^{\theta^{\mathtt{p}}}_{s_{hk},a_{hk}}\left[V^{\pi}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},h+1}\right] &\leq H\sqrt{2\,\mathrm{KL}_{s_{hk},a_{hk}}(\theta^{\mathtt{p}},\hat{\theta}^{\mathtt{p}})} \\
&\leq H\min\left\{1, \sqrt{2\,\mathrm{KL}_{s_{hk},a_{hk}}(\theta^{\mathtt{p}},\hat{\theta}^{\mathtt{p}})}\right\},
\end{aligned}
$$

the last inequality follows because $\forall h,\ \mathbf{S}(V_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}},h+1}) \leq H$.

Remark 5.10. *Traditionally, the expected value difference bound follows from the simulation lemma (Ren et al., 2021). The simulation lemma incurs an extra $\sqrt{H}$ factor compared to our bound.*

We deduce that with probability at least $1-\delta$:

$$
\begin{aligned}
&\sum_{k=1}^{K} V_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1^k) - V^{\pi}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1^k) \\
&\qquad \leq H\sum_{k=1}^{K}\min\left\{1, \sum_{h=1}^{H}\sqrt{2\,\mathrm{KL}_{s_{hk},a_{hk}}(\theta^{\mathtt{p}},\hat{\theta}^{\mathtt{p}})}\right\} + 2H\sqrt{2KH\log(1/\delta)} \qquad (5.9)
\end{aligned}
$$

2) *Bounding the sum of KL divergences.* we explicit the bound of Inequality (5.9) using Assumption 5.4 along with properties of the exponential family (*cf* Section C.2.3). We have for all (s,a),

$$
\forall \theta^{\mathtt{p}}, \theta^{\mathtt{p}\prime}, \quad \frac{\alpha^{\mathtt{p}}}{2}\left\|\theta^{\mathtt{p}\prime} - \theta^{\mathtt{p}}\right\|^2_{G_{s,a}} \leq \mathrm{KL}_{s,a}\left(\theta^{\mathtt{p}}, \theta^{\mathtt{p}\prime}\right) \leq \frac{\beta^{\mathtt{p}}}{2}\left\|\theta^{\mathtt{p}\prime} - \theta^{\mathtt{p}}\right\|^2_{G_{s,a}}. \qquad (5.10)
$$

This implies that

$$\mathrm{KL}_{s,a}\left(\hat{\theta}^{\mathtt{p}}(k),\theta^{\mathtt{p}}\right) \leq \frac{\beta^{\mathtt{p}}}{2}\left\|\theta^{\mathtt{p}}-\hat{\theta}^{\mathtt{p}}(k)\right\|^2_{G_{s,a}} \leq \beta^{\mathtt{p}}\left\|(\bar{G}_k^{\mathtt{p}})^{-1/2}G_{s,a}(\bar{G}_k^{\mathtt{p}})^{-1/2}\right\|\frac{1}{2}\left\|\theta^{\mathtt{p}}-\hat{\theta}^{\mathtt{p}}(k)\right\|^2_{\bar{G}_k^{\mathtt{p}}},$$

where $\bar{G}_k^{\mathtt{p}} \equiv \bar{G}^{\mathtt{p}}_{(k-1)H} := G_k + (\alpha^{\mathtt{p}})^{-1}\eta\mathbf{A}$ and $G_k \equiv \sum_{\tau=1}^{k-1}\sum_{h=1}^{H} G_{s_s^\tau,a_h^\tau}$.

From Corollary C.2, with probability at least $1-\delta$ and for all $k \in \mathbf{N}$

$$\left\|\theta^{\mathtt{p}}-\hat{\theta}^{\mathtt{p}}(k)\right\|^2_{\bar{G}_k^{\mathtt{p}}} \leq 2\beta^{\mathtt{p}}(k,\delta)/\alpha^{\mathtt{p}}.$$

Also, using Lemma C.12, we have

$$\sum_{t=1}^{T}\sum_{h=1}^{H}\min\left\{1,\left\|(\bar{G}_k^{\mathtt{p}})^{-1/2}G_{s,a}(\bar{G}_k^{\mathtt{p}})^{-1/2}\right\|\right\} \leq 2d\log\left(1+\alpha^{\mathtt{p}}\eta^{-1}B_{\varphi,\mathbb{A}}HK\right).$$

Combining these two results we obtain, with probability at least $1-\delta$:

$$\sum_{t=1}^{T}\sum_{h=1}^{H}\min\left\{1,\mathrm{KL}_{s_h^t,a_h^t}\left(\hat{\theta}^{\mathtt{p}}(k),\theta^{\mathtt{p}}\right)\right\} \leq \frac{2\beta^{\mathtt{p}}}{\alpha^{\mathtt{p}}}\beta^{\mathtt{p}}(K,\delta)\gamma_K^{\mathtt{p}}. \tag{5.11}$$

Remark 5.11. *Notice that the minimum with* 1 *is crucial, indeed, without it the bound deteriorates by a factor* H *as was the case in* (Chowdhury, Gopalan, and Maillard, 2021).

3) *Combining the bounds.* By applying Cauchy-Schwarz in Inequality (5.9), we obtain, with probability at least $1-\delta$, and for all $K \in \mathbf{N}$

$$\sum_{k=1}^{K} V_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1^k) - V^{\pi}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1^k) \leq H\sqrt{2\sum_{k=1}^{K}\sum_{h=1}^{H}\mathrm{KL}_{s_{hk},a_{hk}}(\theta^{\mathtt{p}},\hat{\theta}^{\mathtt{p}}) + 2H\sqrt{2KH\log(1/\delta)}}.$$

Injecting Inequality (5.11) proves the desired result with probability at least $1-2\delta$. □

Reward estimation

Previous work uses clipping to help control this error, but in this case it can reduce the optimism probability. (Zanette, Brandfonbrener, et al., 2020) proposes an involved clipping depending on the norms $\|(A_i\varphi(s_h^k,a_h^k))_{i\in[d]}\|_{(\bar{G}_k^{\mathtt{p}})^{-1}}$, which is somewhat delicate to analyze and deploy. We remedy the situation acting solely in the proof. We provide the bound over the regret due to estimating the reward parameter in the following lemma.

Lemma 5.12. *With probability at least* $1-3\delta$*, the following result holds true.*

$$\sum_{k=1}^{K} V^{\pi}_{\hat{\theta}^{\mathfrak{p}},\tilde{\theta}^{\mathfrak{r}},1}(s_1^k) - V^{\pi}_{\hat{\theta}^{\mathfrak{p}},\theta^{\mathfrak{r}},1}(s_1^k) \leq \left(\sqrt{\frac{\beta^{\mathfrak{r}}(K,\delta)}{2\alpha^{\mathfrak{r}}}} + c\sqrt{(\max_{k\leq K} x_k) d \log(dK/\delta)}\right)\beta^{\mathfrak{r}} \times \left(\sqrt{C_d\left(1+\frac{\alpha^{\mathfrak{r}} B_{\varphi,A} H}{\eta}\right)} + \sqrt{K\log(e/\delta^2)}\right)\sqrt{Hd\log\left(1+\alpha^{\mathfrak{r}}\eta^{-1}B_{\varphi,\mathbb{A}}HK\right)},$$

where $\beta^{\mathfrak{p}}(K,\delta) \triangleq \frac{\eta}{2}B_{\mathbf{A}}^2 + \gamma_K^{\mathfrak{p}} + \log(1/\delta)$*, and* $\gamma_K^{\mathfrak{p}} \triangleq d\log(1+\frac{\beta^{\mathfrak{p}}}{\eta}B_{\varphi,\mathbf{A}}HK)$*. Also,* c *is a universal constant and* $C_d \triangleq \frac{3d}{\log(2)}\log\left(1+\frac{\alpha^{\mathfrak{r}}\|\mathbf{A}\|_2^2 B_{\varphi,\mathbf{A}}^2}{\eta\log(2)}\right)$.

Proof. The reward estimation error in Equation (5.8) can be written explicitly. Indeed, using Lemma C.11

$$\begin{aligned}
V^{\pi}_{\hat{\theta}^{\mathfrak{p}},\tilde{\theta}^{\mathfrak{r}},1}(s_1^k) - V^{\pi}_{\hat{\theta}^{\mathfrak{p}},\theta^{\mathfrak{r}},1}(s_1^k) &= \mathbf{E}_{(\tilde{s}_h)_{1\leq h\leq H}\sim\pi|\hat{\theta}^{\mathfrak{p}},s_1^k}\left[\sum_{h=1}^{H}\frac{\mathbb{V}\mathrm{ar}_{\tilde{s}_h,\pi(\tilde{s}_h)}(r)}{2}B^\top M_{\tilde{\theta}^{\mathfrak{r}}-\theta^{\mathfrak{r}}}\varphi(\tilde{s}_h,\pi(\tilde{s}_h))\right]\\
&\leq \mathbf{E}\left[\sum_{h=1}^{H}\frac{\mathbb{V}\mathrm{ar}_{\tilde{s}_h,\pi(\tilde{s}_h)}(r)}{2}\|\tilde{\theta}^{\mathfrak{r}}-\theta^{\mathfrak{r}}\|_{\bar{G}_k^{\mathfrak{r}}}\|(B^\top A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathfrak{r}})^{-1}}\right]\\
&\leq \|\tilde{\theta}^{\mathfrak{r}}-\theta^{\mathfrak{r}}\|_{\bar{G}_k^{\mathfrak{r}}}\mathbf{E}\left[\sum_{h=1}^{H}\frac{\mathbb{V}\mathrm{ar}_{\tilde{s}_h,\pi(\tilde{s}_h)}(r)}{2}\|(B^\top A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathfrak{r}})^{-1}}\right]\\
&\leq \|\tilde{\theta}^{\mathfrak{r}}-\theta^{\mathfrak{r}}\|_{\bar{G}_k^{\mathfrak{r}}}\frac{\beta^{\mathfrak{r}}}{2}\mathbf{E}\Bigg[\underbrace{\sum_{h=1}^{H}\|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathfrak{r}})^{-1}}}_{\triangleq\widetilde{\mathrm{traj}}_k}\Bigg],
\end{aligned}$$

where $\mathrm{traj}_k \triangleq \sum_{h=1}^{H}\|(A_i\varphi(s_h,\pi(s_h)))_{1\leq i\leq d}\|_{(G_k^{\mathfrak{r}})^{-1}}$.

Bad rounds. We separate the analysis of this estimation error into bad and good rounds. Here we analyze the bad rounds, which are define by the following set:

$$\mathcal{T} = \{k\in\mathbb{N}^*, \exists h\in[H], \|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathfrak{r}})^{-1}} \geq 1\}$$

These rounds are why clipping is necessary. Thanks to Lemma C.14, we know that the number of such rounds is at most $\mathcal{O}(d)$. Surprisingly, it depends neither on H nor on K.

We now show that the "bad rounds" incur at most $\mathcal{O}\left(d^{3/2}H^2\right)$ regret, independent of K. Thus, we can omit clipping for free.

Remark 5.13. *In non-episodic settings, the forward algorithm (Azoury and Warmuth, 2001) eliminates the span control issue. See (Ouhamma, Maillard, and Perchet, 2021) for stochastic analysis and an application to linear bandits.*

1) We know that $\|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d}(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d}^\top\|_2^2 \le \|\mathbf{A}\|_2^2 B_{\varphi,\mathbf{A}}^2$. Then, according to Lemma C.14

$$|\mathcal{T}| \le \frac{3d}{\log(2)}\log\left(1+\frac{\alpha\|\mathbf{A}\|_2^2 B_{\varphi,\mathbf{A}}^2}{\eta\log(2)}\right).$$

2) Since G_k is positive semi-definite, we have $\bar{G}_k^{\mathtt{r}} \succeq (\alpha^{\mathtt{r}})^{-1}\eta\mathbf{A}$, and in turn, for all state-action couples (s,a), $\left\|(\bar{G}_k^{\mathtt{r}})^{-1}G_{s,a}\right\| \le \frac{\alpha^{\mathtt{r}}}{\eta}\left\|\mathbf{A}^{-1}G_{s,a}\right\| \le \frac{\alpha^{\mathtt{r}}B_{\varphi,\mathbf{A}}}{\eta}$.

This further yields

$$\left\|I+(\bar{G}_k^{\mathtt{r}})^{-1}\sum_{h=1}^{H}G_{s_h^t,a_h^t}\right\| \le 1+\sum_{h=1}^{H}\left\|(\bar{G}_k^{\mathtt{r}})^{-1}G_{s_h^t,a_h^t}\right\| \le 1+\frac{\alpha^{\mathtt{r}}B_{\varphi,\mathbf{A}}H}{\eta}.$$

Let us define $\bar{G}_{k+H}^{\mathtt{r}} := \bar{G}_k^{\mathtt{r}} + \sum_{h=1}^{H}G_{s_h^k,a_h^k}$. Then,

$$\bar{G}_{k+H}^{-1}G_{s,a} = \left(I+(\bar{G}_k^{\mathtt{r}})^{-1}\sum_{h=1}^{H}G_{s_h^t,a_h^t}\right)^{-1}(\bar{G}_k^{\mathtt{r}})^{-1}G_{s,a}.$$

Therefore, for all pairs (s,a),

$$\begin{aligned}
\|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d}\|_{(\bar{G}_k^{\mathtt{r}})^{-1}} &= \sqrt{{}^\top((A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d}^\top(\bar{G}_k^{\mathtt{r}})^{-1}(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d})} \\
&= \sqrt{{}^\top\left(\left(1+\frac{\alpha^{\mathtt{r}}B_{\varphi,A}H}{\eta}\right)(\bar{G}_{k+H}^{\mathtt{r}})^{-1}G_{s,a}\right)} \\
&\le \sqrt{\left(1+\frac{\alpha^{\mathtt{r}}B_{\varphi,A}H}{\eta}\right)}\|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d}\|_{(\bar{G}_{k+H}^{\mathtt{r}})^{-1}}
\end{aligned}$$

Since $\|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d}\|_{(\bar{G}_{k+H}^{\mathtt{r}})^{-1}} \le 1$, we have

$$\|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d}\|_{(\bar{G}_{k+H}^{\mathtt{r}})^{-1}} \le \min\left\{1, \|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d}\|_{(\bar{G}_k^{\mathtt{r}})^{-1}}\right\}.$$

Consequently

$$\sum_{h=1}^{H}\|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\le i\le d}\|_{(\bar{G}_{k+H}^{\mathtt{r}})^{-1}} \le \sqrt{Hd\log(1+\alpha^{\mathtt{r}}\eta^{-1}B_{\varphi,\mathbf{A}}H)}.$$

3) From *1*) and *2*), we deduce that the total regret induced by rounds from $\mathcal{T}$ is bounded.

$$\sum_{k\in\mathcal{T}}\sum_{h\in[H]} V^{\pi}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},1}(s_1^k) - V^{\pi}_{\hat{\theta}^{\mathrm{p}},\theta^{\mathrm{r}},1}(s_1^k) \leq \|\tilde{\theta}^{\mathrm{r}} - \theta^{\mathrm{r}}\|_{\bar{G}_k^{\mathrm{r}}} \frac{\beta^{\mathrm{r}}}{2} \sqrt{\frac{3d}{\log(2)}\log\left(1+\frac{\alpha^{\mathrm{r}}\|\mathbf{A}\|_2^2 B_{\varphi,\mathbf{A}}^2}{\eta\log(2)}\right)\left(1+\frac{\alpha^{\mathrm{r}}B_{\varphi,A}H}{\eta}\right) Hd\log(1+\alpha^{\mathrm{r}}\eta^{-1}B_{\varphi,\mathbf{A}}H)} \tag{5.12}$$

Remark 5.14. *The bad rounds analysis is one of our most important contributions as it enables us to forgo clipping without consequences. Consequently, this is a novel method to control the reward estimation error that improves on existing work for whom clipping was essential.*

Good rounds. Going forward we consider rounds from $\bar{\mathcal{T}}$. Let us define

$$\zeta_k' \triangleq \mathrm{traj}_k - \mathbf{E}_{(\tilde{s}_h)_{1\leq h\leq H}\sim\pi|\hat{\theta}^{\mathrm{p}},s_1^k}\left[\widetilde{\mathrm{traj}}_k\right].$$

where $\widetilde{\mathrm{traj}}_k$ is the same quantity as traj but with a random realization of state transitions. Since all feature norms are smaller than one, $(\zeta_k')_k$ is a martingale sequence with $|\zeta_k'| \leq \sqrt{Hd\log\left(1+\alpha^{\mathrm{r}}\eta^{-1}B_{\varphi,\mathbb{A}}HK\right)}$. We deduce that with probability at least $1-\delta$:

$$\sum_{k=1}^{K}\zeta_k' \leq \sqrt{2KHd\log\left(1+\alpha^{\mathrm{r}}\eta^{-1}B_{\varphi,\mathbb{A}}HK\right)\log(1/\delta)}$$

Therefore, we have with probability at least $1-3\delta$:

$$\sum_{k\in\mathcal{T}^{\mathrm{c}}} V^{\pi}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},1}(s_1^k) - V^{\pi}_{\hat{\theta}^{\mathrm{p}},\theta^{\mathrm{r}},1}(s_1^k) \leq \left(\sqrt{\frac{\beta^{\mathrm{r}}(K,\delta)}{2\alpha^{\mathrm{r}}}} + c\sqrt{(\max_k x_k)d\log(dK/\delta)}\right) \times \beta^{\mathrm{r}}\sqrt{KHd\log\left(1+\alpha^{\mathrm{r}}\eta^{-1}B_{\varphi,\mathbb{A}}KH\right)\log(e/\delta^2)}.$$

The last inequality follows from controlling the concentration of the reward parameter. First we observe that (Corollary C.4) with probability at least $1-\delta$, uniformly over $k\in\mathbf{N}$, $\left\|\theta^{\mathrm{r}} - \hat{\theta}^{\mathrm{r}}(k)\right\|^2_{\bar{G}_k^{\mathrm{r}}} \leq \frac{2}{\alpha^{\mathrm{r}}}\beta^{\mathrm{r}}(k,\delta)$. Second, we also have that for all $k\geq 1$, with probability at least $1-\delta$, $\|\xi_k\|_{G_k^{\mathrm{r}}} \leq c\sqrt{x_k d\log(d/\delta)}$, we then use a union bound. Combining with Equation (5.12) we find

$$\sum_{k=1}^{K} V^{\pi}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},1}(s_1^k) - V^{\pi}_{\hat{\theta}^{\mathrm{p}},\theta^{\mathrm{r}},1}(s_1^k) \leq \left(\sqrt{\frac{\beta^{\mathrm{r}}(K,\delta)}{2\alpha^{\mathrm{r}}}} + c\sqrt{(\max_k x_k)d\log(dK/\delta)}\right)$$

$$\times \beta^{r}\sqrt{KHd\log\left(1+\alpha^{r}\eta^{-1}B_{\varphi,\mathbb{A}}HK\right)\log(e/\delta^2)}.$$

This concludes the proof of the reward estimation error. □

Remark 5.15. *If we use Lemma C.11 without the martingale difference sequence, it will lead to a linear regret. Indeed, the span of the sum of norms over an episode is of order $\sqrt{H}$. Using the martingale technique instead allows us to retrieve a telescopic sum controlled using the elliptical lemma, this is essential to obtaining a sub-linear regret bound.*

5.5.2 Learning error

We now start the control of an important regret term, due to the distance between the estimated value function and the optimal value function. Our main methodology here is to show that the estimated value is optimistic with a constant probability and that this suffices to control the error.

Lemma 5.16. *If the variance parameter of the injected noise $(\xi_k)_k$ satisfies*

$$x_k \geq \left(H\sqrt{\frac{\beta^{p}\beta^{p}(k,\delta)}{\alpha^{p}\alpha^{r}}}+\frac{\sqrt{\beta^{r}\beta^{r}(k,\delta)\min\{1,\frac{\alpha^{p}}{\alpha^{r}}\}}}{2\alpha^{r}}\right),$$

then the learning error is controlled with probability at least $1-2\delta$ as

$$\sum_{k=1}^{K} V_1^{\star}(s_1^k)-V^{\pi}_{\hat{\theta}^{p},\hat{\theta}^{r}+\bar{\xi}_k,1}(s_1^k) \leq \frac{d\beta^{r}\sqrt{x_k}\left(1+\sqrt{\log(d/\delta)}\right)}{\Phi(-1)}\sqrt{H\log\left(1+\alpha^{r}\eta^{-1}B_{\varphi,\mathbb{A}}HK\right)} \times\left(\sqrt{C_d\left(1+\frac{\alpha^{r}B_{\varphi,A}H}{\eta}\right)}+\sqrt{K\log(e/\delta^2)}\right),$$

where for $i\in[p,r]$, $\beta^{i}(K,\delta)\triangleq\frac{\eta}{2}B_{\mathbf{A}}^2+\gamma_K^{i}+\log(1/\delta)$, and $\gamma_K^{i}\triangleq d\log(1+\frac{\beta^{i}}{\eta}B_{\varphi,\mathbf{A}}HK)$. Also $C_d\triangleq\frac{3d}{\log(2)}\log\left(1+\frac{\alpha^{r}\|\mathbf{A}\|_2^2B_{\varphi,\mathbf{A}}^2}{\eta\log(2)}\right)$, and Φ is the normal CDF.

This result basically means that we are no longer obliged to follow optimistic value functions, the perturbed estimation is enough to have a tight bound on the learning error.

Stochastic optimism

The goal here is to show that by injecting our carefully designed noise in the rewards we can ensure optimism with a constant probability. Consider the optimal policy $\pi^\star$, we have:

$$\begin{aligned}(V_{\hat{\theta}^{\mathtt{p}},\tilde{\theta}^{\mathtt{r}},1} - V^\star_{\theta^{\mathtt{p}},\theta^{\mathtt{r}},1})(s_1) &\geq (Q^\star_{\hat{\theta}^{\mathtt{p}},\tilde{\theta}^{\mathtt{r}},1} - Q^\star_1)(s_1,\pi^\star(s_1)) \\ &\geq \underbrace{V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1) - V^{\pi^\star}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1)}_{\text{first term}} + \underbrace{V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\hat{\theta}^{\mathtt{r}}}(s_1) - V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1)}_{\text{second term}} + \underbrace{V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\tilde{\theta}^{\mathtt{r}}}(s_1) - V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\hat{\theta}^{\mathtt{r}}}(s_1)}_{\text{third term}}\end{aligned}$$

The first and second terms are perturbation free, we handle them like the estimation error, by concentration arguments for $\hat{\theta}^{\mathtt{p}}$ and $\hat{\theta}^{\mathtt{r}}$. For the third term, using transportation of rewards (Lemma C.11) and anti-concentration of ξ_k (Lemma C.6).

First term. By assumption, the expected value of the reward following the true parameter satisfies $\mathbf{E}_{\theta^{\mathtt{r}}}[r(s,a)] \in [0,1]$, then $\mathbf{S}\left(\sum_{t=1}^H \mathbf{E}_{\theta^{\mathtt{r}}}[r(s_t,\pi(s_t))]\right) \leq H$. Consequently, the first term can be controlled using Lemma C.7

$$\begin{aligned}V^{\pi^\star}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1) - V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1) &\leq H\sqrt{\mathrm{KL}(P_{\hat{\theta}^{\mathtt{p}}}(s_2,\dots,s_H), P_{\theta^{\mathtt{p}}}(s_2,\dots,s_H))} \\ &\leq H\sqrt{\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\sum_{t=1}^H \psi(\tilde{s}_{t+1})^\top M_{\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}}}\varphi(\tilde{s}_t,\pi^\star(\tilde{s}_t)) + Z^{\mathtt{p}}_{\theta^{\mathtt{p}}}(\tilde{s}_t,\pi^\star(\tilde{s}_t)) - Z^{\mathtt{p}}_{\hat{\theta}^{\mathtt{p}}}(\tilde{s}_t,\pi^\star(\tilde{s}_t))\right]}\end{aligned}$$

Using Taylor's expansion, for all $h \in [H], \exists \theta_h \in [\theta^{\mathtt{p}}, \hat{\theta}^{\mathtt{p}}]$ such that:

$$\begin{aligned}&\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\psi(\tilde{s}_{t+1})^\top M_{\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}}}\varphi(\tilde{s}_t,\pi^\star(\tilde{s}_t)) + Z^{\mathtt{p}}_{\theta^{\mathtt{p}}}(\tilde{s}_t,\pi^\star(\tilde{s}_t)) - Z^{\mathtt{p}}_{\hat{\theta}^{\mathtt{p}}}(\tilde{s}_t,\pi^\star(\tilde{s}_t))\right] \\ &\quad= \frac{1}{2}(\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}})^\top \mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\nabla^2_{s_h,\pi^\star(s_h)} Z^{\mathtt{p}}(\theta_h)\right](\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}}) \\ &\quad\leq \frac{\beta^{\mathtt{p}}}{2}\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\|\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}}\|^2_{G_{\tilde{s}_h,\pi^\star(\tilde{s}_h)}}\right].\end{aligned}$$

Define $u_k \triangleq \sum_{h=1}^H \mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[(A_i\varphi(\tilde{s}_h,\pi^\star(\tilde{s}_h)))_{i\in[d]}\right]$, then

$$\begin{aligned}V^{\pi^\star}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1) - V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1) &\leq H\sqrt{\frac{\beta^{\mathtt{p}}}{2}\sum_{h=1}^H \mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\|\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}}\|^2_{G_{\tilde{s}_h,\pi^\star(\tilde{s}_h)}}\right]} \\ &\leq H\sqrt{\frac{\beta^{\mathtt{p}}}{2}}\left\|\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}}\right\|_{\sum_{h=1}^H \mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[G_{\tilde{s}_h,\pi^\star(\tilde{s}_h)}\right]} \\ &\leq H\sqrt{\frac{\beta^{\mathtt{p}}}{2}}\left\|\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}}\right\|_{u_k u_k^\top} \\ &\leq H\sqrt{\frac{\beta^{\mathtt{p}}}{2}\left\|(\bar{G}^{\mathtt{p}}_k)^{-1/2}u_k u_k^\top (\bar{G}^{\mathtt{p}}_k)^{-1/2}\right\|}\|\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}}\|_{\bar{G}^{\mathtt{p}}_k}\end{aligned}$$

$$\leq H\sqrt{\frac{\beta^{\mathtt{p}}}{2}}\|u_k\|_{(\bar{G}_k^{\mathtt{p}})^{-1}}\|\hat{\theta}^{\mathtt{p}}-\theta^{\mathtt{p}}\|_{\bar{G}_k^{\mathtt{p}}}$$

The third line follows because $\forall x\in\mathbf{R}^d,\quad \|x\|_{\sum_{i=1}a_ia_i^\top}\leq\|x\|_{(\sum_{i=1}a_i)(\sum_{i=1}a_i)^\top}$, and the last one follows because ${}^\top(AB)\leq^\top(A)^\top(B)$ for any two real positive semi-definite matrices A and B.
We deduce, with probability at least $1-\delta$:

$$V^{\pi^\star}_{\theta^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1)-V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1)\leq H\sqrt{\frac{\beta^{\mathtt{p}}\beta^{\mathtt{p}}(k,\delta)}{\alpha^{\mathtt{p}}}}\left\|\sum_{h=1}^{H}\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[(A_i\varphi(\tilde{s}_h,\pi^\star(\tilde{s}_h)))_{i\in[d]}\right]\right\|_{(\bar{G}_k^{\mathtt{p}})^{-1}}$$

Second term. We have

$$\begin{aligned}V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\hat{\theta}^{\mathtt{r}}}(s_1)-V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\theta^{\mathtt{r}}}(s_1)&=\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\sum_{t=1}^{H}\frac{\mathbb{V}\mathrm{ar}^{\theta_t^{\mathtt{r}}}(r)}{2}B^\top M_{\hat{\theta}^{\mathtt{r}}-\theta^{\mathtt{r}}}\varphi(\tilde{s}_t,\pi^\star(\tilde{s}_t))\right]\\&=(\hat{\theta}^{\mathtt{r}}-\theta^{\mathtt{r}})^\top\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\sum_{t=1}^{H}\frac{\mathbb{V}\mathrm{ar}^{\theta_t^{\mathtt{r}}}(r)}{2}(A_i\varphi(\tilde{s}_t,\pi^\star(\tilde{s}_t)))_{i\in[d]}\right]B\\&\leq\frac{\sqrt{\beta^{\mathtt{r}}}}{2}\|\hat{\theta}^{\mathtt{r}}-\theta^{\mathtt{r}}\|_{\bar{G}_k^{\mathtt{r}}}\left\|\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\sum_{t=1}^{H}(A_i\varphi(\tilde{s}_t,\pi^\star(\tilde{s}_t)))_{i\in[d]}\right]\right\|_{(\bar{G}_k^{\mathtt{r}})^{-1}}\end{aligned}$$

The last inequality comes from Cauchy-Schwarz. Applying that the norm (sum) makes appear only symmetric matrices times the variances so that we can bound the latter by $\beta^{\mathtt{r}}$.
We conclude that with probability at least $1-\delta$,

$$V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\hat{\theta}^{\mathtt{r}}}(s_1)-V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\tilde{\theta}^{\mathtt{r}}}(s_1)\leq\frac{\beta^{\mathtt{r}}\sqrt{\beta^{\mathtt{r}}(k,\delta)}}{\sqrt{2\alpha^{\mathtt{r}}}}\left\|\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\sum_{t=1}^{H}(A_i\varphi(\tilde{s}_t,\pi^\star(\tilde{s}_t)))_{i\in[d]}\right]\right\|_{(\bar{G}_k^{\mathtt{r}})^{-1}}$$

We want to write all the norms in the same matrix. Therefore, with probability at least $1-\delta$,

$$\begin{aligned}V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\hat{\theta}^{\mathtt{r}}}(s_1)-V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\tilde{\theta}^{\mathtt{r}}}(s_1)\leq&\sqrt{\frac{\beta^{\mathtt{r}}\beta^{\mathtt{r}}(k,\delta)\min\{1,\frac{\alpha^{\mathtt{p}}}{\alpha^{\mathtt{r}}}\}}{2\alpha^{\mathtt{r}}}}\\&\times\left\|\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\sum_{t=1}^{H}(A_i\varphi(\tilde{s}_t,\pi^\star(\tilde{s}_t)))_{i\in[d]}\right]\right\|_{(\bar{G}_k^{\mathtt{p}})^{-1}}\end{aligned}$$

Third term. We have

$$\begin{aligned}V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\hat{\theta}^{\mathtt{r}},1}(s_1)-V^{\pi^\star}_{\hat{\theta}^{\mathtt{p}},\tilde{\theta}^{\mathtt{r}},1}(s_1)&=\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\sum_{t=1}^{H}\frac{\mathbb{V}\mathrm{ar}^{\theta_j^{\mathtt{r}}}(r)}{2}B^\top M_{\hat{\theta}^{\mathtt{r}}-\tilde{\theta}^{\mathtt{r}}}\varphi(\tilde{s}_t,\pi^\star(\tilde{s}_t))\right]\\&=\xi_k^\top\,\mathbf{E}_{(\tilde{s}_t)_{t\in[H]}\sim\hat{\theta}^{\mathtt{p}}|s_1^k}\left[\sum_{t=1}^{H}\frac{\mathbb{V}\mathrm{ar}^{\theta_j^{\mathtt{r}}}(r)}{2}(A_i\varphi(\tilde{s}_t,\pi^\star(\tilde{s}_t)))_{i\in[d]}\right]B\end{aligned}$$

Given the normal CDF Φ, we obtain that with probability at least $\Phi(-1)$

$$V^{\pi^\star}_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}}(s_1) - V^{\pi^\star}_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}}}(s_1) \geq \sqrt{x_k \alpha^{\mathrm{r}}} \left\| \left[\sum_{t=1}^{H} \frac{\mathbb{V}\mathrm{ar}^{\theta^{\mathrm{r}}_j}(r)}{2} (A_i \varphi(\tilde{s}_t, \pi^\star(\tilde{s}_t)))_{i\in[d]} \right] \right\|_{(\bar{G}^{\mathrm{p}}_k)^{-1}}$$

Choosing $x_k \geq \left(H\sqrt{\frac{\beta^{\mathrm{p}}\beta^{\mathrm{p}}(k,\delta)}{\alpha^{\mathrm{p}}\alpha^{\mathrm{r}}}} + \frac{\sqrt{\beta^{\mathrm{r}}\beta^{\mathrm{r}}(k,\delta)\min\{1, \frac{\alpha^{\mathrm{p}}}{\alpha^{\mathrm{r}}}\}}}{2\alpha^{\mathrm{r}}} \right)$ and using Lemma C.6, we find that the perturbed value function is optimistic with probability at least $\Phi(-1)$.

Controlling the learning error

In this section we see the core difference with optimistic algorithms. On the one hand, optimistic approaches require the value function generating the agent's policy to be larger than the optimal one with large probability, and can therefore ensure that the learning error is negative. On the other hand, `BEF-RLSVI` only ensures that the value function is optimistic with a constant probability: intuitively when this event holds the learning happens, and if it does not then the policy is still close to a good one thanks to the decreasing estimation error.

Upper bound on $V_1^\star$. Let us draw $(\bar{\xi}_k)_{k\in[K]}$ i.i.d copies of $(\xi_k)_{k\in[K]}$. Define the optimism event at episode k:

$$\bar{O}_k = \{V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k) - V_1^\star(s_1^k) \geq 0\} \tag{5.13}$$

we know that $\mathbf{P}(\bar{O}_k) \geq \Phi(-1)$. This event provides the upper bound:

$$V_1^\star(s_1^k) \leq \mathbf{E}_{\bar{\xi}_k|\bar{O}_k}[V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k)] \tag{5.14}$$

Lower bound on $V_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}}}$. We define this bound with an optimization problem under concentration of the noise. Consider $\underline{\mathrm{V}}_1(s_1^k)$ is the solution of

$$\begin{aligned} &\min_{\xi_k} V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\xi_k,1}(s_{1^k}) \\ &\|\xi_k\|_{\bar{G}^{\mathrm{p}}_k} \leq \sqrt{x_k d \log(d/\delta)}, \quad \forall t \in [H] \end{aligned} \tag{5.15}$$

Under the concentration of our injected noise, we obtain

$$\underline{\mathrm{V}}_1(s_1^k) \leq V_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}}}(s_1^k) \tag{5.16}$$

Combining the error bounds. Combining the upper bound of Equation (5.14) with the lower bound of Equation (5.16), we get, with probability at least $1-\delta$:

$$V_1^\star(s_1^k) - V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k) \leq \mathbf{E}_{\bar{\xi}_k|\bar{O}_k}[V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k) - \underline{V}_1(s_1^k)]$$

Also, using the tower rule,

$$\begin{aligned}
&\mathbf{E}_{\bar{\xi}_k}[V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k) - \underline{V}_1(s_1^k)] \\
&\quad = \mathbf{E}_{\bar{\xi}_k|\bar{O}_k}[V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k) - \underline{V}_1(s_1^k)]\mathbf{P}(\bar{O}_k) + \mathbf{E}_{\bar{\xi}_k|\bar{O}_k^{\mathrm{c}}}[V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k) - \underline{V}_1(s_1^k)]\mathbf{P}(\bar{O}_k^{\mathrm{c}})
\end{aligned}$$

Therefore,

$$\begin{aligned}
&V_1^\star(s_1^k) - V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k) \\
&\quad \leq \left(\mathbf{E}_{\bar{\xi}_k}[V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k) - \underline{V}_1(s_1^k)] - \mathbf{E}_{\bar{\xi}_k|\bar{O}_k^{\mathrm{c}}}[V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\bar{\xi}_k,1}(s_1^k) - \underline{V}_1(s_1^k)]\mathbf{P}(\bar{O}_k^{\mathrm{c}})\right)/\mathbf{P}(\bar{O}_k) \\
&\quad = \left(\mathbf{E}_{\xi_k}[V^\pi_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\xi_k,1}(s_1^k) - \underline{V}_1^\pi(s_1^k)] - \mathbf{E}_{\xi_k|\bar{O}_k^{\mathrm{c}}}[V_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\xi_k,1}(s_1^k) - \underline{V}_1(s_1^k)]\mathbf{P}(\bar{O}_k^{\mathrm{c}})\right)/\mathbf{P}(\bar{O}_k).
\end{aligned}$$

The last line follows since ξ_k and $\bar{\xi}_k$ are i.i.d.

The rest of the analysis proceeds similarly to the proof of the reward estimation.

Let us call the argument of the minimum in Equation (5.15) as $\underline{\xi}_k$. Using Lemma C.11, we find

$$\begin{aligned}
&V^\pi_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},1}(s_1^k) - V^\pi_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\underline{\xi}_k,1}(s_1^k) \\
&\qquad = \mathbf{E}_{(\tilde{s}_h)_{1\leq h\leq H}\sim\pi|\hat{\theta}^{\mathrm{p}},s_1^k}\left[\sum_{h=1}^H \frac{\mathbb{V}\mathrm{ar}_{\tilde{s}_h,\pi(\tilde{s}_h)}(r)}{2} B^\top M_{\tilde{\theta}^{\mathrm{r}}-\hat{\theta}^{\mathrm{r}}-\underline{\xi}_k}\varphi(\tilde{s}_h,\pi(\tilde{s}_h))\right] \\
&\qquad \leq \mathbf{E}\left[\sum_{h=1}^H \frac{\mathbb{V}\mathrm{ar}_{\tilde{s}_h,\pi(\tilde{s}_h)}(r)}{2}\|\tilde{\theta}^{\mathrm{r}}-\hat{\theta}^{\mathrm{r}}-\underline{\xi}_k\|_{\bar{G}_k^{\mathrm{p}}}\|(B^\top A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathrm{p}})^{-1}}\right] \\
&\qquad \leq \|\tilde{\theta}^{\mathrm{r}}-\hat{\theta}^{\mathrm{r}}-\underline{\xi}_k\|_{\bar{G}_k^{\mathrm{p}}}\mathbf{E}\left[\sum_{h=1}^H \frac{\mathbb{V}\mathrm{ar}_{\tilde{s}_h,\pi(\tilde{s}_h)}(r)}{2}\|(B^\top A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathrm{p}})^{-1}}\right] \\
&\qquad \leq \|\tilde{\xi}_k-\underline{\xi}_k\|_{\bar{G}_k^{\mathrm{p}}}\frac{\beta^{\mathrm{r}}}{2}\mathbf{E}\left[\sum_{h=1}^H \|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathrm{p}})^{-1}}\right]
\end{aligned}$$

Then,

$$\begin{aligned}
&\mathbf{E}_{\tilde{\xi}_k}\left[V^\pi_{\hat{\theta}^{\mathrm{p}},\tilde{\theta}^{\mathrm{r}},1}(s_1^k) - V^\pi_{\hat{\theta}^{\mathrm{p}},\hat{\theta}^{\mathrm{r}}+\underline{\xi}_k,1}(s_1^k)\right] \\
&\qquad \leq \frac{\beta^{\mathrm{r}}}{2}\mathbf{E}_{\tilde{\xi}_k}[\|\tilde{\xi}_k-\underline{\xi}_k\|_{\bar{G}_k^{\mathrm{p}}}]\mathbf{E}_{(\tilde{s}_h)\sim\pi|\hat{\theta}^{\mathrm{p}}}\left[\sum_{h=1}^H \|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathrm{p}})^{-1}}\right].
\end{aligned}$$

Also,

$$\begin{aligned}\left|\mathbf{E}_{\xi_k|\bar{O}_k^{\mathtt{c}}}[V_{\hat{\theta}^{\mathtt{p}},\hat{\theta}^{\mathtt{r}}+\xi_k,1}(s_1^k)-\underline{V}_1(s_1^k)]\right| \\ \leq \frac{\beta^{\mathtt{r}}}{2}\mathbf{E}_{\tilde{\xi}_k|\bar{O}_k^{\mathtt{c}}}[\|\tilde{\xi}_k-\underline{\xi}_k\|_{\bar{G}_k^{\mathtt{p}}}]\mathbf{E}_{(\tilde{s}_h)\sim\pi|\hat{\theta}^{\mathtt{p}}}\left[\sum_{h=1}^{H}\|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathtt{p}})^{-1}}\right] \\ \leq \frac{\beta^{\mathtt{r}}}{2}\mathbf{E}_{\tilde{\xi}_k}[\|\tilde{\xi}_k-\underline{\xi}_k\|_{\bar{G}_k^{\mathtt{p}}}]\mathbf{E}_{(\tilde{s}_h)\sim\pi|\hat{\theta}^{\mathtt{p}}}\left[\sum_{h=1}^{H}\|(A_i\varphi(\tilde{s}_h,\pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^{\mathtt{p}})^{-1}}\right].\end{aligned}$$

We have a bound on the expected value of the sum of feature norms in the proof of Lemma 5.12. Also,

$$\begin{aligned}\mathbf{E}_{\tilde{\xi}_k}[\|\tilde{\xi}_k-\underline{\xi}_k\|_{\bar{G}_k^{\mathtt{p}}}] &\leq \mathbf{E}_{\tilde{\xi}_k}[\|\tilde{\xi}_k\|_{\bar{G}_k^{\mathtt{p}}}]+\mathbf{E}_{\tilde{\xi}_k}[\|\underline{\xi}_k\|_{\bar{G}_k^{\mathtt{p}}}] \\ &\leq \sqrt{\mathbf{E}_{\tilde{\xi}_k}[\|\tilde{\xi}_k\|^2_{\bar{G}_k^{\mathtt{p}}}]}+\sqrt{x_k d\log(d/\delta)} \\ &\leq \sqrt{x_k d}+\sqrt{x_k d\log(d/\delta)}\end{aligned}$$

The second line follows from Cauchy-Schwarz and by definition of $\underline{\xi}_k$. The last line is due to the fact that $x_k(\bar{G}_k^{\mathtt{p}})^{-1}\sim\mathcal{N}(0,x_kI_d)$, which implies $\|\tilde{\xi}_k\|^2_{\bar{G}_k^{\mathtt{p}}}\sim\mathcal{N}(0,dx_k)$. We conclude the proof by taking the sum of feature norms from the proof of Lemma 5.12.

We conclude that with probability at least $1-2\delta$:

$$\begin{aligned}\sum_{k=1}^{K}V_1^\star(s_1^k)-V_{\hat{\theta}^{\mathtt{p}},\hat{\theta}^{\mathtt{r}}+\bar{\xi}_k,1}(s_1^k) \leq \frac{\beta^{\mathtt{r}}}{\Phi(-1)}(\sqrt{x_k d}+\sqrt{x_k d\log(d/\delta)}) \\ \left[\sqrt{\frac{3d}{\log(2)}\log\left(1+\frac{\alpha^{\mathtt{r}}\|\mathbf{A}\|_2^2B^2_{\varphi,\mathbf{A}}}{\eta\log(2)}\right)\left(1+\frac{\alpha^{\mathtt{r}}B_{\varphi,A}H}{\eta}\right)Hd\log(1+\alpha^{\mathtt{r}}\eta^{-1}B_{\varphi,\mathbf{A}}H)}\right. \\ \left.+\sqrt{KHd\log\left(1+\alpha^{\mathtt{r}}\eta^{-1}B_{\varphi,\mathbb{A}}HK\right)\log(e/\delta^2)}\right].\end{aligned}$$

5.6 Related works: functional representations of MDPs with regret and tractability

Our work extends the endeavor of using functional representations for regret minimization in continuous state-action MDPs. Now, we posit our contributions in existing literature.

Kernel value function representation. (**pmlr-v119-ayoub20a**) studies MDPs with a linear mixtures model then extends to an RKHS setting, this generalizes our work and that of (Yang and Wang, 2020). However, the paper proposes an Eluder-dimension analysis, for RKHS settings this leads to the result of (Yang and Wang, 2020), *i.e.* a regret $H\log(T)^d$ higher than

for BEF-RLSVI.Recently, (Huang et al., 2021) shows that for RKHS, Eluder dimension and the information gain are strictly equivalent, which brings in the extra factor.

General functional representation. The Eluder dimension is a complexity measure often used to analyze RL with general function space, (Huang et al., 2021) asserts that "common examples of where it is known to be small are function spaces (vector spaces)". (Dai, Shaw, et al., 2018) provides the first convergence guarantee for general nonlinear function representations in the Maximum Entropy RL setting, where entropy of a policy is used as a regularizer to induce exploration. Thus, the analysis cannot address episodic RL, where we have to explicitly ensure exploration with optimism. In the episodic setting, (Wang, Salakhutdinov, and Yang, 2020) leverage the UCB approach for tabular MDPs and function spaces with bounded Eluder dimension, this strategy achieves a and achieve a $\tilde{O}\left(\sqrt{d^4H^2T}\right)$ regret for linear MDPs. (Ishfaq et al., 2021) considers the same setting, proposes an RLSVI based algorithm, and achieves a $\tilde{O}(\sqrt{d^3H^4K})$ for linear MDPs. However, the latter assumes an oracle perturbing the estimation to achieve anti-concentration while maintaining a bounded covering number, which is a counter-intuitive mix of boundedness and anti-concentration. Indeed, (Zanette, Brandfonbrener, et al., 2020) studied the linear MDP case, and while it managed to design an ingenious clipping verifying previous assumptions, the method is extremely intricate and the proof is involved and unlikely to extend for general value function spaces. *To concertize our design, we focus on the general but explicit BEF of MDPs than any abstract representation. We also remove the requirement to clip with a novel analysis.*

Bilinear exponential family of MDPs. Exponential families are studied widely in RL theory, from bandits to MDPs (Lu, Meisami, and Tewari, 2021; Korda, Kaufmann, and Munos, 2013; Filippi et al., 2010; Kveton and Hauskrecht, 2006), as an expressive parametric family to design theoretically-grounded model-based algorithms. (Chowdhury, Gopalan, and Maillard, 2021) first studies episodic RL with Bilinear Exponential Family (BEF) of transitions, which is linear in both state-action pairs and the next-state. It proposes a regularized log-likelihood method to estimate the model parameters, and two optimistic algorithms with upper confidence bounds and posterior sampling. Due to its generality to unifiedly model tabular MDPs, factored MDPs, and linearly controlled dynamical systems, the BEF-family of MDPs has received increasing attention (Li, Li, et al., 2021). (Li, Li, et al., 2021) estimates the model parameters based on score matching that enables them to replace regularity assumption on the log-partition function with Fisher-information and assumption on the parameters. Both (Chowdhury, Gopalan, and Maillard, 2021; Li, Li, et al., 2021) achieve a worst-case regret of order $\tilde{O}(\sqrt{d^2H^4K})$ for known reward. On a different note, (Du, Kakade, Lee, et al., 2021; Foster et al., 2021) also introduces a new structural framework for generalization in RL, called bilinear classes as it requires the Bellman error to be upper bounded by a bilinear form. Instead of using bilinear forms to capture non-linear structures, this class is not identical to BEF class of MDPs, and studying the connection is out of the scope of this paper. Specifically, *we address the shortcomings of the*

existing works on BEF-family of MDPs that assume known rewards, absence of RLSVI-type algorithms, and access to oracle planners.

Tractable planning and linearity. Planning is a major byproduct of the chosen functional representation. In general, planning can incur high computational complexity if done naïvely. Specially, (Du, Kakade, Wang, et al., 2019) shows that for some settings, even with a linear ε-approximation of the Q-function, a planning procedure able to produce an ε-optimal policy has a complexity at least 2^H. Thus, different works (Shariff and Szepesvári, 2020; Lattimore, Szepesvari, and Weisz, 2020; Van Roy and Dong, 2019) propose to leverage different low-dimensional representations of value functions or transitions to perform efficient planning. Here, we take note from (Ren et al., 2021) that Gaussian transitions induce an explicit linear value function in an RKHS. And generalize this observation with the bilinear exponential. Moreover, using uniformly good features (Rahimi and Recht, 2007) to approximate transition dynamics from our model enables us to design a tractable planner. We provide a detailed discussion of this approximation in Section 5.4. More practically, (Ren et al., 2021; Nachum and Yang, 2021) use representations given by random Fourier features (Rahimi and Recht, 2007) to approximate the transition dynamics and provide experiments validating the benefits of this approach for high-dimensional Atari-games. *Thus, we propose the first algorithm with tractable planning for BEF-family.*

5.7 Discussion

We propose the `BEF-RLSVI` algorithm for the bilinear exponential family of MDPs in the setting of episodic-RL. `BEF-RLSVI` explores using a Gaussian perturbation of rewards, and plans tractably (complexity of $\mathcal{O}(pH^3K\log(HK))$) thanks to properties of the RBF kernel. Our proof shows that clipping can be forwent for similar `RLSVI`-type algorithms. Moreover, we prove a $\sqrt{d^3H^3K}$ frequentist regret bound, which improves over existing work, accommodates unknown rewards, and matches the lower bound in terms of H and K. Regarding future work, we believe that our proof approach can be extended to rewards with bounded variance. We also believe that the extra $\sqrt{d}$ in our bound is an artefact of the proof, and specifically, the anti-concentration. We will investigate it further. Finally, we plan to study the practical efficiency of `BEF-RLSVI` through experiments on tasks with continuous state-action spaces in an extended version of this work.

Chapter 6

Deep policy gradient: improved learning of value functions

Policy gradient algorithms have proven to be successful in diverse decision making and control tasks. However, these methods suffer from high sample complexity and instability issues. In this paper, we address these challenges by providing a different approach for training the critic in the actor-critic framework. Our work builds on recent studies indicating that traditional actor-critic algorithms do not succeed in fitting the true value function, calling for the need to identify a better objective for the critic. In our method, the critic uses a new state-value (resp. state-action-value) function approximation that learns the value of the states (resp. state-action pairs) relative to their mean value rather than the absolute value as in conventional actor-critic. We prove the theoretical consistency of the new gradient estimator and observe dramatic empirical improvement across a variety of continuous control tasks and algorithms. Furthermore, we validate our method in tasks with sparse rewards, where we provide experimental evidence and theoretical insights.[1]

Contents

[1]This chapter is based on a collaboration with Yannis Flet-Berliac, Odalric Maillard and Philippe Preux (Flet-Berliac, Ouhamma, et al., 2021). It was accepted for publication at the *The International Conference on Learning Representations* (*ICLR*).

6.1 Introduction

Model-free deep reinforcement learning (RL) has been successfully used in a wide range of problem domains, ranging from teaching computers to control robots to playing sophisticated strategy games (Silver et al., 2014; Schulman, Moritz, et al., 2016; Lillicrap et al., 2016; Mnih, Badia, et al., 2016). State-of-the-art policy gradient algorithms currently combine ingenious learning schemes with neural networks as function approximators in the so-called actor-critic framework (Sutton, McAllester, et al., 2000; Schulman, Wolski, et al., 2017; Haarnoja et al., 2018). While such methods demonstrate great performance in continuous control tasks, several discrepancies persist between what motivates the conceptual framework of these algorithms and what is implemented in practice to obtain maximum gains.

For instance, research aimed at improving the learning of value functions often restricts the class of function approximators through different assumptions, then propose a critic formulation that allows for a more stable policy gradient. However, new studies (Tucker et al., 2018; Ilyas et al., 2020) indicate that state-of-the-art policy gradient methods (Schulman, Levine, et al., 2015; Schulman, Wolski, et al., 2017) fail to fit the true value function and that recently proposed state-action-dependent baselines (Gu et al., 2016; Liu et al., 2018; Wu et al., 2018) do not reduce gradient variance more than state-dependent ones.

These findings leave the reader skeptical about actor-critic algorithms, suggesting that recent research tends to improve performance by introducing a bias rather than stabilizing the learning. Consequently, attempting to find a better baseline is questionable, as critics would typically fail to fit it (Ilyas et al., 2020). In (Tucker et al., 2018), the authors argue that "much larger gains could be achieved by instead improving the accuracy of the value function". Following this line of thought, we are interested in ways to better approximate the value function. One approach addressing this issue is to put more focus on relative state-action values, an idea introduced in the literature on advantage reinforcement learning (Harmon and Baird III, n.d.) followed by works on dueling (Wang, Schaul, et al., 2016) neural networks. More recent work (Lin and Zhou, 2020) also suggests that considering the *relative action values*, or more precisely the ranking of actions in a state leads to better policies. The main argument behind this intuition is that it suffices to identify the optimal actions to solve a task. We extend this principle of relative action value with respect to the mean value to cover both state and state-action-value functions with a new objective for the critic: minimizing the variance of residual errors.

In essence, this modified loss function puts more focus on the values of states (resp. state-actions) relative to their mean value rather than their absolute values, with the intuition that solving a task corresponds to identifying the optimal action(s) rather than estimating the exact value of each state. In summary, this paper:

- Introduces **A**ctor with **V**ariance **E**stimated **C**ritic (AVEC), an actor-critic method providing a new training objective for the critic based on the residual variance.
- Provides evidence for the improvement of the value function approximation as well as theoretical consistency of the modified gradient estimator.
- Demonstrates experimentally that AVEC, when coupled with state-of-the-art policy gradient algorithms, yields a significant performance boost on a set of challenging tasks, including environments with sparse rewards.
- Provides empirical evidence supporting a better fit of the true value function and a substantial stabilization of the gradient.

6.2 Related Work

Our approach builds on three lines of research, of which we give a quick overview: policy gradient algorithms, regularization in policy gradient methods, and exploration in RL.

Policy gradient methods use stochastic gradient ascent to compute a policy gradient estimator. This was originally formulated as the REINFORCE algorithm (Williams, 1992). Kakade and Langford (2002) later created conservative policy iteration and provided lower bounds for the minimum objective improvement. (Peters, Mulling, and Altun, 2010) replaces regularization by a trust region constraint to stabilize training. In addition, extensive research investigated methods to improve the stability of gradient updates, and although it is possible to obtain an unbiased estimate of the policy gradient from empirical trajectories, the corresponding variance can be extremely high. To improve stability, (Weaver and Tao, 2001) shows that subtracting a baseline (Williams, 1992) from the value function in the policy gradient can be very beneficial in reducing variance without damaging the bias. However, in practice, these modifications on the actor-critic framework usually result in improved performance without a significant variance reduction (Tucker et al., 2018; Ilyas et al., 2020). Currently, one of the most dominant on-policy methods are proximal policy optimization (PPO) (Schulman, Wolski, et al., 2017) and trust region policy optimization (TRPO) (Schulman, Levine, et al., 2015), both of which require new samples to be collected for each gradient step. Another direction of research that overcomes this limitation is off-policy algorithms, which therefore benefit from all sample transitions; soft actor-critic (SAC) (Haarnoja et al., 2018) is one such approach achieving state-of-the-art performance.

Several works also investigate regularization effects on the policy gradient (Jaderberg et al., 2016; Namkoong and Duchi, 2017; Kartal, Hernandez-Leal, and Taylor, 2019; Flet-Berliac and Preux, 2019; Flet-Berliac and Preux, 2020); it is often used to shift the bias-variance trade-off towards reducing the variance while introducing a small bias. In RL, regularization is often

used to encourage exploration and takes the form of an entropy term (Williams and Peng, 1991; Schulman, Wolski, et al., 2017). Moreover, while regularization in machine learning generally consists in smoothing over the observation space, in the RL setting, (Thodoroff et al., 2018) shows that it is possible to smooth over the temporal dimension as well. Furthermore, (Zhao et al., 2016) analyzes the effects of a regularization using the variance of the policy gradient (the idea is reminiscent of SVRG descent (Johnson and Zhang, 2013)) which proves to provide more consistent policy improvements at the expense of reduced performance. In contrast, as we will see later, `AVEC` does not change the policy network optimization procedure nor involves any additional computational cost.

Exploration has been studied under different angles in RL, one common strategy is ε-greedy, where the agent explores with probability ε by taking a random action. This method, just like entropy regularization, enforces uniform exploration and has achieved recent success in game playing environments (Mnih, Kavukcuoglu, et al., 2013; Van Hasselt, Guez, and Silver, 2015; Mnih, Badia, et al., 2016). On the other hand, for most policy-based RL, exploration is a natural component of any algorithm following a stochastic policy, choosing sub-optimal actions with non-zero probability. Furthermore, policy gradient literature contains exploration methods based on uncertainty estimates of values (Kaelbling, 1993; Tokic, 2010), and algorithms which provide intrinsic exploration or curiosity bonus to encourage exploration (Schmidhuber, 2006; Bellemare et al., 2016; Flet-Berliac, Ferret, et al., 2021).

While existing research may share some motivations with our method, no previous work in RL applies the variance of residual errors as an objective loss function. In the context of linear regression, (Brown, 1947) considers a median-unbiased estimator minimizing the risk with respect to the absolute-deviation loss function (Pham-Gia and Hung, 2001) (similar in spirit to the variance of residual errors), their motivation is nonetheless different to ours. Indeed, they seek to be robust to outliers whereas, when considering noiseless RL problems, one usually seeks to capture those (sometimes rare) signals corresponding to the rewards.

6.3 Preliminaries

6.3.1 Background and Notations

We consider an infinite-horizon Markov Decision Problem (MDP) with continuous states $s \in \mathcal{S}$, continuous actions $a \in \mathcal{A}$, transition distribution $s_{t+1} \sim \mathcal{P}(s_t, a_t)$ and reward function $r_t \sim \mathcal{R}(s_t, a_t)$. Let $\pi_\theta(a|s)$ denote a stochastic policy with parameter θ, we restrict policies to being Gaussian distributions. In the following, π and π_θ denote the same object. The agent repeatedly interacts with the environment by sampling action $a_t \sim \pi(.|s_t)$, receives reward r_t and transitions to a new state s_{t+1}. The objective is to maximize the expected sum of discounted

rewards:

$$J(\pi) \triangleq \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r\left(s_t, a_t\right)\right], \tag{6.1}$$

where $\gamma \in [0, 1)$ is a discount factor (Puterman, 1994), and $\tau = (s_0, a_0, r_0, s_1, a_1, r_1, \dots)$ is a trajectory sampled from the environment using policy π. We denote the value of a state s in the MDP framework while following a policy π by $V^\pi(s) \triangleq \mathbb{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r\left(s_t, a_t\right)|s_0 = s\right]$ and the value of a state-action pair of performing action a in state s and then following policy π by $Q^\pi(s, a) \triangleq \mathbf{E}_{\tau\sim\pi}\left[\sum_{t=0}^{\infty}\gamma^t r\left(s_t, a_t\right)|s_0 = s, a_0 = a\right]$. Finally, the advantage function which quantifies how an action a is better than the average action in state s is denoted $A^\pi(s, a) \triangleq Q^\pi(s, a) - V^\pi(s)$.

6.3.2 Critics in Deep Policy Gradients

In this section, we consider the case where the value functions are learned using function estimators and then used in an approximation of the gradient. Without loss of generality, we consider the algorithms that approximate the state-value function V. The analysis holds for algorithms that approximate the state-action-value function Q. Let $f_\phi : \mathcal{S} \to \mathbb{R}$ be an estimator of $\hat{V}^\pi$ with ϕ its parameter. f_ϕ is traditionally learned through minimizing the mean squared error (MSE) against $\hat{V}^\pi$. At iteration k, the critic minimizes:

$$\mathcal{L}_{\text{AC}} = \mathbb{E}_s\left[\left(f_\phi(s) - \hat{V}^{\pi_{\theta_k}}(s)\right)^2\right], \tag{6.2}$$

where the states s are collected under policy π_{θ_k}, and $\hat{V}^{\pi_{\theta_k}}(s)$ is an empirical estimate of V (see Section 6.4.3 for details). Similarly, using $f_\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ instead, one can fit an empirical target $\hat{Q}^\pi$.

6.4 Method: Actor with Variance Estimated Critic

In this section, we introduce `AVEC` and discuss its correctness, motivations and implementation.

6.4.1 Defining an Alternative Critic

Recent work (Ilyas et al., 2020) empirically demonstrates that while the value network succeeds in the supervised learning task of fitting $\hat{V}^\pi$ (resp. $\hat{Q}^\pi$), it does not fit V^π (resp. Q^π). We address this deficiency in the estimation of the critic by introducing an alternative value network loss. Following empirical evidence indicating that the problem is the approximation error and not the estimator *per se*, `AVEC` adopts a loss that can provide a better approximation error, and

yields better estimators of the value function (as will be shown in Section 6.5.4). At update k:

$$\mathcal{L}_{\texttt{AVEC}} = \mathbb{E}_s \left[\left((f_\phi(s) - \hat{V}^{\pi_{\theta_k}}(s)) - \mathbb{E}_s \left[f_\phi(s) - \hat{V}^{\pi_{\theta_k}}(s) \right] \right)^2 \right], \tag{6.3}$$

with states s collected using π_{θ_k}. Note that the gradient flows in f_ϕ twice using Equation 6.3. Then, we define our bias-corrected estimator: $g_\phi : \mathcal{S} \to \mathbb{R}$ such that $g_\phi(s) = f_\phi(s) + \mathbb{E}_s[\hat{V}^{\pi_{\theta_k}}(s) - f_\phi(s)]$. Analogously to Equation 6.3, we define an alternative critic for the estimation of Q^π by replacing $\hat{V}^\pi$ by $\hat{Q}^\pi$ and $f_\phi(s)$ by $f_\phi(s, a)$.

Lemma 6.1 (AVEC Policy Gradient)**.** *If $f_\phi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ satisfies the parameterization assumption (Sutton, McAllester, et al., 2000) then g_ϕ provides an unbiased policy gradient:*

$$\nabla_\theta J(\pi_\theta) = \mathbb{E}_{(s,a)\sim\pi_\theta} \left[\nabla_\theta \log(\pi_\theta(s, a)) g_\phi(s, a) \right].$$

This result also holds for the estimation of V^{π_θ} with $f_\phi : \mathcal{S} \to \mathbb{R}$.

Proof. we consider the case in which the state-action-value function of a policy π_θ is approximated. We prove that given some assumptions on this estimator function, we can use it to yield a valid gradient direction, *i.e.* we are able to prove policy improvement when following this direction.

In this setting, the critic minimizes the loss of Equation 6.3 where the targeted function is $\hat{Q}^{\pi_\theta}(s, a)$. Therefore, when a local optimum is reached, the gradient of the latter expression is zero:

$$\nabla_\phi \mathcal{L}_{\texttt{AVEC}} = \mathbb{E}_{(s,a)\sim\pi} \Bigg[\left(\hat{Q}^{\pi_\theta}(s, a) - f_\phi(s, a) - \mathbb{E}_{(s,a)\sim\pi} \left[\hat{Q}^{\pi_\theta}(s, a) - f_\phi(s, a) \right] \right) \times \left(\left(\frac{\partial f_\phi(s, a)}{\partial \phi} - \mathbb{E}_{(s,a)\sim\pi} \left[\frac{\partial f_\phi(s, a)}{\partial \phi} \right] \right) \right] = 0$$

In the expression above, the expected value of the partial derivative disappears because the term in the first bracket is centered:

$$\mathbb{E}_{(s,a)\sim\pi} \left[(\hat{Q}^{\pi_\theta}(s, a) - f_\phi(s, a) - \mathbb{E}_{(s,a)\sim\pi}[\hat{Q}^{\pi_\theta}(s, a) - f_\phi(s, a)]) \mathbb{E}_{(s,a)\sim\pi}[\frac{\partial f_\phi(s, a)}{\partial \phi}] \right]$$
$$= \mathbb{E}_{(s,a)\sim\pi} \left[\frac{\partial f_\phi(s, a)}{\partial \phi} \right] \mathbb{E}_{(s,a)\sim\pi}[\hat{Q}^{\pi_\theta}(s, a) - f_\phi(s, a) - \mathbb{E}_{(s,a)\sim\pi}[\hat{Q}^{\pi_\theta} - f_\phi]] = 0.$$

Simplifying the gradient at the local optimum becomes:

$$\mathbb{E}_{(s,a)\sim\pi}\left[(\hat{Q}^{\pi_\theta}(s,a) - f_\phi(s,a) - \mathbb{E}_{(s,a)\sim\pi}[\hat{Q}^{\pi_\theta}(s,a) - f_\phi(s,a)])(\frac{\partial f_\phi(s,a)}{\partial \phi})\right] = 0. \tag{6.4}$$

Then, if we denote $g_\phi = f_\phi(s,a) + \mathbb{E}_{(s,a)\sim\pi}[\hat{Q}^{\pi}(s,a) - f_\phi(s,a)]$, and use the policy parameterization assumption:

$$\frac{\partial f_\phi(s,a)}{\partial \phi} = \frac{\partial \pi_\theta(s,a)}{\partial \theta}\frac{1}{\pi_\theta(s,a)}, \tag{6.5}$$

We obtain:

$$\nabla_\theta J = \mathbb{E}_{(s,a)\sim\pi_\theta}\left[\nabla_\theta \log(\pi_\theta(s,a)) g_\phi(s,a)\right]. \tag{6.6}$$

The latter follows by combining the parameterization assumption in Equation 6.5 with Equation 6.4. Indeed this entails:

$$\mathbb{E}_{(s,a)\sim\pi_\theta}\left[(\hat{Q}^{\pi_\theta}(s,a) - g_\phi(s,a))\frac{\partial \pi_\theta(s,a)}{\partial \theta}\frac{1}{\pi_\theta(s,a)}\right] = 0. \tag{6.7}$$

Since the expression above is null, we get:

$$\begin{aligned}
\nabla_\theta J &= \mathbb{E}_{(s,a)\sim\pi_\theta}[\nabla_\theta \log(\pi_\theta(s,a))\hat{Q}^{\pi_\theta}(s,a)] \\
&= \mathbb{E}_{(s,a)\sim\pi_\theta}[\nabla_\theta \log(\pi_\theta(s,a))\hat{Q}^{\pi_\theta}(s,a)] - \mathbb{E}_{(s,a)\sim\pi_\theta}[(\hat{Q}^{\pi_\theta}(s,a) - g_\phi(s,a))\frac{\partial \pi_\theta(s,a)}{\partial \theta}\frac{1}{\pi_\theta(s,a)}] \\
&= \mathbb{E}_{(s,a)\sim\pi_\theta}[\nabla_\theta \log(\pi_\theta(s,a)) g_\phi(s,a)].
\end{aligned}$$

Which finished the proof. □

Remark 6.2. *While the proof seems more or less generic, the assumption in Equation 6.5 is extremely constraining to the possible approximators. (Sutton, McAllester, et al., 2000) quotes J. Tsitsiklis who believes that a linear g_ϕ in the features of the policy may be the only feasible solution for this condition.*

Concretely, such an assumption cannot hold since neural networks are the standard approximators used in practice. Moreover, empirical analysis (Ilyas et al., 2020) indicates that commonly used algorithms fail to fit the true value function. However, this does not rule out the usefulness of the approach but rather begs for more questioning of the true effect of such biased baselines.

6.4.2 Building Motivation

Here, we present the intuition behind using AVEC for actor-critic algorithms. (Tucker et al., 2018) and (Ilyas et al., 2020) indicate that the approximation error $\|\hat{V}^\pi - V^\pi\|$ is problematic, suggesting that the variance of the empirical targets $\hat{V}^\pi(s_t)$ is high. Using $\mathcal{L}_{\texttt{AVEC}}$, our approach reduces the variance term of the MSE (or distance to V^π) but mechanistically also increases the bias. Our intuition is that since the bias is already quite substantial (Ilyas et al., 2020), it may be possible to reduce the variance enough so that even though the bias increases, the total MSE reduces.

State-value function estimation. In this case, optimizing the critic with $\mathcal{L}_{\texttt{AVEC}}$ can be interpreted as fitting $\hat{V}'^\pi(s) = \hat{V}^\pi(s) - \mathbb{E}_{s'}[\hat{V}^\pi(s')]$ using the MSE. We show that the targets $\hat{V}'^\pi$ are better estimations of $V'^\pi(s) = V^\pi(s) - \mathbb{E}_{s'}[V^\pi(s')]$ than $\hat{V}^\pi$ are of V^π. To illustrate this, consider T independent random variables $(X_i)_{i\in\{1,\dots,T\}}$. We denote $X'_i = X_i - \frac{1}{T}\sum_{j=1}^T X_j$ and $\mathbb{V}(X)$ the variance of X. Then, $\mathbb{V}(X'_i) = \mathbb{V}(X_i) - \frac{2}{T}\mathbb{V}(X_i) + \frac{1}{T^2}\sum_{j=1}^T \mathbb{V}(X_j)$ and $\mathbb{V}(X'_i) < \mathbb{V}(X_i)$ as long as $\forall i\ \frac{1}{T}\sum_{j=1}^T \mathbb{V}(X_j) < 2\mathbb{V}(X_i)$, or more generally when state-values are not strongly negatively correlated[2] and not very discordant. This entails that $\hat{V}'^\pi$ has a more compact span, and is consequently easier to fit. This analysis shows that the variance term of the MSE is reduced compared to traditional actor-critic algorithms, but does not guarantee it counterbalances the bias increase. Nevertheless, in practice, the bias is so high that the difference due to learning with AVEC is only marginal and the total MSE decreases. We empirically demonstrate this claim in Section 6.5.4.

State-action-value function estimation. In this case, Equation 6.3 translates into replacing $\hat{V}^\pi(s)$ by $\hat{Q}^\pi(s,a)$ and $f_\phi(s)$ by $f_\phi(s,a)$ and the rationale for optimizing the residual variance of the value function instead of the full MSE becomes more straightforward: the practical use of the Q-function is to disentangle the relative values of actions for each state (Sutton, McAllester, et al., 2000). AVEC's effect on relative values is illustrated in a didactic regression with one variable example in Figure 6.1 where grey markers are observations and the blue line is our current estimation. Minimizing the MSE, the line is expected to move towards the orange one in order to reduce errors uniformly. Minimizing the residual variance, it is expected to move near the red one. In fact, $\mathcal{L}_{\texttt{AVEC}}$ tends to further penalize observations that are far away from the mean, implying that AVEC allows a better recovery of the "shape" of the target near extrema. In particular, we see in the figure that the maximum and minimum observation values are quickly identified. Would the approximators be linear and the target state-values independent, the two losses become equivalent since ordinary least squares would provide minimum-variance mean-unbiased estimation.

[2] (Greensmith, Bartlett, and Baxter, 2004) analyzes the dependent case: in general, weakly dependent variables tend to concentrate more than independent ones.

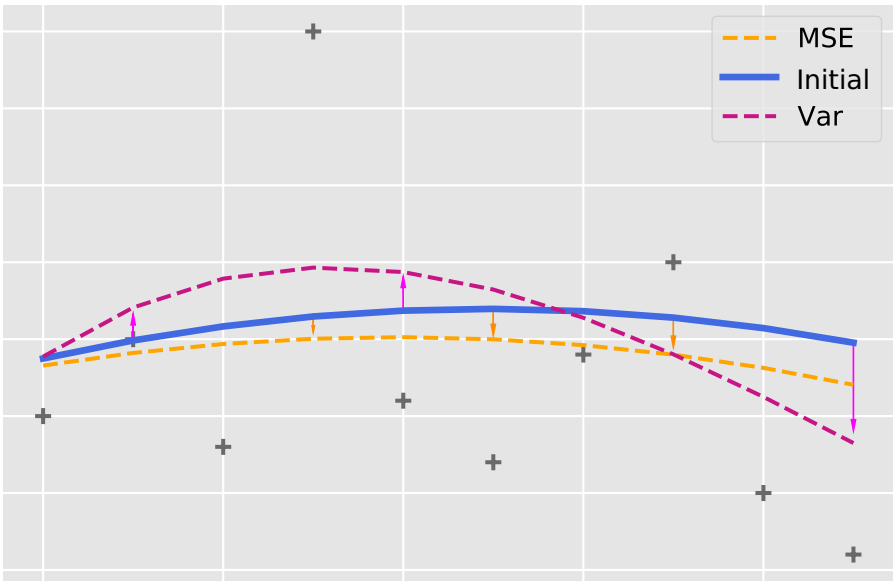


Figure 6.1 – Comparison of simple models derived when $\mathcal{L}_{\texttt{AVEC}}$ is used instead of the MSE.

It should be noted that, as in all the works related to ours, we consider noiseless tasks, *i.e.* the transition matrix is deterministic. As such, there are no outliers and extreme state-action values correspond to learning signals. In this context, high estimation errors indicate where (in the state or action-state space) the training of the value function should be improved, as opposed to possible outliers.

6.4.3 Implementation

We apply this new formulation to three of the most dominant deep policy gradient methods to study whether it results in a better estimation of the value function. A better estimation of the value function implies better policy improvements. We now describe how AVEC incorporates its residual variance objective into the critics of PPO (Schulman, Wolski, et al., 2017), TRPO (Schulman, Levine, et al., 2015) and SAC (Haarnoja et al., 2018). Let $\mathcal{B}$ be a batch of transitions. In PPO and TRPO, AVEC modifies the learning of V_ϕ (line 12 of Algorithm 6.1) using:

$$\mathcal{L}^1_{\texttt{AVEC}}(\phi) = \mathbb{E}_{s\sim\mathcal{B}}\Big[\left(f_\phi(s) - \hat{V}^\pi(s)\right) - \mathbb{E}_{s\sim\mathcal{B}}\left[f_\phi(s) - \hat{V}^\pi(s)\right]\Big]^2,$$

then $V_\phi = f_\phi(s) + \mathbb{E}_{s\sim\mathcal{B}}[\hat{V}^\pi(s) - f_\phi(s)]$, where $\hat{V}^\pi(s_t) = f_{\phi_{\text{old}}}(s_t) + A_t$ such that $f_{\phi_{\text{old}}}(s_t)$ are the estimates given by the last value function and A_t is the advantage of the policy, *i.e.* the returns minus the expected values (A_t is often estimated using generalized advantage estimation (Schulman, Moritz, et al., 2016).

Algorithm 6.1: `AVEC` for PPO or TRPO. J^{ALGO} denotes the policy loss of either algorithm (described in (Schulman, Wolski, et al., 2017; Schulman, Levine, et al., 2015)).

Input parameters: $\lambda_\pi \geq 0, \lambda_V \geq 0$
Initialize policy parameter θ and value function parameter ϕ
for each update step **do**
  batch $\mathcal{B} \leftarrow \emptyset$
  for each environment step **do**
    $a_t \sim \pi_\theta(s_t)$
    $s_{t+1} \sim \mathcal{P}(s_t, a_t)$
    $\mathcal{B} \leftarrow \mathcal{B} \cup \{(s_t, a_t, r_t, s_{t+1})\}$
  end for
  for each gradient step **do**
    $\theta \leftarrow \theta - \lambda_\pi \hat{\nabla}_\theta J^{\text{ALGO}}(\pi_\theta)$
    $\phi \leftarrow \phi - \lambda_V \hat{\nabla}_\phi \mathcal{L}^1_{\texttt{AVEC}}(\phi)$
  end for
end for

In SAC, `AVEC` modifies the objective function of $(Q_{\phi_i})_{i=1,2}$ (line 13 of Algorithm 6.2) using:

$$\mathcal{L}^2_{\texttt{AVEC}}(\phi_i) = \mathbb{E}_{(s,a)\sim\mathcal{B}}\left[(f_{\phi_i}(s,a) - \hat{Q}^\pi(s,a)) - \mathbb{E}_{(s,a)\sim\mathcal{B}}\left[f_{\phi_i}(s,a) - \hat{Q}^\pi(s,a)\right]\right]^2,$$

then $Q_{\phi_i} = f_{\phi_i}(s,a) + \mathbb{E}_{(s,a)\sim\mathcal{B}}[\hat{Q}^\pi(s,a) - f_{\phi_i}(s,a)]$, where $\hat{Q}^\pi(s,a)$ is estimated using temporal difference (see (Haarnoja et al., 2018)): $\hat{Q}^\pi(s_t,a_t) = r(s_t,a_t) + \gamma\mathbb{E}_{s_{t+1}\sim\pi}[V_{\bar{\psi}}(s_{t+1})]$ with $\bar{\psi}$ the value function parameter (see Algorithm 6.2).

Algorithm 6.2: `AVEC` coupled with SAC.

Input parameters: $\beta \in [0,1], \lambda_V \geq 0, \lambda_Q \geq 0, \lambda_\pi \geq 0$
Initialize policy parameter θ, value function parameter ψ and $\bar{\psi}$ and Q-functions parameters ϕ_1 and ϕ_2 and $\mathcal{D} \leftarrow \emptyset$
for each iteration **do**
  for each step **do**
    $a_t \sim \pi_\theta(a_t|s_t)$
    $s_{t+1} \sim \mathcal{P}(. \mid s_t, a_t)$
    $\mathcal{D} \leftarrow \mathcal{D} \cup \{(s_t, a_t, r_t, s_{t+1})\}$
  end for
  for each gradient step **do**
    sample batch $\mathcal{B}$ from $\mathcal{D}$
    Update: $\psi \leftarrow \psi - \lambda_V \hat{\nabla}_\psi J_V(\psi)$ and $\phi_i \leftarrow \phi_i - \lambda_Q \hat{\nabla}_{\phi_i} \mathcal{L}^2_{\texttt{AVEC}}(\phi_i)$ for $i \in \{1,2\}$
    Update $\theta \leftarrow \theta - \lambda_\pi \hat{\nabla}_\theta J(\pi_\theta)$; $\bar{\psi} \leftarrow \beta\psi + (1-\beta)\bar{\psi}$
  end for
end for

Finally, notice that `AVEC` does not modify any other part of the considered algorithms whatsoever, which keeps their implementation and computational complexity unchanged.

Remark 6.3. *Theoretically, $\mathcal{L}_{\texttt{AVEC}}$ is defined as the residual variance of the value function* (cf *Equation 6.3*)*. However, state-values for a non-optimal policy are dependent and the variance is not tractable without access to the joint law of state-values. Consequently, to implement* AVEC *in practice we use the best-known proxy at hand, which is the empirical variance formula assuming independence:*

$$\mathcal{L}_{\texttt{AVEC}} = \frac{1}{T-1}\sum_{t=1}^{T}\left((f_\phi(s_t) - \hat{V}^\pi(s_t)) - \frac{1}{T}\sum_{t=1}^{T}(f_\phi(s_t) - \hat{V}^\pi(s_t))\right)^2,$$

where T is the size of the sampled trajectory.

(Greensmith, Bartlett, and Baxter, 2004) provides some support for this approximation by showing that weakly dependent variables tend to concentrate more than independent ones.

6.5 Experimental Study

In this section, we conduct experiments along four orthogonal directions. (a) We validate the superiority of AVEC compared to the traditional actor-critic training. (b) We evaluate AVEC in environments with sparse rewards. (c) We clarify the practical implications of using AVEC by examining the bias in both the empirical and true value function estimations as well as the variance in the empirical gradient. (d) We provide an ablation analysis and study the bias-variance trade-off in the critic by considering two continuous control tasks.
We point out that a comparison to variance-reduction methods is not considered in this paper: (Tucker et al., 2018) demonstrated that their implementations diverge from the unbiased methods presented in the respective papers and unveiled that not only do they fail to reduce the variance of the gradient, but that their unbiased versions do not improve performance either. Note that in all experiments we choose the hyperparameters providing the best performance for the considered methods which can only penalyze AVEC (*cf* Appendix D.1). In all the figures hereafter (except Figure 6.5c and 6.5d), lines are average performances and shaded areas represent one standard deviation.

6.5.1 Continuous Control

For ease of comparison with other methods, we evaluate AVEC on the MuJoCo (Todorov, Erez, and Tassa, 2012) and the PyBullet (Coumans and Bai, 2016) continuous control benchmarks (see Appendix D.2 for details) using OpenAI Gym (Brockman et al., 2016). Note that the

Task	SAC	AVEC-SAC	PPO	AVEC-PPO
Ant	3084	$\mathbf{3650 \pm 127}\,(+\mathbf{18}\%)$	972	$\mathbf{1202 \pm 148}\,(+\mathbf{24}\%)$
AntBullet	1193	$\mathbf{2252 \pm 82}\,(+\mathbf{89}\%)$	1174	$\mathbf{2216 \pm 99}\,(+\mathbf{89}\%)$
HalfCheetah	10028	$\mathbf{11018 \pm 102}\,(+\mathbf{10}\%)$	1068	$\mathbf{1403 \pm 37}\,(+\mathbf{31}\%)$
HalfCheetahBullet	1255	$\mathbf{1331 \pm 184}\,(+\mathbf{6}\%)$	1329	$\mathbf{2223 \pm 62}\,(+\mathbf{67}\%)$
Humanoid	4084	$\mathbf{4472 \pm 424}\,(+\mathbf{10}\%)$	391	$\mathbf{415 \pm 4.6}\,(+\mathbf{6}\%)$
Reacher	-6.0	$\mathbf{-5.0 \pm 0.1}\,(+\mathbf{20}\%)$	-7.4	$\mathbf{-5.9 \pm 0.3}\,(+\mathbf{25}\%)$
Walker2d	3452	$\mathbf{4334 \pm 128}\,(+\mathbf{26}\%)$	2193	$\mathbf{2923 \pm 151}\,(+\mathbf{33}\%)$

Table 6.1 – Average total reward of the last 100 episodes over 6 runs of 10^6 timesteps. Comparative evaluation of AVEC with SAC and PPO. $\pm$ corresponds to a single standard deviation over trials and $(.\%)$ is the change in performance due to AVEC.

PyBullet versions of the locomotion tasks are harder than the MuJoCo equivalents[3]. We choose a representative set of tasks for the experimental evaluation; their action and observation space dimensions are reported in Appendix D.3. We assess the benefits of AVEC when coupled with the most prominent policy gradient algorithms, currently state-of-the-art methods: PPO (Schulman, Wolski, et al., 2017) and TRPO (Schulman, Levine, et al., 2015), both on-policy methods, and SAC (Haarnoja et al., 2018), an off-policy maximum entropy deep RL algorithm.

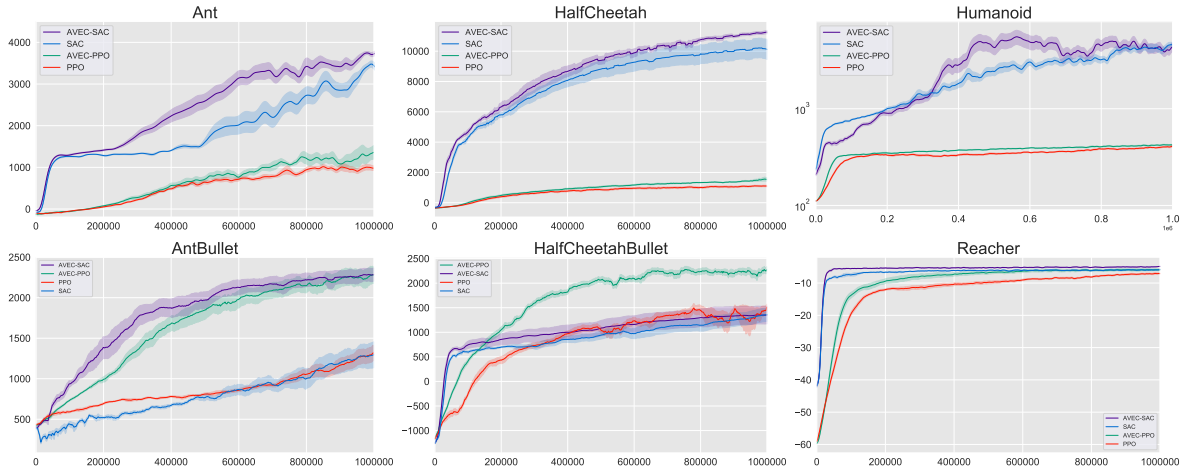


Figure 6.2 – Comparative evaluation (6 seeds) of AVEC with SAC and PPO on PyBullet ("TaskBullet") and MuJoCo ("Task") tasks. X-axis: number of timesteps. Y-axis: average total reward.

Table 6.1 reports the results while Figure 6.2 and 6.3 show the total average return for SAC and PPO. TRPO results are provided in Section 6.5.2 for readability. We provide the list of hyperparameters and further implementation details in Appendix D.1.

When coupled with SAC and PPO, AVEC brings very significant improvement (on average **+26%** for SAC and **+39%** for PPO) in the performance of the policy gradient algorithms, improvement which is consistent across tasks. As for TRPO, while the improvement in performance is less striking, AVEC still manages to be more efficient in terms of sampling in all tasks.

[3]Bullet Physics SDK GitHub Issue.

Overall, AVEC improves TRPO, PPO and SAC in terms of performance and efficiency. This does not imply that our method would also improve other policy gradient methods that use the traditional actor-critic framework, but since we evaluate our method coupled with three of the best performing on- and off-policy algorithms, we believe that these experiments are sufficient to prove the relevance of AVEC.

In addition, in Figure 6.3, we plot the total average return for AVEC coupled with SAC and PPO on the Walker2d task. Similar to considered other continuous control tasks from MuJoCo and PyBullet, AVEC brings a significant performance improvement (+26% for SAC and +33% for PPO), confirming the generality of our approach.

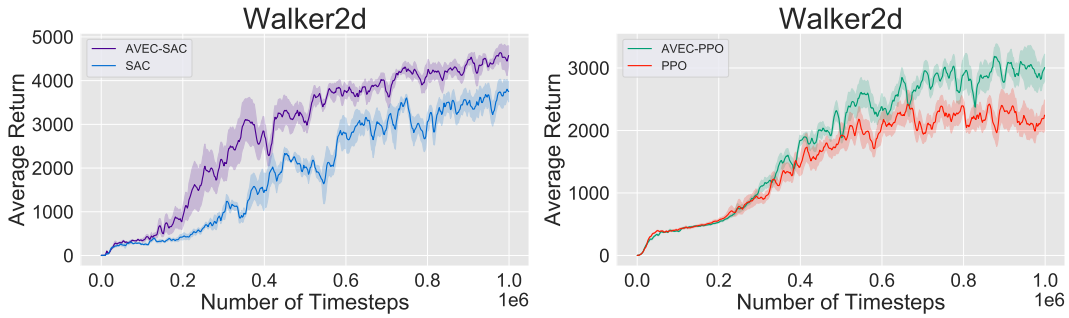


Figure 6.3 – Comparative evaluation (6 seeds) of AVEC with SAC (left) and PPO (right) on the Walker2d MuJoCo task. Lines are average performances and shaded areas represent one standard deviation.

Finally, in our experiments we do not seek the best hyperparameters for the AVEC variants, we simply adopt the parameters allowing us to optimally reproduce the baselines. Alternatively, if one seeks to evaluate AVEC independently of a considered baseline, further hyperparameter tuning should produce better results. Notice that since no additional calculations are needed in AVEC's implementation, computational complexity remains unchanged.

6.5.2 Comparison with TRPO

In order to evaluate the performance gains in using AVEC instead of the usual actor-critic framework, we produce some additional experiments with the TRPO (Schulman, Levine, et al., 2015) algorithm. Figure 6.4 shows the learning curves while Table 6.2 reports the results.

6.5.3 Sparse Reward Signals

Domains with sparse rewards are challenging to solve with uniform exploration as agents receive no feedback on their actions before starting to collect rewards. In such conditions AVEC performs better, suggesting that the *shape* of the value function is better approximated, encouraging exploration.

Task	TRPO	AVEC-TRPO
Ant	-50.5	$\mathbf{-43.5 \pm 2.2}\,(\mathbf{+16\%})$
AntBullet	564	$\mathbf{970 \pm 70}\,(\mathbf{+72\%})$
HCheetah	346	$\mathbf{466 \pm 56}\,(\mathbf{+35\%})$
HCBullet	1154	$\mathbf{1281 \pm 94}\,(\mathbf{+11\%})$
Humanoid	$\mathbf{352}$	$344 \pm 1.2\,(-3\%)$
Reacher	$\mathbf{-8.5}$	$-9.9 \pm 1.3\,(-16\%)$

Table 6.2 – Average total reward of the last 100 episodes over 6 runs of 10^6 timesteps. Comparative evaluation of AVEC with TRPO. $\pm$ corresponds to a single standard deviation over trials and $(.\%)$ is the change in performance due to AVEC.

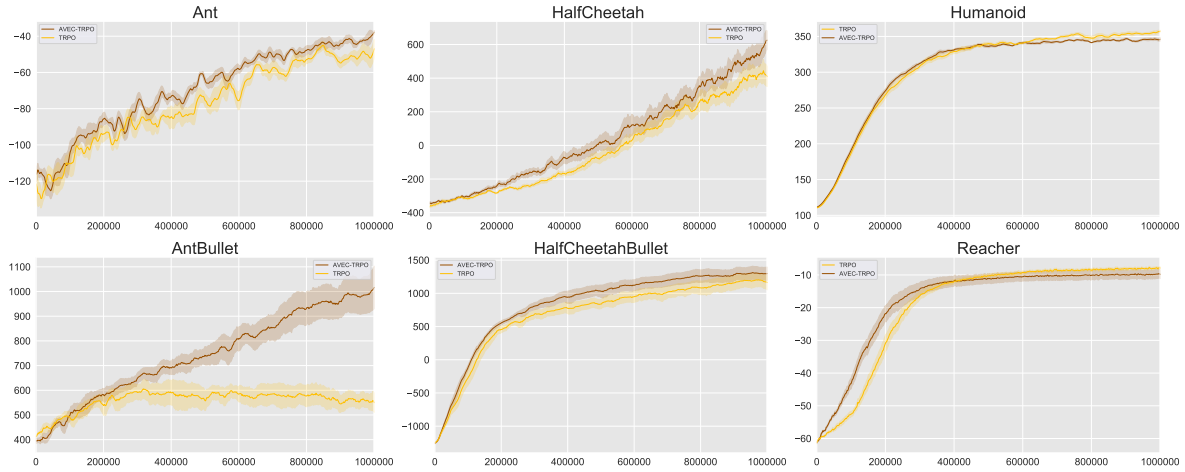
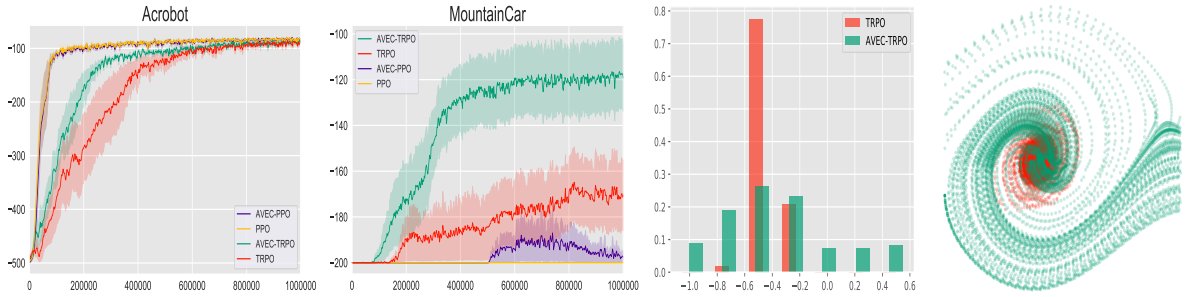


Figure 6.4 – Comparative evaluation of AVEC with TRPO. We run with 6 different seeds: lines are average performances and shaded areas represent one standard deviation.

The relative value estimate of an unseen state is more accurate: in Section 6.4.2, AVEC identifies extreme state-values (*e.g.*, non-zero rewards in tasks with sparse rewards) faster. In Figures 6.5a and 6.5b, we report the performance of AVEC in the Acrobot and MountainCar environments: both have sparse rewards. AVEC enhances TRPO and PPO in both experiments. When PPO and AVEC-PPO both reach the best possible performance, AVEC-PPO exhibits better sample efficiency. Figures 6.5c and 6.5d illustrate how the agent improves its exploration strategy in MountainCar: while the PPO agent remains stuck at the bottom of the hill (red), the graph suggest that AVEC-PPO learns the difficult locomotion principles in the absence of rewards and visits a much larger part of the state space (green).

This improved performance in sparse environments can be explained by the fact that AVEC is able to pick up on experienced positive reward more easily. Moreover, the reconstructed shape of the value function is more accurate around such rewarding states, which pushes the agent to explore further around experienced states with high values.



(a) (b) (c) (d)

Figure 6.5 – (a,b): Comparative evaluation (6 seeds) of AVEC in sparse reward tasks. X-axis: number of timesteps. Y-axis: average total reward. (c,d): Respectively state visitation frequency and phase portrait of visited states of AVEC-TRPO (green) and TRPO (red) in MountainCar.

6.5.4 Analysis of the Variance Estimated Critic

In order to further validate AVEC, we evaluate the performance of the value network in more detail: we examine (a) the estimation error (distance to the empirical target), (b) the approximation error (distance to the true target) and (c) the empirical variance of the gradient. (a,b) should be put into perspective with the conclusions of (Ilyas et al., 2020) where it is found that the critic only fits the empirical value function but not the true one. (c) should be placed in light of (Tucker et al., 2018) highlighting a failure of recently proposed state-action-dependent baselines to reduce the variance.

Learning the Empirical Target. In Figure 6.6, we report the quality of fit (MSE) of the empirical target $\hat{V}^{\pi}$ in the methods PPO and AVEC-PPO in the AntBullet and HalfCheetahBullet tasks.

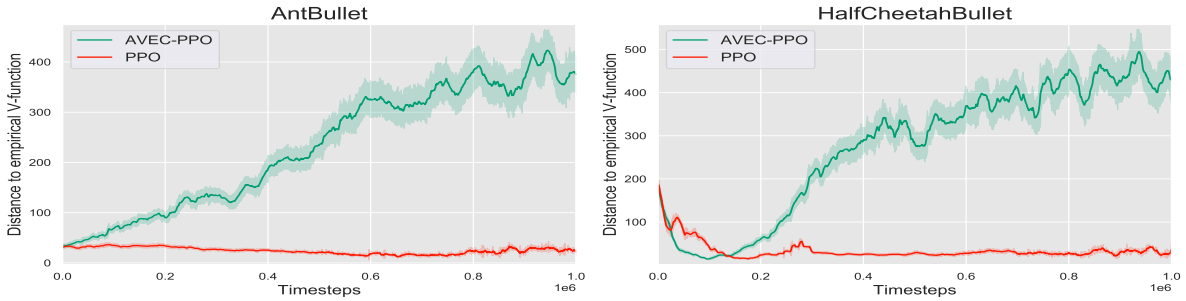


Figure 6.6 – L_2 distance to $\hat{V}^{\pi}$.

We observe that PPO better fits the empirical target than when equipped with AVEC, which is to be expected since vanilla PPO optimizes the MSE directly. This result put aside the remarkable improvement in the performance of AVEC-PPO (Figure 6.2) suggests that AVEC might be a better estimator of the true value function. We examine this claim below because if true, it would indicate that it is indeed possible to simultaneously improve the performance of the agents and the stability of the method.

Learning the True Target. A fundamental premise of policy gradient methods is that optimizing the objective based on empirical return leads to a better policy. Which is why we investigate the quality of fit of the true target. To approximate the true value function, we fit the returns sampled from the current policy using a large number of transitions $(3 \cdot 10^5)$.

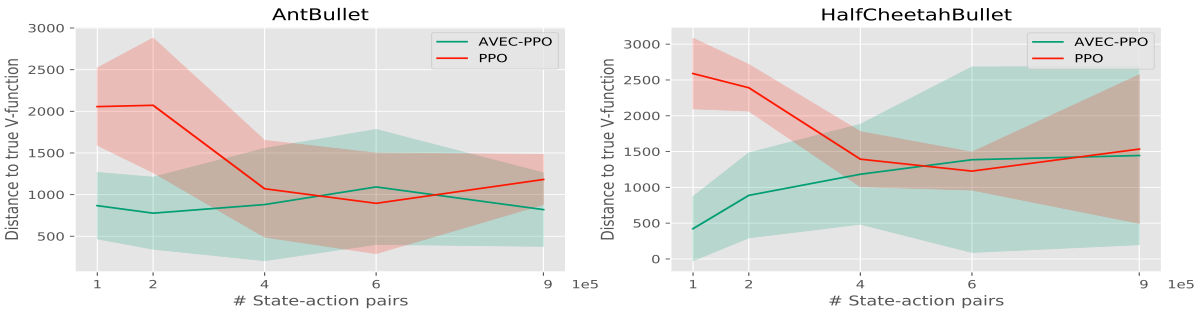


Figure 6.7 – L_2 distance to V^π. X-axis: we run PPO and AVEC-PPO and $\forall t \in \{1, 2, 4, 6, 9\} \cdot 10^5$ we stop training, use the current policy to collect $3 \cdot 10^5$ transitions and estimate V^π.

Figure 6.7 shows that g_ϕ is far closer to the true value function half of the time (horizon is 10^6) than the estimator obtained with MSE, then as close to it. Comparing Figure 6.7 with Figure 6.6, we see that the distance to the true target is close to the estimation error for AVEC-PPO, while for PPO, it is at least two orders of magnitude higher at all times. We further investigate these results in Figure6.8 where we study the variation of the squared bias and variance components of the MSE to the true target (MSE $=$ Var $+$ Bias^2).

Variation of the Bias and Variance terms: PPO. In Figure 6.8, we show the variation of the bias and variance terms in the MSE between the estimators (of AVEC-PPO and PPO) and the true target: $\mathbb{E}[\|g_\phi - V^\pi\|_2^2] = \text{Bias}(\texttt{AVEC})^2 + \text{Var}(\texttt{AVEC})$ and $\mathbb{E}[\|V_\phi(\text{PPO}) - V^\pi\|_2^2] = \text{Bias}(\text{PPO})^2 + \text{Var}(\text{PPO})$ where $V_\phi(\text{PPO})$ is the value function estimator in PPO. Let us define what we study in exact terms: $\%\text{Variation}(\text{Bias}) = \frac{\text{Bias}^2(\texttt{AVEC-PPO}) - \text{Bias}^2(\text{PPO})}{\text{Bias}^2(\text{PPO})}$ and $\%\text{Variation}(\text{Var}) = \frac{\text{Var}(\texttt{AVEC-PPO}) - \text{Var}(\text{PPO})}{\text{Var}(\text{PPO})}$. X-axis: we run PPO and AVEC-PPO and for every $t \in \{1, 2, 4, 6, 9\} \cdot 10^5$, we stop training, use the current policy to interact with the environment for $3 \cdot 10^5$ transitions, and use these transitions to estimate the true value function.

We observe that the variance reduction is more substantial than that of the bias. Using those results and Figure 6.7 showing that the distance of the estimator to V^π is lower when using AVEC confirms that the variance reduction effect counterbalances the bias increase. Note that the % Variation of the Var term is always negative in our experiments, and that the shaded areas that suggest otherwise are merely due to a false assumption of symmetrical deviations, itself due to the assumption of Gaussianity needed to construct confidence intervals.

For completeness, we also analyze the distance to the true target for the Q-function estimator in SAC and AVEC-SAC in AntBullet and HalfCheetahBullet. Indeed, in Figure 6.9, we compare

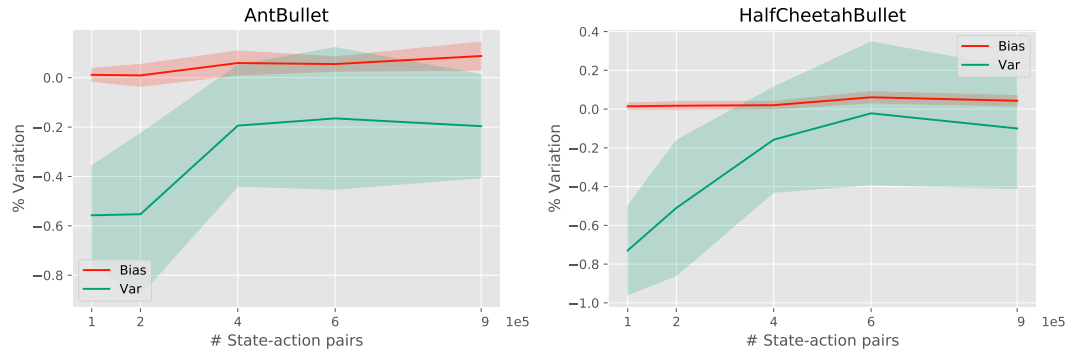


Figure 6.8 – % Variation of the bias and variance terms in the MSE between the estimator and the true target. Lines are average variations and shaded areas represent one standard deviation (5 seeds).

the error between the Q-function estimator and the true Q-function for SAC and AVEC-SAC in AntBullet and HalfCheetahBullet.

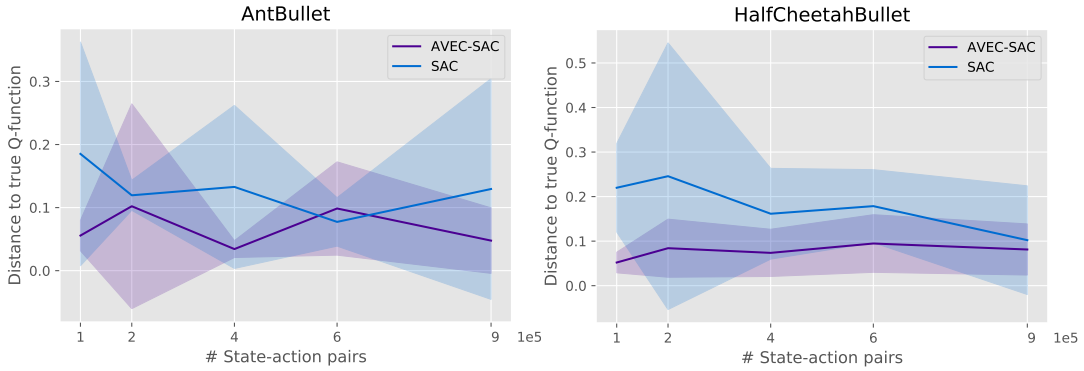


Figure 6.9 – Distance to the true Q-function (SAC). X-axis: we run SAC and AVEC-SAC and for every $t \in \{1, 2, 4, 6, 9\} \cdot 10^5$ we stop training, use the current policy to interact with the environment for $3 \cdot 10^5$ transitions, and use these transitions to estimate the true value function. Lines are average performances and shaded areas represent one standard deviation.

We note a modest but consistent reduction in this error when using AVEC coupled with SAC, echoing the significant performance gains in Figure 6.2. We conclude that AVEC improves the value function approximation and we expect that the gradient is more stable.

Empirical Variance Reduction. We choose to study the gradient variance using the average pairwise cosine similarity metric as it allows a comparison with (Ilyas et al., 2020), with which we share the same experimental setup and scales.

Figure 6.10 shows that AVEC yields a higher average pairwise cosine similarity, which means closer batch-estimates of the gradient and, in turn, indicates smaller gradient variance.

In Figure 6.11, we study the empirical variance of the gradient in measuring the average pairwise cosine similarity (10 gradient measurements) in two additional tasks: HopperBullet and Walker2DBullet. We also vary the trajectory size used in the estimation of the gradient.

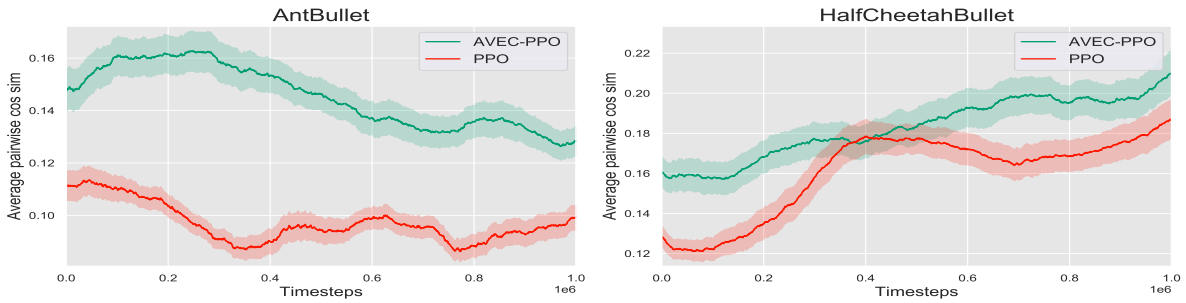


Figure 6.10 – Average gradient cosine-similarity (over 10 batches per iteration).

The variance reduction effect observed in several environments suggests that AVEC is the first method since the introduction of the value function baseline to further reduce the variance of the gradient and improve performance.

6.5.5 Ablation Study

In this section, we examine how changing the relative importance of the bias and the residual variance in the loss of the value network affects learning. For this study, we choose difficult tasks of PyBullet and use PPO because it is more efficient than TRPO and requires less computations than SAC. For an estimator $\hat{y}_n$ of $(y_i)_{i\in\{1,\dots,n\}}$, we write $\text{Bias} = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)$ and $\text{Var} = \frac{1}{n-1}\sum_{i=1}^{n}(\hat{y}_i - y_i - \text{Bias})^2$. Consequently: $\text{MSE} = \text{Var} + \text{Bias}^2$. We denote $\mathcal{L}_\alpha = \text{Var} + \alpha\text{Bias}^2$, with $\alpha \in \mathbb{R}$. In Figure 6.12, *Bias-*α means that we use $\mathcal{L}_\alpha$ and *Var-*α means that we use $\mathcal{L}_{\frac{1}{\alpha}}$. We observe that while no consistent order on the choices of α is identified, AVEC seems to outperform all other weightings. Note that, for readability purposes, the graphs have been split and the curves of AVEC-PPO and PPO are the same in Figures 6.12a and 6.12c, and in Figures 6.12b and 6.12d. A more extensive hyper-parameter study with more α values might provide even higher performances, nevertheless we believe that the stability of an algorithm is crucial for a reliable performance. As such, the tuning of hyperparameters to achieve good results should remain mild.

6.6 Discussion

In this work, we introduce a new training objective for the critic in actor-critic algorithms to better approximate the true value function. In addition to being well-motivated by recent studies on the behaviour of deep policy gradient algorithms, we demonstrate that this modification is both theoretically sound and intuitively supported by the need to improve the approximation error of the critic. The application of **A**ctor with **V**ariance **E**stimated **C**ritic (AVEC) to state-of-the-art policy gradient methods produces considerable gains in performance (on average +26% for SAC and +39% for PPO) over the standard actor-critic training, without any additional hyperparameter tuning.

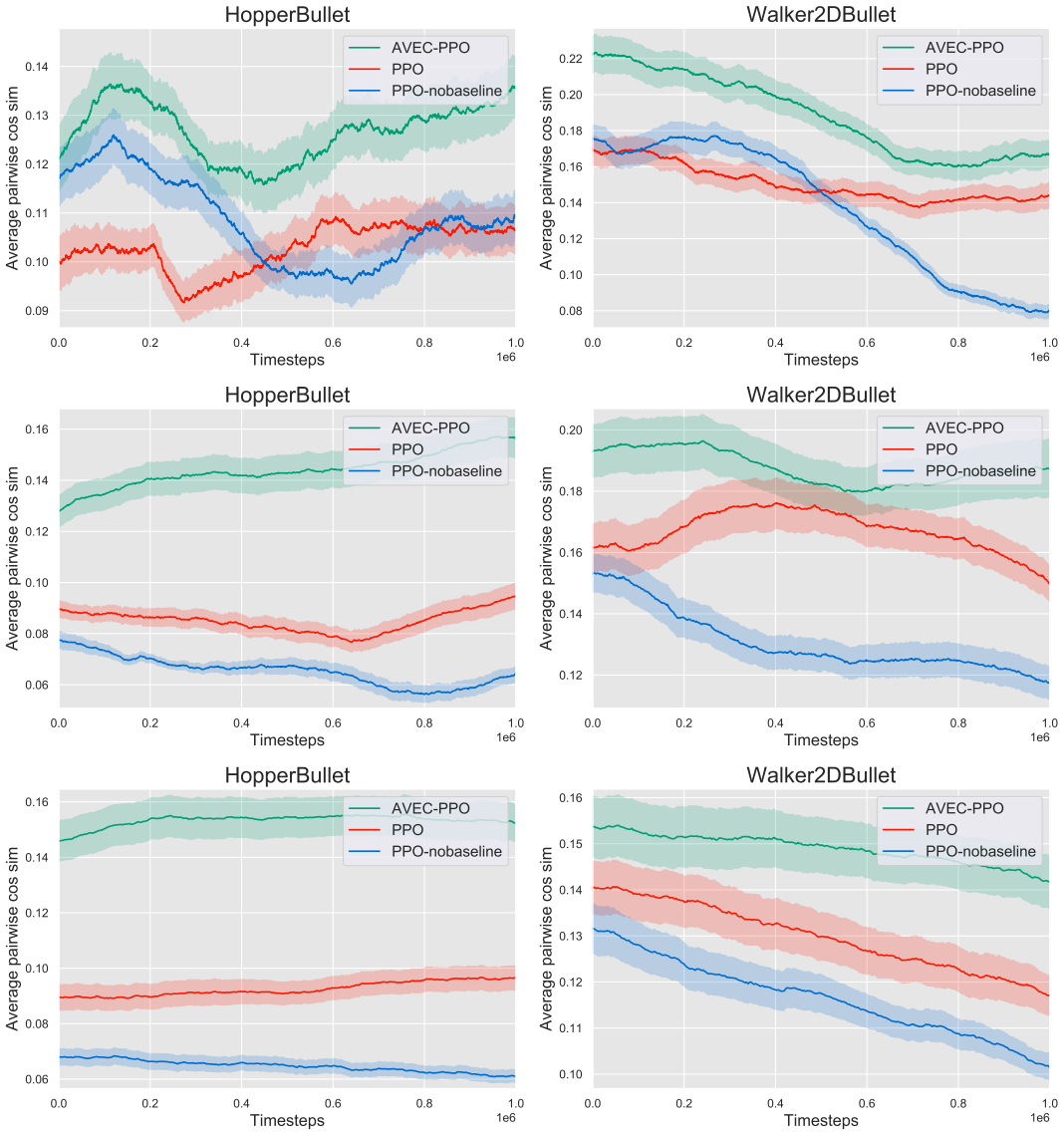


Figure 6.11 – Average cosine similarity between gradient measurements. `AVEC` empirically reduces the variance compared to PPO or PPO without a baseline (PPO-nobaseline). Trajectory size used in estimation of the gradient variance: 3000 (upper row), 6000 (middle row), 9000 (lower row). Lines are average performances and shaded areas represent one standard deviation.

First, for SAC-like algorithms where the critic learns a state-action-value function, our results strongly suggest that state-actions with extreme values are identified more quickly. Second, for PPO-like methods where the critic learns the state-values, we show that the variance of the gradient is reduced and empirically demonstrate that this is due to a better approximation of the state-values. In sparse reward environments, the theoretical intuition behind a variance estimated critic is more explicit and is also supported by empirical evidence. In addition to corroborating the results in (Ilyas et al., 2020) proving that the value estimator fails to fit V^{π}, we propose a method that succeeds in improving both the sample complexity and the stability of prominent actor-critic algorithms. Furthermore, `AVEC` benefits from its simplicity

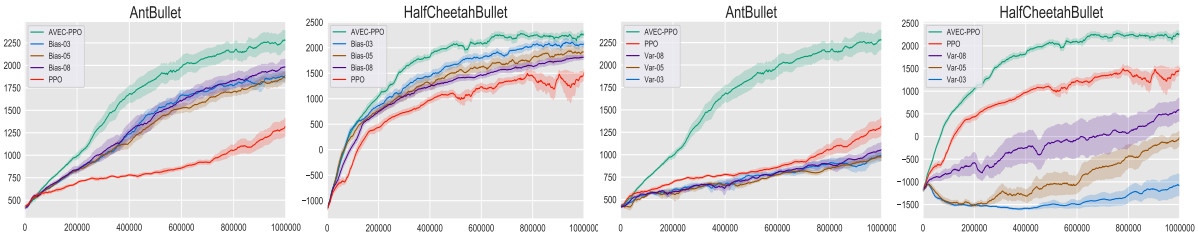


Figure 6.12 – Sensitivity (6 seeds) of AVEC-PPO with respect to (a,b): the bias; (c,d): the variance. X-axis: number of timesteps. Y-axis: average total reward.

of implementation since no further assumptions are required (such as horizon awareness (Tucker et al., 2018) to remedy the deficiency of existing variance-reduction methods) and the modification of current algorithms represents only a few lines of code.

In this paper, we have demonstrated the benefits of a more thorough analysis of the critic objective in policy gradient methods. Despite our strongly favorable results, we do not claim that the residual variance is the optimal loss for the state-value or the state-action-value functions, and we note that the design of comparably superior estimators for critics in deep policy gradient methods merits further study. In future work, further analysis of the bias-variance trade-off and extension of the results to stochastic environments is anticipated; we consider the problem of noise separation in the latter, as this is the first obstacle to accessing the variance and distinguishing extreme values from outliers.

Chapter 7

General Conclusion and Perspectives

The past is gone, the future is unseen
Rise and seize the opportunity in between

Poet unknown[1]

In this thesis, we have made significant contributions to the fields of bandits, linear regression, and reinforcement learning. Our overarching goal was to make the RL framework, algorithms, and assumptions more realistic.

We began by studying the simplest model for sequential decision making, multi-armed bandits, in **Chapter 3**. We considered the finite number of arms case and studied a pure exploration objective called Thresholding bandits. We proposed a Frank-Wolfe based method of designing algorithms, which we made to extend and solve a wider class of losses. By doing so, we were able to improve the state-of-the-art both theoretically and empirically. Moreover, we applied our newly formulated, intuitive, and generic proof scheme to improve the bounds on the number of mistakes made by all previously existing algorithms in this setting. In terms of the limitations of our algorithm, it should be noted that our current state-of-the-art bound remains within a factor of 4 in the exponent from the ideal oracle. Additionally, during our research, we made an intriguing observation that adaptive algorithms can sometimes outperform non-adaptive (*i.e.* offline) oracles. A theoretical understanding of this phenomenon has yet to be developed. Lastly, we are also keen on extending this setting to large or continuous actions, as there are connections with other index-based algorithms such as IMED (Honda and Takemura, 2015). Further examination of these connections could lead to a deeper understanding of how to scale our generic algorithm.

[1] Arabic proverbs collected and translated to english by @ArabicWords. The original wording of the verse is:

ما فات مضى وما سيأتيك فأين ** قم فاغتنم الفرصة بين العدمين

In **Chapter 4**, we delved into the linear regression problem and uncovered an old result from the adversarial setting, which we adapted for the stochastic case. Our findings revealed that this should be the default algorithm instead of the commonly used ridge regression. Additionally, we applied this modification to linear bandits and proved that it allows for the elimination of a pervasive assumption. Our analysis holds value not just in theory but also for practical use, as it presents a novel perspective on a frequently employed method and highlights novel insights about the effect of regularization. One of the main limitations of this method becomes evident in the context of MDPs, as it can be difficult to regularize using a random future explanatory variable. However, we were still able to eliminate the assumption of a *known* bound on observations in MDPs using other techniques in **Chapter 5**. Additionally, in **Chapter 4**, we discussed how the dependence on d in our bounds can be improved using Theorem 1 of (Tirinzoni, Pirotta, et al., 2020). Possible avenues for future research building on this contribution include applying these techniques to more complex regression scenarios where the features are themselves dynamic, such as neural networks and regression trees.

To further our progress, we focused our interest on MDPs with continuous state action spaces in **Chapter 5**. At first we were intrigued by the ubiquity of the linear MDP model, this prevalence can be interpreted as a lack of optimism in finding efficient and effective representational assumptions for learning. Defying this attitude, we became interested in a recent MDP representation that has shown great promise and compelling expressive power, modeling real MDPs like Tabular, Factored, and Linear Quadratic Regulators. While investigating this model, we discovered a new and crucial property about it that allowed us to create an algorithm with tractable planning, which enjoys an optimal regret bound in its dependence on the number of episodes and their length. Furthermore, we presented several results that hold independent value, such as the ability to eliminate the need for clipping value functions, which removes an superfluous and omnipresent non-linearity. A major challenge we identified in our research is the dependence on d in our bound. From the available literature, it appears that this dependence is always suboptimal for tractable algorithms, and that a slight modification to UCB-style algorithms (which are intractable) can lead to the optimal dependence. We aim to investigate whether the increased dependence on dimensionality is a necessary trade-off for achieving tractability. Another area for future research is testing this representation in practice to evaluate its effectiveness. Additionally, a necessary direction for future work is learning features instead of providing them to the learner, which could be achieved through techniques such as causal discovery (Zhang et al., 2022; Chalupka, Eberhardt, and Perona, 2017) or automatic Bayesian search (Malkomes, Schaff, and Garnett, 2016).

In **Chapter 6**, the final contribution of this thesis, we examined a framework of decision-making known as deep policy gradients. This is an infinite-horizon setting that utilizes deep neural networks to model both the value function and the policy, and is optimized through the use of gradient descent. Inspired by some recently discovered -practical- limitations in

existing algorithms, we proposed a minor modification to the value function loss which led to significant enhancements. Our proposed method was supported by both theoretical arguments and empirical evidence, showing improvements over traditional training loss and proving the accuracy of our insights. For instance, we validated independent findings, hypothesized possible implications for these shortcomings, and validated our intuition as well through practical experimentation on simulated environments. In terms of extending this work, one potential next step would be to find a balance between bias and variance in the mean square error loss, rather than completely eliminating bias as we have done. Additionally, some researchers propose that other parameters such as regularization or the weight of the stability penalty could also be optimized using gradient descent for automation (Haarnoja et al., 2018). Lastly, from a theoretical perspective, we are curious about the possibility of replacing the iterative optimization technique with theoretically sound algorithms for bi-level problems.

To summarize our contributions, they are organized between 1) improvements of existing methods, including reducing assumptions like the boundedness in linear regression Chapter 4 and simplifying algorithms like removing clipping Chapter 5 and slightly modifying the loss Chapter 6. And 2), notable novel contributions such as generic algorithm design schemes and general proof techniques Chapter 2, and discovering novel implications for useful structures Chapter 5 therefore leading to improved tractability and simpler algorithms. Our goal throughout this journey was unique: to contribute to the RL research by improving and proposing algorithms that are both theoretically sound and user-friendly for practitioners, and to support the advancement towards a more realistic RL formalism. We have made notable contributions to the research supporting this goal by proposing novel methods, adapting existing algorithms to new settings, and uncovering new properties of commonly used models. Overall, we believe that this thesis has made significant contributions to the field of RL and will have a lasting impact on the field.

Our directly relevant perspectives fall into two categories: **1) Designing tractable and optimal RL algorithms**, for which we believe that the bilinear exponential family could be very perinent. Indeed it is an expressive representation enjoying many interesting properties that can be useful for other RL objectives. For instance, we intend to study this model for best policy identification as well as reward-free reinforcement learning. Furthermore, we wish to study the empirical performance of this representation, and possibly incorporate provable methods to automatically build the feature mapping (Malkomes, Schaff, and Garnett, 2016) instead of requiring it as prior knowledge. 2) **Improving model learning in RL**, we are interested in studying new RL settings and attempting to improve model estimation. This could take the form of theoretical improvements, *e.g.* in the mixture of experts setting, information about the level of confidence for each expert could be available and we don't know how to take advantage of it to make a final prediction. Improving model learning concerns empirical methods as well, *e.g.* in the deep learning literature, if validation data is available then the learned model could

be calibrated in the hope of better adjusting to any eventual change of distribution. Since the coordinates of the test set are given, it should be studied whether incorporating this information could be beneficial, in the form of a regularization for example.

All in all, our goal in this thesis and for the future is to combine probability theory with machine learning and optimization techniques to tackle sequential decision making problems. Practically, we are interested in studying online learning and reinforcement learning problems and designing algorithms that can a) can be generic enough to model realistic scenarios or to constitute plausible approximations, and b) can be analyzed theoretically without stringent assumptions.

Appendix A

Complements on Chapter 3

Contents

A.1 Properties of index-based algorithms

An algorithm is index-based if, at any round t, it pulls $k_t = \arg\min_k I^k_{N_{k,t-1}}$ where the index $I^k_{N_{k,t}}$ depends only on the number of pulls and on rewards of arm k. That index does not change when other arms are pulled.

For $C \geq 0$, let $\mathcal{F}_C \triangleq \{\exists T' \leq T, \forall k \in [K], I^k_{N_{k,T'}} \geq C\}$ be the event that at some time before T, all arm indices are above a value C. And let $\tau_k(C) = \min\{n | I^k_n \geq C\}$ be the minimal number of pulls of arm k such that its index becomes greater than C.

We start with two immediate remarks about index-based algorithms.

Lemma A.1. *If $I^j_{N_{j,t}} \geq C$, then at the next time t' when an index-based algorithm pulls arm j, it necessarily holds that $\min_k I^k_{N^k_{t'-1}} \geq C$.*

Lemma A.2. *If $\min_k I^k_{N_{k,t}} \geq C$ then for all k, by definition of $\tau_k(C)$, $N_{k,t} \geq \tau_k(C)$.*

This next lemma explicits $\mathcal{F}_C$ using $(\tau_k(C))_{k\in[K]}$.

Lemma A.3. *An index-based algorithm verifies $\mathcal{F}_C = \{\sum_k \tau_k(C) \leq T\}$.*

Proof. We first prove the inclusion $\mathcal{F}_C \subseteq \{\sum_k \tau_k(C) \leq T\}$. At the time T' defined in $\mathcal{F}_C$, it holds $\min_k I^k_{N_{T'}} \geq C$. The results then follows from Lemma A.2: $\sum_k \tau_k(C) \leq \sum_k N_{k,T'} = T' \leq T$.

We now prove $\{\sum_k \tau_k(C) \leq T\} \subseteq \mathcal{F}_C$. If there is no j with $N_{j,T} > \tau_j(C)$, we have $T = \sum_k N_{k,T} \leq \sum_k \tau_k(C) \leq T$. Hence there is equality and we have $N_{k,T} = \tau_k(C)$ for all k and $\mathcal{F}_C$ is true for $T' = T$.

If there is some j such that $N_{j,T} > \tau_j(C)$, then after the time at which arm j was pulled $\tau_j(C)$ times it verified $I^j_{N_{j,t}} \geq C$. Arm j is again pulled at least once at some time t', and at that time we have by Lemma A.1 that for all k, $I^k_{N_{k,t'-1}} \geq I^j_{N_{j,t'-1}} = I^j_{N_{j,t}} \geq C$. Stated otherwise, the event $\mathcal{F}_C$ happens. □

Lemma A.4. *Let $t_{\max} = \arg\max_{t\in[T]} \min_{k\in[K]} I^k_{N_{k,t}}$. Then for all arms except at most one, $N_{k,t_{\max}} = N_{k,T}$.*

Proof. The algorithm switches arm only if the index of the pulled arm becomes strictly greater than the minimal index of the others. As a consequence, the value of the minimal index at times of arm changes is increasing. If two or more arms are pulled since $t_{\max}$, there is an arm change later than $t_{\max}$ and the minimal index value at that time is higher than at $t_{\max}$. This is a contradiction. □

A.2 Concentration lemmas

First, we state a simple but useful lemma.

Lemma A.5. *Let* $\Delta_{n,k}^{\delta} = \Delta_k - \sqrt{\frac{\log(1/\delta)}{n}}$. *For all* $\delta \in (0,1)$ *such that* $\Delta_{n,k}^{\delta} \geq 0$, *Hoeffding's inequality implies that* $\mathbb{P}(\hat{\Delta}_{n,k} < \Delta_{n,k}^{\delta}) \leq \delta$.

The next lemma will be used to bound the sum of exponentially tailed distributions.

Lemma A.6 ((Janson, 2018))**.** *Let* $Z_1, \ldots, Z_K$ *be independent random variables and* $a_1, \ldots, a_K \in \mathbb{R}^+$ *be such that for all* $k \in [K]$ *and* $x \in \mathbb{R}^+$, $\mathbb{P}(Z_k \geq x) \leq e^{-a_k x}$. *Then for all* $\lambda \geq 0$,

$$\mathbb{P}(\sum_k Z_k \geq \lambda \sum_k \frac{1}{a_k}) \leq e^{1-\lambda} .$$

Corollary A.7. *Let* $Y_1, \ldots, Y_K$ *be independent random variables and* $y_1, \ldots, y_K \in \mathbb{R}$, $a_1, \ldots, a_K \in \mathbb{R}^+$ *be such that for all* $k \in [K]$ *and* $x \in \mathbb{R}^+$, $\mathbb{P}(Y_k \geq y_k + x) \leq e^{-a_k x}$. *Then for all* $x \geq \sum_k y_k$,

$$\mathbb{P}(\sum_k Y_k \geq x) \leq e \times \exp\left(-\frac{x - \sum_k y_k}{\sum_k 1/a_k}\right) .$$

The corollary is a direct application of Lemma A.6 to $Z_k = Y_k - y_k$.

Appendix B

Complements on Chapter 4

Contents

B.1 Technical results

Lemma B.1. (*Tail inequality*) *For all $\delta > 0, \sigma' > 0$, with probability at least $1 - \delta$, for all $T > 0$:*

$$|S_T| \leq \sqrt{2(A_T + 1/\sigma'^2) \log\left(\frac{\sqrt{\sigma'^2 A_T + 1}}{\delta}\right)}$$

Proof. We use the method of mixtures, denote

$$M_t^\lambda = \exp\left(\lambda \varepsilon_t (\theta_T - \theta_*)^\top x_t - \frac{\lambda^2}{2}\left((\theta_T - \theta_*)^\top x_t\right)^2\right).$$

Without loss of generality, we can assume that $(\varepsilon_s)_{s\geq 1}$ is 1-sub-Gaussian (this can be achieved by scaling features appropriately), then $\mathbb{E}[M_t^\lambda] \leq 1$.
Let $\Lambda \sim \mathbb{N}(0, \sigma'^2)$ be a Gaussian random variable and define $M_t = \mathbb{E}[M_t^\Lambda | F^\infty]$. We have $\mathbb{E}[M_t] = \mathbb{E}[\mathbb{E}[M_t^\Lambda | \Lambda]] \leq 1$. By making explicit M_t and using Markov's inequality we get that for any stopping time τ, for all $\delta > 0$, with probability at least $1 - \delta$:

$$\frac{|S_\tau|^2}{1/\sigma'^2 + A_\tau} \leq 2\sigma^2 \log\left(\frac{\sqrt{1 + \sigma'^2 A_\tau}}{\delta}\right).$$

We conclude using the same stopping time construction in Proof B.1. □

Theorem. (*Theorem 3.3 of (Maillard, 2016)*) (*Ordinary Least-squares*) *Assume that N is a stopping time adapted to the filtration of the past. Then in the sub-Gaussian streaming regression model, for any $\delta > 0$, with probability at least $1 - \delta, \forall T \geq 1$ if $|G_T(0)| > 0$:*

$$\|\theta_* - \theta_T\|^2_{G_T(0)} \leq 2(1+\kappa)(1+\alpha)\sigma^2 \log \frac{\kappa_d(e^2 \lambda_{max}(G_T))}{\delta}$$

where $\kappa_d(x)$ is function of κ and α, $\kappa_d(x) = \frac{2}{3}\pi^2 \log(x/e)^2 \left\lceil \frac{\log(x)}{2} \right\rceil \left[(12(d+1)\sqrt{d})^d x^d + d\right]$ for $\kappa = \alpha = 1$.

Lemma B.2. (*Technical inequality*) *For all sequences $\{x_t\}_t \in \mathbb{R}^d$ such that $\forall t, \|x_t\|_2 \leq X$, for all $\lambda \in \mathbb{R}_+, T_0, T \in \mathbb{N}$*

$$\sum_{t=T_0}^{T} \|x_t\|^2_{G_t^{-1}} \leq d \log\left(1 + TX^2/\lambda_{min}(G_{T_0})d\right)$$

where $G_t = G_t(\lambda)$.

Proof. Using the Weinstein–Aronszajn identity: $\|x_t\|^2_{G_t^{-1}} = 1 - \frac{|G_{t-1}|}{|G_t|}$, and that $z - 1 \geq \log(z)$ leads to:

$$\sum_{t=T_0}^{T} \|x_t\|^2_{G_t^{-1}} \leq \sum_{t=1}^{T} -\log \frac{|G_{t-1}|}{|G_t|} = \log\left(\frac{|G_T|}{|G_{T_0}|}\right).$$

Since $\|x_t\|_2 \leq X$, using the AM-GM inequality:

$$\sum_{t=T_0}^{T} \log\left(1 + \|x_t\|^2_{G_{t-1}^{-1}}\right) \leq d \log\left(1 + TX^2/\lambda_{min}(G_{T_0})d\right).$$

□

Lemma B.3. (*Technical inequality, Ridge regression*) *For all sequences* $\{x_t\}_t \in \mathbb{R}^d$ *such that* $\forall t, \|x_t\|_2 \leq X$, *for all* $\lambda \in \mathbb{R}_+, T \in \mathbb{N}$

$$\sum_{t=1}^{T} ||x_t||^2_{G_{t-1}^{-1}} \leq \frac{X^2/\lambda}{\log(1 + X^2/\lambda)} d \log\left(1 + TX^2/\lambda d\right)$$

Proof. We use the Weinstein–Aronszajn identity: $\|x_t\|^2_{G_{t-1}^{-1}} = \frac{|G_t|}{|G_{t-1}|} - 1$, which leads to:

$$\sum_{t=1}^{T} \log\left(1 + \|x_t\|^2_{G_{t-1}^{-1}}\right) = \log\left(\frac{G_T}{G_0}\right).$$

Then since $\|x_t\|_2 \leq X$ and using the AM-GM inequality:

$$\sum_{t=1}^{T} \log\left(1 + \|x_t\|^2_{G_{t-1}^{-1}}\right) \leq d \log\left(1 + TX^2/\lambda d\right).$$

This next part is what differs from Lemma B.2, using $||x_t||^2_{G_{t-1}^{-1}} \leq \lambda_{\max}(G_{t-1}^{-1})||x_t||_2^2 \leq X^2/\lambda$ and the concavity of the function $\log$ we find:

$$\sum_{t=1}^{T} \|x_t\|^2_{G_{t-1}^{-1}} \leq \sum_{t=1}^{T} \frac{X^2/\lambda}{\log(1 + X^2/\lambda)} \log\left(1 + \|x_t\|^2_{G_{t-1}^{-1}}\right).$$

The last inequality can also be proved by noting that $x \to x/\log(1+x)$ is non-decreasing which can be used to bound every feature norm. □

Lemma B.4. (*Tail inequality, see Corollary 8 of (Abbasi-Yadkori, Pal, and Szepesvari, 2012)*) *Define* $S_t = \sum_{s=1}^{t} \varepsilon_s(\theta_{s-1} - \theta_*)^\top x_s$ *and let* $(F_t)_{t\geq 0}$ *be a filtration such that* x_t *is* F_{t-1} *measurable and* ε_t *is* F_t *measurable. Then* S_t *is a martingale with respect to* F_t *and for any* $\delta > 0, \sigma' > 0$, *with probability at least* $1 - \delta$, *for all* $t \geq 0$:

$$|S_t| \leq \sigma \sqrt{2\left(1/\sigma'^2 + \sum_{s=1}^{t} \left((\theta_{t-1} - \theta_*)^\top x_t\right)^2\right) \log\left(\frac{\sqrt{1 + \sigma'^2 \sum_{s=1}^{t} \left((\theta_{t-1} - \theta_*)^\top x_t\right)^2}}{\delta}\right)}$$

Proof. The proof of this result follows the same line in the proof of Theorem 1 of Abbasi-Yadkori, Pál, and Szepesvári (2011), first we define for $\lambda \in \mathbb{R}^d, t > 0 : M_t^\lambda = \exp\left(\sum_{s=1}^t \left[\varepsilon_s \lambda (\theta_{t-1} - \theta_*)^\top x_t - \lambda^2 \left((\theta_{t-1} - \theta_*)^\top x_t \right)^2 / 2 \right] \right)$.
Without loss of generality, we can assume that $(\varepsilon_s)_{s \geq 1}$ is 1-sub-Gaussian (this can be achieved by scaling features). Let τ be a stopping time with respect to the filtration $\{F_t\}_{t=0}^\infty$. Then M_τ^λ is well-defined almost surely and

$$\mathbb{E}[M_\tau^\lambda] \leq 1.$$

Let $\Lambda \sim \mathbb{N}(0, \sigma'^2)$ be a Gaussian random variable and define $M_t = \mathbb{E}[M_t^\Lambda | F^\infty]$. We have $\mathbb{E}[M_t] = \mathbb{E}[\mathbb{E}[M_t^\Lambda | \Lambda]] \leq 1$. By expliciting M_t and using Markov's inequality we get that for $\delta > 0$, with probability $1 - \delta$:

$$|S_\tau|^2 \leq \left(1/\sigma'^2 + \sum_{t=1}^\tau \left((\theta_{t-1} - \theta_*)^\top x_t \right)^2 \right) 2\sigma^2 \log \left(\frac{\sqrt{1 + \sigma'^2 \sum_{t=1}^\tau \left((\theta_{t-1} - \theta_*)^\top x_t \right)^2}}{\delta} \right). \quad \text{(B.1)}$$

Next we use a stopping time construction from Freedman (1975): Define the bad event:

$$B_t(\delta) = \left\{ \omega \in \Omega : \frac{|S_t|^2}{1/\sigma'^2 + \sum_{s=1}^t \left((\theta_{s-1} - \theta_*)^\top x_s \right)^2} > 2\sigma^2 \log \left(\frac{\sqrt{1 + \sigma'^2 \sum_{s=1}^t \left((\theta_{s-1} - \theta_*)^\top x_s \right)^2}}{\delta} \right) \right\}$$

We are interested in bounding the probability that $\bigcup_{t>0} B_t(\delta)$ happens. Define $\tau(\omega) = \min\{t \geq 0 : \omega \in B_t(\delta)\}$, with the convention that $\min \emptyset = \infty$. Then, τ is a stopping time. Further,

$$\bigcup_{t \geq 0} B_t(\delta) = \{\omega : \tau(\omega) < \infty\}$$

Thus, by Equation B.1:

$$\Pr\left[\bigcup_{t \geq 0} B_t(\delta) \right] = \Pr[\tau < \infty] = \Pr\left[B_\tau(\delta), \tau < \infty \right] \leq \Pr\left[B_\tau(\delta) \right] \leq \delta$$

□

Theorem B.5. (*Confidence ellipsoid for the Forward algorithm*) *For any $\delta > 0$, with probability at least $1 - \delta$, for allt > 0:*

$$\|\theta_t - \theta_*\|_{G_t} \leq \sqrt{\beta_t(\delta)} = \sigma \sqrt{d \log \left(\frac{1 + tX^2/\lambda d}{\delta} \right)} + (\lambda^{1/2} + X)S.$$

Proof. Denote $X_t = (x_1^\top, \dots, x_t^\top), \varepsilon_t = (\varepsilon_1, \dots, \varepsilon_t)^\top$. Using

$$\begin{aligned}\theta_t &= G_{t+1}^{-1} X_t^\top (X\theta_* + \varepsilon_t) \\ &= G_{t+1}^{-1} X_t^\top \varepsilon_t + G_{t+1}^{-1}(X_t^\top X_t + \lambda I + x_{t+1}^\top x_{t+1})\theta_* - G_{t+1}^{-1}(\lambda I + x_{t+1}^\top x_{t+1})\theta_* \\ &= G_{t+1}^{-1} X_t^\top \varepsilon_t + \theta_* - G_{t+1}^{-1}(\lambda I + x_{t+1}^\top x_{t+1})\theta_*,\end{aligned}$$

we get

$$\begin{aligned}|x^\top \theta_t - x^\top \theta_*| &= |x^\top G_{t+1}^{-1} X_t \varepsilon_t - x^\top G_{t+1}^{-1}(\lambda\theta_* + x_{t+1} x_{t+1}^\top \theta_*)| \\ &\le \|x\|_{G_{t+1}^{-1}} \Big(\|X_t^\top \varepsilon_t\|_{G_{t+1}^{-1}} + (\sqrt{\lambda} + X)\|\theta_*\|_2 \Big),\end{aligned}$$

where in the last inequality we used Cauchy-Schwartz inequality and that by the Sherman-Morrison formula $x_{t+1}^\top G_{t+1}^{-1} x_{t+1} = \frac{x_{t+1}^\top G_t^{-1} x_{t+1}}{1 + x_{t+1}^\top G_t^{-1} x_{t+1}} \le 1$.

We know that: $||X_t^\top \varepsilon_t||_{G_{t+1}^{-1}} \le ||X_t^\top \varepsilon_t||_{G_t^{-1}}$ which allows us to use Theorem 1 from Abbasi-Yadkori, Pál, and Szepesvári (2011) that we recall directly after this proof. We conclude by plugging in $x = G_{t+1}(\theta_t - \theta_*)$. □

Theorem. (*Self-Normalized Bound for Vector-Valued Martingales*). *Let $\{F_t\}_{t=0}^\infty$ be a filtration. Let $\{\eta_t\}_{t=1}^\infty$ be a real-valued stochastic process such that η_t is F_t -measurable and η_t is conditionally R -sub-Gaussian for some $R \ge 0$ i.e.*

$$\forall \lambda \in \mathbb{R} \quad \mathbf{E}\left[e^{\lambda \eta_t} \mid F_{t-1}\right] \le \exp\left(\frac{\lambda^2 R^2}{2}\right)$$

Let $\{X_t\}_{t=1}^\infty$ be an $\mathbb{R}^d$ -valued stochastic process such that X_t is F_{t-1} -measurable. Assume that V is a $d \times d$ positive definite matrix. For any $t \ge 0$, define

$$\bar{V}_t = V + \sum_{s=1}^t X_s X_s^\top \quad S_t = \sum_{s=1}^t \eta_s X_s.$$

Then, for any $\delta > 0$, with probability at least $1 - \delta$, for all $t \ge 0$,

$$\|S_t\|_{V_t^{-1}}^2 \le 2R^2 \log\left(\frac{\det\left(\bar{V}_t\right)^{1/2} \det(V)^{-1/2}}{\delta}\right).$$

Note that the deviation of the martingale $\|S_t\|_{\bar{V}_t^{-1}}^2$ is measured by the norm weighted by the matrix $\bar{V}_t^{-1}$ which is itself derived from the martingale, hence the name "self-normalized bound".

Lemma B.6. (*Tail inequality, Forward algorithm*) *Define $S_t = \sum_{s=1}^t \varepsilon_s(\theta_{s-1} - \theta_*)^\top x_s$ and let $(F_t)_{t\ge 0}$ be a filtration such that x_t is F_{t-1} measurable and ε_t is F_t measurable. Then S_t is a martingale with*

respect to F_t and for any $\delta > 0, \sigma' > 0$, with probability at least $1-\delta$, for all $t \geq 0$:

$$|S_t| \leq \sigma \sqrt{2\left(1/\sigma'^2 + \sum_{s=1}^{t} \left((\theta_{t-1} - \theta_*)^\top x_t\right)^2\right) \log\left(\frac{\sqrt{1+\sigma'^2 \sum_{s=1}^{t} \left((\theta_{t-1} - \theta_*)^\top x_t\right)^2}}{\delta}\right)}$$

Proof. The proof of this result proceeds in the exact same way as for Lemma B.4. □

B.2 Experimental details and instructions:

The experiments were run on a personal laptop with Intel Core i7-8665U, CPU 1.90GHz × 8. Code for the experiments for online regression and linear bandits can be provided upon request to the authors. For the experiments of non-stationary linear bandits that we present next, we used an existing code from the Github page of Russac, Vernade, and Cappé (2019) and we added an implementation of `D-LinUCB` [f] to compare with previous algorithms.

Appendix C

Complements on Chapter 5

Contents

C.1 Concentrations

C.1.1 Concentration of the transition parameter

We recall the important concentration of the maximum likelihood estimator for general bilinear exponential families (*cf* Theorem 1 of (Chowdhury, Gopalan, and Maillard, 2021)).

Theorem C.1. *Suppose $\{\mathcal{F}_t\}_{t=0}^{\infty}$ is a filtration such that for each t, (i) s_{t+1} is $\mathcal{F}_t$-measurable, (ii) (s_t, a_t) is $\mathcal{F}_{t-1}$ measurable, and (iii) given (s_t, a_t), $s_{t+1} \sim P_{\theta^p}^p(\cdot \mid s_t, a_t)$ according to the exponential family defined by Equation (5.1). Let $\hat{\theta}^p(k)$ be the penalized MLE defined by Equation (5.6), and let $Z_{s,a}^p(\theta)$ be strictly convex in θ for all (s, a). Then, for any $\delta \in (0, 1]$, with probability at least $1 - \delta$, the following holds uniformly over all $n \in \mathbb{N}$:*

$$\sum_{t=1}^{k} \mathrm{KL}_{s_t,a_t}\left(\hat{\theta}^p(k), \theta^p\right) + \frac{\eta}{2}\left\|\theta^p - \hat{\theta}^p(k)\right\|_{\mathbb{A}}^2 - \frac{\eta}{2}\left\|\theta^p\right\|_{\mathbb{A}}^2 \leq \log\left(\frac{C_{\mathrm{A},k}^p}{\delta}\right),$$

where $C_{\mathrm{A},k}^p = \left(\int_{\mathbf{R}^d} \exp\left(-\frac{\eta}{2}\left\|\theta'\right\|_{\mathbb{A}}^2\right) d\theta'\right) / \left(\int_{\mathbf{R}^d} \exp\left(-\sum_{t=1}^{k} \mathrm{KL}_{s_t,a_t}\left(\theta_k, \theta'\right) - \frac{\eta}{2}\left\|\theta' - \theta_k\right\|_{\mathbb{A}}^2\right) d\theta'\right)$. Define $G_{s,a} \triangleq \left(\varphi(s,a)^\top A_i^\top A_j \varphi(s,a)\right)_{i,j\in[d]}$, we have

$$C_{\mathbb{A},k}^p \leq \det\left(I + \beta^p \eta^{-1} \mathbb{A}^{-1} \sum_{t=1}^{k} G_{s_t,a_t}\right),$$

where $\beta^p = \sup_{\theta,s,a} \lambda_{\max}\left(\mathbb{C}_{s,a}^{\theta}\left[\psi\left(s'\right)\right]\right)$.

A proof of this result can be found in the work (Chowdhury, Gopalan, and Maillard, 2021). We provide an almost similar proof for the concentration of rewards in the next section.

Corollary C.2. *The previous theorem implies a simple euclidean confidence region. Indeed, with probability at least $1 - \delta$, for all $k \in \mathbf{N}$*

$$\left\|\theta^p - \hat{\theta}^p(k)\right\|_{\bar{G}_n^p}^2 \leq \frac{2}{\alpha^p}\beta^p(k, \delta),$$

where $\beta^p(k, \delta) \triangleq \beta_{(k-1)H}^p(\delta) = \frac{2}{2}B_A^2 + \log\left(2C_{A,k}^p/\delta\right)$.

Proof. The result follows from the following simple calculations:

$$\begin{aligned}\frac{1}{2}\left\|\theta^{\mathtt{p}} - \hat{\theta}^{\mathtt{p}}(k)\right\|_{\bar{G}_k}^2 &= \frac{(\alpha^{\mathtt{p}})^{-1}\eta}{2}\left\|\theta^{\mathtt{p}} - \hat{\theta}^{\mathtt{p}}(k)\right\|_{\mathbb{A}}^2 + \sum_{\tau=1}^{k-1}\sum_{h=1}^{H}\frac{1}{2}\left\|\theta^{\mathtt{p}} - \hat{\theta}^{\mathtt{p}}(k)\right\|_{G_{s_h^\tau,a_h^\tau}}^2 \\ &\leq (\alpha^{\mathtt{p}})^{-1}\left(\frac{\eta}{2}\left\|\theta^{\mathtt{p}} - \hat{\theta}^{\mathtt{p}}(k)\right\|_{\mathbb{A}}^2 + \sum_{\tau=1}^{k-1}\sum_{h=1}^{H}\mathrm{KL}_{s_h^\tau,a_h^\tau}\left(\theta_k, \theta\right)\right).\end{aligned}$$

□

C.1.2 Concentration of the reward parameter (contribution)

Theorem C.3. *Suppose $\{\mathcal{F}_t\}_{t=0}^{\infty}$ is a filtration such that for each t, (i) $r(s_t, a_t)$ is $\mathcal{F}_t$-measurable, (ii) (s_t, a_t) is $\mathcal{F}_{t-1}$ measurable, and (iii) given (s_t, a_t), $r(s_t, a_t) \sim P^{\mathtt{r}}_{\theta^{\mathtt{r}}}(\cdot \mid s_t, a_t)$ according to the exponential family defined by (5.2). Let $\hat{\theta}^{\mathtt{r}}(k)$ be the penalized MLE defined by Equation (5.7), and let $Z^{\mathtt{r}}_{s,a}(\theta)$ be strictly convex in θ for all (s, a). Then, for any $\delta \in (0, 1]$, with probability at least $1 - \delta$, the following holds uniformly over all $k \in \mathbb{N}$:*

$$\sum_{t=1}^{k} \mathrm{KL}_{s_t,a_t}\left(\hat{\theta}^{\mathtt{r}}(k), \theta^{\mathtt{r}}\right) + \frac{\eta}{2}\left\|\theta^{\mathtt{r}} - \hat{\theta}^{\mathtt{r}}(k)\right\|^2_{\mathbb{A}} - \frac{\eta}{2}\left\|\theta^{\mathtt{r}}\right\|^2_{\mathbb{A}} \leq \log\left(\frac{C^{\mathtt{r}}_{\mathrm{A},k}}{\delta}\right),$$

where $C^{\mathtt{r}}_{\mathrm{A},k} = \left(\int_{\mathbf{R}^d} \exp\left(-\frac{\eta}{2}\left\|\theta'\right\|^2_{\mathbb{A}}\right) d\theta'\right) / \left(\int_{\mathbf{R}^d} \exp\left(-\sum_{t=1}^{k} \mathrm{KL}_{s_t,a_t}(\theta_k, \theta') - \frac{\eta}{2}\left\|\theta' - \theta_k\right\|^2_{\mathbb{A}}\right) d\theta'\right)$. Define $G_{s,a} \triangleq \left(\varphi(s,a)^\top A_i^\top A_j \varphi(s,a)\right)_{i,j\in[d]}$, we have

$$C_{\mathbb{A},k} \leq \det\left(I + \beta^{\mathtt{r}} \eta^{-1} \mathbb{A}^{-1} \sum_{t=1}^{k} G_{s_t,a_t}\right),$$

where $\beta^{\mathtt{r}} := \|B\|_2^2 \sup_{\theta,s,a} \mathbb{V}\mathrm{ar}^{\theta}_{s,a}(r)$.

Proof. We proceed similar to the proof of Theorem 1 in (Chowdhury and Gopalan, 2019).

Step 1: Martingale construction. First, observe that by assuming strict convexity, the log-partition function $Z^{\mathtt{r}}_{s,a}$ becomes a Legendre function. Now for the conditional exponential family model, the KL divergence between $\mathbb{P}^{\mathtt{r}}_{\theta^{\mathtt{r}}}(\cdot \mid s, a)$ and $\mathbb{P}^{\mathtt{r}}_{\theta'^{\mathtt{r}}}(\cdot \mid s, a)$ can be expressed as a Bregman divergence associated to $Z^{\mathtt{r}}_{s,a}$ with the parameters reversed, i.e.

$$\mathrm{KL}_{s,a}\left(\theta^{\mathtt{r}}, \theta'^{\mathtt{r}}\right) := \mathrm{KL}\left(P_{\theta^{\mathtt{r}}}(\cdot \mid s, a), P_{\theta'^{\mathtt{r}}}(\cdot \mid s, a)\right) = B_{Z_{s,a}}\left(\theta'^{\mathtt{r}}, \theta^{\mathtt{r}}\right).$$

Now, for any $\lambda \in \mathbb{R}^d$, we introduce the function $B_{Z_{n,\alpha},\theta^{\mathtt{r}}}(\lambda) = B_{Z_{n,\alpha}}\left(\theta^{\mathtt{r}} + \lambda, \lambda\right)$ and define

$$M_n^{\lambda} = \exp\left(\lambda^\top S_n - \sum_{t=1}^{n} B_{Z_{n_t,a_t},\theta^{\mathtt{r}}}(\lambda)\right)$$

where $\forall i \leq d$, we denote $(S_n)_i = \sum_{t=1}^{n}\left(r\left(s_t, a_t\right) - \mathbb{E}^{\theta^{\mathtt{r}}}_{s_t,a_t}[r]\right) B^\top A_i \varphi\left(s_t, a_t\right)$. Note that $M_n^{\lambda} > 0$ and it is $\mathcal{F}_{n^-}$ measurable. Furthermore, we have for all (s, a),

$$\mathbb{E}^{\theta^{\mathtt{r}}}_{s,a}\left[\exp\left(\sum_{i=1}^{d} \lambda_i \left(r\left(s_t, a_t\right) - \mathbb{E}^{\theta^{\mathtt{r}}}_{s_t,a_t}[r]\right) B^\top A_i \varphi\left(s_t, a_t\right)\right)\right]$$

$$= \exp\left(-\lambda^\top \nabla Z^{\mathtt{r}}_{s,a}\left(\theta^{\mathtt{r}}\right)\right) \int_{\mathcal{S}} \exp\left(\sum_{i=1}^{d} \left(\theta^{\mathtt{r}}_i + \lambda_i\right) B^\top A_i \varphi(s,a) - Z^{\mathtt{r}}_{s,a}(\theta^{\mathtt{r}})\right) dr$$
$$= \exp\left(Z^{\mathtt{r}}_{s,a}(\theta^{\mathtt{r}} + \lambda) - Z^{\mathtt{r}}_{s,a}(\theta^{\mathtt{r}}) - \lambda^\top \nabla Z^{\mathtt{r}}_{s,a}(\theta^{\mathtt{r}})\right) = \exp\left(B_{Z^{\mathtt{r}}_{s,a}}(\theta^{\mathtt{r}})\right)$$

This implies $\mathbb{E}\left[\exp\left(\lambda^\top S_n\right) \mid \mathcal{F}_{n-1}\right] = \exp\left(\lambda^\top S_{n-1} + B_{Z_{n n,a n,\theta^{\mathtt{r}}}}(\lambda)\right)$ thus $\mathbb{E}\left[M^\lambda_n \mid \mathcal{F}_{n-1}\right] = M^\lambda_{n-1}$. Therefore $\left\{M^\lambda_n\right\}_{n=0}^{\infty}$ is a non-negative martingale adapted to the filtration $\{\mathcal{F}_n\}_{n=0}^{\infty}$ and actually satisfies $\mathbb{E}\left[M^\lambda_n\right] = 1$. For any prior density $q(\theta)$ for θ, we now define a mixture of martingales

$$M_n = \int_{\mathbb{R}^d} M^\lambda_n q\left(\theta^{\mathtt{r}} + \lambda\right) d\lambda \tag{C.1}$$

Then $\{M_n\}_{n=0}^{\infty}$ is also a non-negative martingale adapted to $\{\mathcal{F}_n\}_{n=0}^{\infty}$ and in fact, $\mathbb{E}\left[M_n\right] = 1$.

Step 2: Method of mixtures. Considering the prior density $\mathcal{N}(0, (\eta \mathbf{A})^{-1})$, we obtain from (C.1) that

$$M_n = c_0 \int_{\mathbb{R}^d} \exp\left(\lambda^\top S_n - \sum_{t=1}^{n} B_{Z^{\mathtt{r}}_{x_t,a_t},\theta^{\mathtt{r}}}(\lambda) - \frac{\eta}{2}\left\|\theta^{\mathtt{r}} + \lambda\right\|^2_{\mathbf{A}}\right) d\lambda, \tag{C.2}$$

where $c_0 = \frac{1}{\int_{\mathbf{R}^d} \exp\left(-\frac{\eta}{2}\|\theta'\|^2_\Lambda\right) d\theta'}$. We now introduce the function $Z^{\mathtt{r}}_n(\theta) = \sum_{t=1}^{n} Z^{\mathtt{r}}_{s_t,a_t}(\theta)$. Note that $Z^{\mathtt{r}}_n$ is a also Legendre function and its associated Bregman divergence satisfies

$$B_{Z^{\mathtt{r}}_n}\left(\theta', \theta\right) = \sum_{t=1}^{n} \left(Z^{\mathtt{r}}_{s_t,a_t}\left(\theta'\right) - Z^{\mathtt{r}}_{s_t,a_t}(\theta) - \left(\theta' - \theta\right)^\top \nabla Z^{\mathtt{r}}_{S_t,a_t}(\theta)\right) = \sum_{t=1}^{n} B_{Z^{\mathtt{r}}_{s_t,\alpha_t}}\left(\theta', \theta\right)$$

Furthermore, we have $\sum_{t=1}^{n} B_{Z^{\mathtt{r}}_{s_t,\alpha_t},\theta^{\mathtt{r}}}(\lambda) = B_{Z^{\mathtt{r}}_n,\theta^{\mathtt{r}}}(\lambda)$. From the penalized likelihood formula (5.7), recall that

$$\forall i \leq d, \quad \sum_{t=1}^{n} \nabla_i Z^{\mathtt{r}}_{s_t,a_t}\left(\hat{\theta}^{\mathtt{r}}(k)\right) + \frac{\eta}{2} \nabla_i \|\hat{\theta}^{\mathtt{r}}(k)\|^2_{\mathbf{A}} = \sum_{t=1}^{k} r_t B^\top A_i \varphi\left(s_t, a_t\right).$$

This yields

$$S_k = \sum_{t=1}^{k} \left(\nabla Z^{\mathtt{r}}_{s_t,a_t}\left(\hat{\theta}^{\mathtt{r}}(k)\right) - \nabla Z^{\mathtt{r}}_{s_t,a_t}\left(\theta^{\mathtt{r}}\right)\right) + \eta \mathbf{A} \hat{\theta}^{\mathtt{r}}(k) = \nabla Z^{\mathtt{r}}_k\left(\hat{\theta}^{\mathtt{r}}(k)\right) - \nabla Z^{\mathtt{r}}_k\left(\theta^{\mathtt{r}}\right) + \eta \mathbf{A} \hat{\theta}^{\mathtt{r}}(k) \tag{C.3}$$

We now obtain from (C.2) and (C.3) that

$$M_k = c_0 \cdot \exp\left(-\frac{\eta}{2}\left\|\theta^{\mathtt{r}}\right\|^2_A\right) \int_{\mathbb{R}^d} \exp\left(\lambda^\top x_k - B_{Z_k,\theta^*}(\lambda) + g_k(\lambda)\right) d\lambda, \tag{C.4}$$

where we introduced $g_k(\lambda) = \frac{\eta}{2}\left(2\lambda^\top \mathbf{A} \hat{\theta}^{\mathtt{r}}(k) + \|\theta^{\mathtt{r}}\|^2_{\mathbf{A}} - \|\theta^{\mathtt{r}} + \lambda\|^2_{\mathbf{A}}\right)$ and $x_k = \nabla Z^{\mathtt{r}}_k\left(\hat{\theta}^{\mathtt{r}}(k)\right) - \nabla Z^{\mathtt{r}}_k\left(\theta^{\mathtt{r}}\right)$.

Now, note that $\sup_{\lambda\in\mathbb{R}^d} g_k(\lambda) = \frac{\eta}{2}\left\|\theta^{\mathtt{r}} - \hat{\theta}^{\mathtt{r}}(k)\right\|_{\mathbf{A}}^2$, where the supremum is attained at $\lambda^\star = \hat{\theta}^{\mathtt{r}}(k) - \theta^{\mathtt{r}}$. We then have

$$\begin{aligned} g_k(\lambda) &= g_n(\lambda) + \sup_{\lambda\in\mathbb{R}^\star} g_k(\lambda) - g_k\left(\lambda^\star\right) \\ &= \frac{\eta}{2}\left\|\hat{\theta}^{\mathtt{r}}(k) - \theta^{\mathtt{r}}\right\|_{\mathbf{A}}^2 + \eta\left(\lambda-\lambda^\star\right)^\top \mathbf{A}\left(\theta^{\mathtt{r}}+\lambda^\star\right) + \frac{\eta}{2}\left\|\theta^{\mathtt{r}}+\lambda^\star\right\|_A^2 - \frac{\eta}{2}\left\|\theta^{\mathtt{r}}+\lambda\right\|_{\mathbf{A}}^2 \\ &= B_{Z_0^{\mathtt{r}}}\left(\theta^{\mathtt{r}}, \hat{\theta}^{\mathtt{r}}(k)\right) + \left(\lambda-\lambda^\star\right)^\top \nabla Z_0^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^\star\right) + Z_0^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^\star\right) - Z_0^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda\right) \end{aligned} \tag{C.5}$$

where we have introduced the Legendre function $Z_0^{\mathtt{r}}(\theta) = \frac{\eta}{2}\|\theta\|_{\mathbf{A}}^2$. We now have from (C.9) that

$$\begin{aligned} &\sup_{\lambda\in\mathbb{R}^d}\left(\lambda^\top x_n - B_{Z_n^{\mathtt{r}},\theta^{\mathtt{r}}}(\lambda)\right) \\ &\quad = B^\star_{Z_n^{\mathtt{r}},\theta^{\mathtt{r}}}\left(x_n\right) = B^\star_{Z_n^{\mathtt{r}},\theta^{\mathtt{r}}}\left(\nabla Z_n^{\mathtt{r}}\left(\hat{\theta}^{\mathtt{r}}(n)\right) - \nabla Z_n^{\mathtt{r}}\left(\theta^{\mathtt{r}}\right)\right) = B_{Z^{\mathtt{r}}n}\left(\theta^{\mathtt{r}}, \hat{\theta}^{\mathtt{r}}(n)\right). \end{aligned}$$

Further, any optimal λ must satisfy

$$\nabla Z_n^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda\right) - \nabla Z_n^{\mathtt{r}}\left(\theta^{\mathtt{r}}\right) = x_n \Longrightarrow \nabla Z_n^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda\right) = \nabla Z_n^{\mathtt{r}}\left(\hat{\theta}^{\mathtt{r}}(n)\right).$$

One possible solution is $\lambda = \lambda^\star$. Now, since $Z_n^{\mathtt{r}}$ is strictly convex, the supremum is indeed attained at $\lambda = \lambda^\star$. We then have

$$\begin{aligned} &\lambda^\top x_n - B_{Z_n^{\mathtt{r}},\theta^{\mathtt{r}}}(\lambda) \\ &\quad = \lambda^\top x_n - B_{Z_n^{\mathtt{r}},\theta^{\mathtt{r}}}(\lambda) + B_{Z_n^{\mathtt{r}}}\left(\theta^{\mathtt{r}}, \hat{\theta}^{\mathtt{r}}(n)\right) - \left(\lambda^\star x_n - B_{Z_n^{\mathtt{r}},\theta^{\mathtt{r}}}\left(\lambda^\star\right)\right) \\ &\quad = B_{Z_n^{\mathtt{r}}}\left(\theta^{\mathtt{r}}, \hat{\theta}^{\mathtt{r}}(n)\right) + \left(\lambda-\lambda^\star\right)^\top \nabla Z_n^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^\star\right) + B_{Z_n^{\mathtt{r}},\theta^*}\left(\lambda^\star\right) - B_{Z_n^{\mathtt{r}},\theta^*}(\lambda) \\ &\qquad - \left(\lambda-\lambda^\star\right)^\top \nabla Z_n^{\mathtt{r}}\left(\theta^{\mathtt{r}}\right) \\ &\quad = B_{Z_n^{\mathtt{r}}}\left(\theta^{\mathtt{r}}, \hat{\theta}^{\mathtt{r}}(n)\right) + \left(\lambda-\lambda^\star\right)^\top \nabla Z_n^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^\star\right) + Z_n^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^\star\right) - Z_n^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda\right) \end{aligned} \tag{C.6}$$

Plugging Equation (C.5) and Equation (C.6) in Equation (C.4), we obtain

$$\begin{aligned} M_n &= c_0 \cdot \exp\left(\sum_{j\in\{0,n\}} B_{Z_j^{\mathtt{r}}}\left(\theta^{\mathtt{r}}, \theta_j\right) - \frac{\eta}{2}\left\|\theta^{\mathtt{r}}\right\|_A^2\right) \\ &\qquad \times \int_{\mathbb{R}^d} \exp\left(\sum_{j\in\{0,n\}}\left(\left(\lambda-\lambda^\star\right)^\top \nabla Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^\star\right) + Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^\star\right) - Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda\right)\right)\right) d\lambda \\ &= c_0 \cdot \exp\left(\sum_{j\in\{0,n\}} B_{Z_j^{\mathtt{r}}}\left(\theta^{\mathtt{r}}, \hat{\theta}^{\mathtt{r}}(n)\right) - \frac{\eta}{2}\left\|\theta^{\mathtt{r}}\right\|_{\mathring{\mathrm{A}}}^2\right) \end{aligned}$$

$$
\begin{aligned}
&\times \exp\left(-\sum_{j\in\{0,n\}}\left(\left(\theta^{\mathtt{r}}+\lambda^{\star}\right)^{\top}\nabla Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^{\star}\right)-Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^{\star}\right)\right)\right)\\
&\times\int_{\mathbb{R}^d}\exp\left(\sum_{j\in\{0,n\}}\left(\left(\theta^{\mathtt{r}}+\lambda\right)^{\top}\nabla Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^{\star}\right)-Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda\right)\right)\right)d\lambda\\
=&\frac{c_0}{c_{\mathrm{n}}}\exp\left(\sum_{j\in\{0,n\}}B_{Z_j^{\mathtt{r}}}\left(\theta^{\mathtt{r}},\hat{\theta}^{\mathtt{r}}(n)\right)-\frac{\eta}{2}\left\|\theta^{\mathtt{r}}\right\|_{\mathbb{A}}^2\right)\\
&\times\frac{\int_{\mathbb{R}^d}\exp\left(\sum_{j\in\{0,n\}}\left(\left(\theta^{\mathtt{r}}+\lambda\right)^{\top}\nabla Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^{\star}\right)-Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda\right)\right)\right)d\lambda}{\int_{\mathbb{R}^d}\exp\left(\sum_{j\in\{0,n\}}\left(\left(\theta'\right)^{\top}\nabla Z_j^{\mathtt{r}}\left(\theta^{\mathtt{r}}+\lambda^{\star}\right)-Z_j^{\mathtt{r}}\left(\theta'\right)\right)\right)d\theta'}\\
=&\frac{c_0}{c_n}\cdot\exp\left(B_{Z_n}\left(\theta^{\mathtt{r}},\hat{\theta}^{\mathtt{r}}(n)\right)+B_{Z_0}\left(\theta^{\mathtt{r}},\hat{\theta}^{\mathtt{r}}(n)\right)-\frac{\eta}{2}\left\|\theta^{\mathtt{r}}\right\|_{\mathbb{A}}^2\right),
\end{aligned}
$$

where we introduced $c_n=\frac{\exp\left(\sum_{j\in\{0,n\}}\left(\left(\theta^{\mathtt{r}}+\lambda^{\star}\right)^{\top}\nabla Z_j^{\mathtt{r}}(\theta^{\mathtt{r}}+\lambda^{\star})-Z_j^{\mathtt{r}}(\theta^{\mathtt{r}}+\lambda^{\star})\right)\right)}{\int_{\mathbf{R}^d}\exp\left(\sum_{j\in\{0,n\}}\left(\left(\theta'\right)^{\top}\nabla Z_j^{\mathtt{r}}(\theta^{\mathtt{r}}+\lambda^{\star})-Z_j^{\mathtt{r}}(\theta')\right)\right)d\theta'}$.

Since $\lambda^{\star}=\hat{\theta}^{\mathtt{r}}(n)-\theta^{\mathtt{r}}$, we have

$$
\begin{aligned}
c_n&=\frac{1}{\int_{\mathbb{R}^d}\exp\left(-\sum_{j\in\{0,n\}}B_{Z_j^{\mathtt{r}}}\left(\theta',\theta^{\mathtt{r}}+\lambda^{\star}\right)\right)d\theta'}\\
&=\frac{1}{\int_{\mathbb{R}^d}\exp\left(-\sum_{t=1}^n B_{Z_{s_t,a_t}}\left(\theta',\hat{\theta}^{\mathtt{r}}(n)\right)-\frac{\eta}{2}\left\|\theta'-\hat{\theta}^{\mathtt{r}}(n)\right\|_{\mathbb{A}'}^2\right)d\theta'}
\end{aligned}
$$

Therefore, we have from (5) that

$$
C_{A,n}:=\frac{c_n}{c_0}=\frac{\int_{\mathbb{R}^d}\exp\left(-\frac{\eta}{2}\left\|\theta'\right\|_{\mathbb{A}}^2\right)d\theta'}{\int_{\mathbb{R}^d}\exp\left(-\sum_{t=1}^n \mathrm{KL}_{s_t,a_t}\left(\hat{\theta}^{\mathtt{r}}(n),\theta'\right)-\frac{\eta}{2}\left\|\theta'-\hat{\theta}^{\mathtt{r}}(n)\right\|_{\mathbb{A}}^2\right)d\theta'}
$$

An application of Markov's inequality now yields

$$
\begin{aligned}
\mathbb{P}\left[\sum_{t=1}^n \mathrm{KL}_{s_t,a_t}\left(\hat{\theta}^{\mathtt{r}}(n),\theta^{\mathtt{r}}\right)+\frac{\eta}{2}\left\|\theta^{\mathtt{r}}-\hat{\theta}^{\mathtt{r}}(n)\right\|_{\mathbb{A}}^2-\frac{\eta}{2}\left\|\theta^{\mathtt{r}}\right\|_{\mathbb{A}}^2\geq\log\left(\frac{C_{A,n}}{\delta}\right)\right]&=\mathbb{P}\left[M_n\geq\frac{1}{\delta}\right]\\
&\leq\delta\mathbb{E}\left[M_n\right]=\delta
\end{aligned}
$$

Step 3: A stopped martingale and its control. Let N be a stopping time with respect to the filtration $\{\mathcal{F}_n\}_{n=0}^{\infty}$. Now, by the martingale convergence theorem, $M_{\infty}=\lim_{n\to\infty}M_n$ is almost surely well-defined, and thus M_N is well-defined as well irrespective of whether $N<\infty$ or not. Let $Q_n=M_{\min\{N,n\}}$ be a stopped version of $\{M_n\}_n$. Then an application of Fatou's lemma

yields

$$\mathbb{E}\left[M_N\right] = \mathbb{E}\left[\liminf_{n\to\infty} Q_n\right] \leq \liminf_{n\to\infty} \mathbb{E}\left[Q_n\right] = \liminf_{n\to\infty} \mathbb{E}\left[M_{\min\{N,n\}}\right] \leq 1,$$

since the stopped martingale $\left\{M_{\min\{N,n\}}\right\}_{n\geq 1}$ is also a martingale. Therefore, by the properties of M_n, (12) also holds for any random stopping time $N < \infty$. To complete the proof, we now employ a random stopping time construction as in Abbasi-Yadkori et al. (2011)

We define a random stopping time N by

$$N = \min\left\{n \geq 1 : \sum_{t=1}^{n} \mathrm{KL}_{s_t,a_t}\left(\hat{\theta}^{\mathtt{r}}(n), \theta^{\mathtt{r}}\right) + \frac{\eta}{2}\left\|\theta^{\mathtt{r}} - \hat{\theta}^{\mathtt{r}}(n)\right\|_A^2 - \frac{\eta}{2}\left\|\theta^{\mathtt{r}}\right\|_A^2 \geq \log\left(\frac{C_A, n}{\delta}\right)\right\}$$

with $\min\{\emptyset\} := \infty$ by convention. We then have

$$\mathbb{P}\left[\exists n \geq 1, \sum_{t=1}^{n} \mathrm{KL}_{s_t,a_t}\left(\hat{\theta}^{\mathtt{r}}(n), \theta^{\mathtt{r}}\right) + \frac{\eta}{2}\left\|\theta^{\mathtt{r}} - \hat{\theta}^{\mathtt{r}}(n)\right\|_{\mathrm{A}}^2 - \frac{\eta}{2}\left\|\theta^{\mathtt{r}}\right\|_{\mathrm{A}}^2 \geq \log\left(\frac{C_{A,n}}{\delta}\right)\right] = \mathbb{P}[N < \infty] \leq \delta,$$

which concludes the proof of the first part.

Proof of second part: upper bound on $C_{A,n}$. First, we have for some $\tilde{\theta} \in \left[\hat{\theta}^{\mathtt{r}}(n), \theta'\right]_\infty$ that

$$\mathrm{KL}_{s,a}\left(\hat{\theta}^{\mathtt{r}}(n), \theta'\right) = \frac{1}{2}\sum_{i,j=1}^{d}\left(\theta' - \hat{\theta}^{\mathtt{r}}(n)\right)_i \mathbb{V}\mathrm{ar}_{s,a}^{\theta}(r) \times \varphi(s,a)^\top A_i^\top BB^\top A_j \varphi(s,a)\left(\theta' - \hat{\theta}^{\mathtt{r}}(n)\right)_j \tag{C.7}$$

Now (C.7) implies that

$$\begin{aligned}\sum_{t=1}^{n} \mathrm{KL}_{s_t,a_t}\left(\hat{\theta}^{\mathtt{r}}(n), \theta'\right) &\leq \frac{\beta}{2}\sum_{t=1}^{n}\sum_{i,j=1}^{d}\left(\theta' - \hat{\theta}^{\mathtt{r}}(n)\right)_i \varphi\left(s_t,a_t\right)^\top A_i^\top A_j \varphi\left(s_t,a_t\right)\left(\theta' - \hat{\theta}^{\mathtt{r}}(n)\right)_j \\ &= \frac{\beta^{\mathtt{r}}}{2}\left\|\theta' - \hat{\theta}^{\mathtt{r}}(n)\right\|^2_{\sum_{t=1}^{n} G_{s_t,a_t}},\end{aligned}$$

where $\beta^{\mathtt{r}} := \lambda_{\max}\left(BB^\top\right) \times \sup_{\theta,s,a} \mathbb{V}\mathrm{ar}_{s,a}^{\theta}(r)$ and $\forall i,j \leq d,\ \left(G_{s,a}\right)_{i,j} := \varphi(s,a)^\top A_i^\top A_j \varphi(s,a)$. Therefore, we obtain

$$\begin{aligned}C_{\mathrm{A},n} &\leq \frac{\int_{\mathbb{R}^d} \exp\left(-\frac{\eta}{2}\left\|\theta'\right\|_{\mathrm{A}}^2\right) d\theta'}{\int_{\mathbb{R}^d} \exp\left(-\frac{1}{2}\left\|\theta' - \hat{\theta}^{\mathtt{r}}(n)\right\|^2_{\left(\beta^{\mathtt{r}}\sum_{t=1}^{n} G_{s_t,a_t} + \eta\mathrm{A}\right)}\right) d\theta'} \\ &= \frac{(2\pi)^{d/2}}{\det(\eta\mathbb{A})^{1/2}} \times \frac{\det\left(\beta^{\mathtt{r}}\sum_{t=1}^{n} G_{s_t,a_t} + \eta\mathbb{A}\right)^{1/2}}{(2\pi)^{d/2}} = \det\left(I + \beta^{\mathtt{r}}\eta^{-1}\mathbb{A}^{-1}\sum_{t=1}^{n} G_{s_t,a_t}\right),\end{aligned}$$

which completes the proof of the second part.

□

Corollary C.4. *Here also, the theorem implies a euclidean control. With probability at least* $1-\delta$ *uniformly over* $k \in \mathbf{N}$

$$\left\|\theta^r - \hat{\theta}^r(k)\right\|^2_{\bar{G}^r_k} \leq \frac{2}{\alpha^r}\beta^r(k,\delta),$$

where $\beta^r(k,\delta) \triangleq \beta^r_{(k-1)H}(\delta) = \frac{2}{2}B_A^2 + \log\left(2C^r_{A,k}/\delta\right)$.

C.1.3 Gaussian concentration and anti-concentration

Lemma C.5 (Gaussian concentration, ref. Appendix A in (Abeille and Lazaric, 2017))**.** *Let* $\overline{\xi}_{tk} \sim \mathcal{N}(0, H\nu_k(\delta)\Sigma_{tk}^{-1})$. *For any* $\delta > 0$, *with probability* $1-\delta$

$$\|\overline{\xi}_{tk}\|_{\Sigma_{tk}} \leq c\sqrt{Hd\nu_k(\delta)\log(d/\delta)} \tag{C.8}$$

for some absolute constant c.

Lemma C.6 (Gaussian anti-concentration, ref. Appendix A in (Abeille and Lazaric, 2017))**.** *Let* $\xi \sim \mathcal{N}(0, I_d)$, *for any* $u \in \mathbf{R}^d$ *with* $\|u\| = 1$, *we have:*

$$\mathbf{P}(u^\top \xi \geq 1) \geq \Phi(-1),$$

where Φ *is the normal CDF.*

Thanks to lower bounds on the error function, we have the following bound on the probability of anti-concentration $\Phi(-1) \geq 1/(4\sqrt{e\pi})$.

C.2 Technical results

C.2.1 A transportation lemma

For any function $f : \mathcal{X} \to \mathbb{R}$, we define its span as $\mathbb{S}(f) := \max_{x\in\mathcal{X}} f(x) - \min_{x\in\mathcal{X}} f(x)$. For a probability distribution P supported on the set $\mathcal{X}$, let $\mathbb{E}_P[f] := \mathbb{E}_P[f(X)]$ and $\mathbb{V}_P[f] := \mathbb{V}_P[f(X)] = \mathbb{E}_P\left[f(X)^2\right] - \mathbb{E}_P[f(X)]^2$ denote the mean and variance of the random variable $f(X)$, respectively. We now state the following transportation inequalities, which can be adapted from (Boucheron, Lugosi, and Massart, 2013) (Lemma 4.18).

Lemma C.7. (*Transportation inequalities*) *Assume* f *is such that* $S(f)$ *and* $\mathbb{V}_P[f]$ *are finite. Then it holds*

$$\forall Q \ll P, \quad \mathbb{E}_Q[f] - \mathbb{E}_P[f] \leq \sqrt{2\mathbb{V}_P[f]\mathrm{KL}(Q,P)} + \frac{2S(f)}{3}\mathrm{KL}(Q,P)$$
$$\forall Q \ll P, \quad \mathbb{E}_P[f] - \mathbb{E}_Q[f] \leq \sqrt{2\mathbb{V}_P[f]\mathrm{KL}(Q,P)}$$

C.2.2 Bregman divergence

For a Legendre function $F : \mathbb{R}^d \to \mathbb{R}$, the Bregman divergence between $\theta', \theta \in \mathbb{R}^d$ associated with F is defined as $B_F(\theta', \theta) := F(\theta') - F(\theta) - (\theta' - \theta)^\top \nabla F(\theta)$. Now, for any fixed $\theta \in \mathbb{R}^d$, we introduce the function

$$B_{F,\theta}(\lambda) := B_F(\theta + \lambda, \lambda) = F(\theta + \lambda) - F(\theta) - \lambda^\top \nabla F(\theta).$$

It then follows that $B_{F,\theta}$ is a convex function, and we define its dual as

$$B^\star_{F,\theta}(x) = \sup_{\lambda \in \mathbb{R}^d} \left(\lambda^\top x - B_{F,\theta}(\lambda) \right)$$

We have for any $\theta, \theta' \in \mathbb{R}^d$:

$$B_F(\theta', \theta) = B^\star_{F,\theta'}\left(\nabla F(\theta) - \nabla F(\theta')\right) \tag{C.9}$$

To see this, we observe that

$$\begin{aligned} &B^\star_{F,\theta'}\left(\nabla F(\theta) - \nabla F(\theta')\right) \\ &\quad = \sup_{\lambda \in \mathbb{R}^d} \lambda^\top \left(\nabla F(\theta) - \nabla F(\theta')\right) - \left[F(\theta' + \lambda) - F(\theta') - \lambda^\top \nabla F(\theta')\right] \\ &\quad = \sup_{\lambda \in \mathbb{R}^d} \lambda^\top \nabla F(\theta) - F(\theta' + \lambda) + F(\theta'). \end{aligned}$$

Now an optimal λ must satisfy $\nabla F(\theta) = \nabla F(\theta' + \lambda)$. One possible choice is $\lambda = \theta - \theta'$. Since, by definition, F is strictly convex, the supremum will indeed be attained at $\lambda = \theta - \theta'$. Plugin-in this value, we obtain

$$B^\star_{F,\theta'}\left(\nabla F(\theta) - \nabla F(\theta')\right) = (\theta - \theta')^\top \nabla F(\theta) - F(\theta) + F(\theta') = B_F(\theta', \theta).$$

Note that (C.9) holds for any convex function F. Only difference is that, in this case, $B_F(\cdot, \cdot)$ will not correspond to the Bregman divergence.

C.2.3 Properties of the bilinear exponential family

In this section, we detail some useful results related to exponential families in our model.

Derivatives

Lemma C.8. (*Gradients*) *We provide the derivatives of the log-partitions in closed form. As usual with exponential families, these are intimately linked to moments of the random variable. We have:*

$$\left(\nabla_i Z^{\mathtt{p}}_{s,a}\right)(\theta) = \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)\right]^{\top} A_i \varphi(s,a).$$

And

$$\left(\nabla_i Z^{\mathtt{r}}_{s,a}\right)(\theta) = \mathbb{E}^{\theta}_{s,a}\left[r\right]\; B^{\top} A_i \varphi(s,a).$$

Proof. We prove the lemma as follows

$$\begin{aligned}
\left(\nabla_i Z^{\mathtt{p}}_{s,a}\right)(\theta) &= \int_{\mathcal{S}} \psi\left(s'\right)^{\top} A_i \varphi(s,a) \frac{\exp\left(\sum_{i=1}^{d} \theta_i \psi\left(s'\right)^{\top} A_i \varphi(s,a)\right)}{\int_{\mathcal{S}} \exp\left(\sum_{i=1}^{d} \theta_t \psi\left(s'\right)^{\top} A_i \varphi(s,a)\right) ds'} ds' \\
&= \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)\right]^{\top} A_i \varphi(s,a) \\
\left(\nabla_i Z^{\mathtt{r}}_{s,a}\right)(\theta) &= \int_{\mathcal{S}} r B^{\top} A_i \varphi(s,a) \frac{\exp\left(r \sum_{i=1}^{d} \theta_i B^{\top} A_i \varphi(s,a)\right)}{\int_{\mathcal{S}} \exp\left(r \sum_{i=1}^{d} \theta_i B^{\top} A_i \varphi(s,a)\right) dr} dr \\
&= \mathbb{E}^{\theta}_{s,a}\left[r\right]\; B^{\top} A_i \varphi(s,a)
\end{aligned}$$

□

Lemma C.9. (*Hessians*) *The entries of the Hessians of the log partition functions are given by*

$$\left(\nabla^2_{i,j} Z^{\mathtt{p}}_{s,a}\right)(\theta) = \varphi(s,a)^{\top} A_i^{\top} \mathbb{C}^{\theta}_{s,a}\left[\psi\left(s'\right)\right] A_j \varphi(s,a),$$

where $\mathbb{C}^{\theta}_{s,a}\left[\psi\left(s'\right)\right] \triangleq \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)\psi\left(s'\right)^{\top}\right] - \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)\right]\mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)^{\top}\right]$.

Similarly,

$$\left(\nabla^2_{i,j} Z^{r}_{s,a}\right)(\theta) = \mathbb{V}\mathrm{ar}^{\theta}_{s,a}(r) \times \varphi(s,a)^{\top} A_i^{\top} B B^{\top} A_j \varphi(s,a),$$

where $\mathbb{V}\mathrm{ar}^{\theta}_{s,a}(r) \triangleq \left(\mathbb{E}^{\theta}_{s,a}\left[r^2\right] - \mathbb{E}^{\theta}_{s,a}\left[r\right]^2\right)$ *is the variance of the reward under* θ.

Proof. We prove these formulas by differentiating under the integral sign.

$$\begin{aligned}
\left(\nabla^2_{i,j} Z^{\mathtt{p}}_{s,a}\right)(\theta) &= \int_{\mathcal{S}} \psi\left(s'\right)^{\top} A_i \varphi(s,a) \psi\left(s'\right)^{\top} A_j \varphi(s,a) \frac{\exp\left(\sum_{i=1}^{d} \theta_i \psi\left(s'\right)^{\top} A_i \varphi(s,a)\right)}{\int_{\mathcal{S}} \exp\left(\sum_{i=1}^{d} \theta_i \psi\left(s'\right)^{\top} A_i \varphi(s,a)\right) ds'} ds' \\
&\quad - \int_{\mathcal{S}} \psi\left(s'\right)^{\top} A_i \varphi(s,a) \frac{\exp\left(\sum_{i=1}^{d} \theta_i \psi\left(s'\right)^{\top} A_i \varphi(s,a)\right)}{\int_{\mathcal{S}} \exp\left(\sum_{i=1}^{d} \theta_i \psi\left(s'\right)^{\top} A_i \varphi(s,a)\right) ds'} ds' \left(\nabla_j Z_{s,a}\right)(\theta) \\
&= \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)^{\top} A_i \varphi(s,a) \psi\left(s'\right)^{\top} A_j \varphi(s,a)\right]
\end{aligned}$$

$$
\begin{aligned}
&- \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)^{\top} A_i \varphi(s,a)\right] \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)^{\top} A_j \varphi(s,a)\right] \\
=&\varphi(s,a)^{\top} A_i^{\top}\left(\mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)\psi\left(s'\right)^{\top}\right] - \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)\right]\mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)^{\top}\right]\right) A_j \varphi(s,a) \\
=&\varphi(s,a)^{\top} A_i^{\top} \mathbb{C}^{\theta}_{s,a}\left[\psi\left(s'\right)\right] A_j \varphi(s,a),
\end{aligned}
$$

where we introduce in the last line the $p \times p$ covariance matrix given by

$$
\mathbb{C}^{\theta}_{s,a}\left[\psi\left(s'\right)\right] = \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)\psi\left(s'\right)^{\top}\right] - \mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)\right]\mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)^{\top}\right]
$$

The proof of the form of the Hessian for the reward partition function follows the same steps as above. □

Lemma C.10. (*KL Divergences*) *For any two θ, θ' and for some pair (s,a),*

$$
\exists \tilde{\theta} \in [\theta, \theta']_{\infty}, \quad \mathrm{KL}\left(P^{\mathtt{p}}_{\theta}(\cdot \mid s,a), P^{\mathtt{p}}_{\theta'}(\cdot \mid s,a)\right) = \frac{1}{2}\left(\theta - \theta'\right)^{\top}\left(\nabla^2 Z^{\mathtt{p}}_{s,a}\right)(\tilde{\theta})\left(\theta - \theta'\right),
$$

where $[\theta, \theta']_{\infty}$ denotes the d-dimensional hypercube joining θ to θ'.

Similarly

$$
\exists \tilde{\theta} \in [\theta, \theta']_{\infty}, \quad \mathrm{KL}\left(P^{r}_{\theta}(\cdot \mid s,a), P^{r}_{\theta'}(\cdot \mid s,a)\right) = \frac{1}{2}\left(\theta - \theta'\right)^{\top}\left(\nabla^2 Z^{r}_{s,a}\right)(\tilde{\theta})\left(\theta - \theta'\right).
$$

Proof. We start by writing:

$$
\log\left(\frac{P^{\mathtt{p}}_{\theta}\left(s' \mid s,a\right)}{P^{\mathtt{p}}_{\theta'}\left(s' \mid s,a\right)}\right) = \sum_{i=1}^{d}\left(\theta_i - \theta'_i\right)\psi\left(s'\right)^{\top} A_i \varphi(s,a) - Z^{\mathtt{p}}_{s,a}(\theta) + Z^{\mathtt{p}}_{s,a}\left(\theta'\right),
$$

then

$$
\begin{aligned}
\mathrm{KL}\left(P^{\mathtt{p}}_{\theta}(\cdot \mid s,a), P^{\mathtt{p}}_{\theta'}(\cdot \mid s,a)\right) &= \sum_{i=1}^{d}\left(\theta_i - \theta'_i\right)\mathbb{E}^{\theta}_{s,a}\left[\psi\left(s'\right)\right]^{\top} A_i \varphi(s,a) - Z^{\mathtt{p}}_{s,a}(\theta) + Z^{\mathtt{p}}_{s,a}\left(\theta'\right) \\
&= \frac{1}{2}\left(\theta - \theta'\right)^{\top}\left(\nabla^2 Z^{\mathtt{p}}_{s,a}\right)(\tilde{\theta})\left(\theta - \theta'\right),
\end{aligned}
$$

where in the last line, we used, by a Taylor expansion, that $Z_{s,a}\left(\theta'\right) = Z_{s,a}(\theta) + \left(\nabla Z_{s,a}(\theta)\right)^{\top}\left(\theta' - \theta\right) + \frac{1}{2}(\theta - \theta')^{\top}\left(\nabla^2 Z_{s,a}(\tilde{\theta})\right)\left(\theta - \theta'\right)$ for some $\tilde{\theta} \in [\theta, \theta']_{\infty}$.

The proof of the form of the KL divergence for the reward follows the same steps as above. □

A transportation lemma for rewards

Lemma C.11. *We provide a closed-form formula for the difference of expected rewards under two distinct parameters:*

$$\exists \theta_3 \in [\theta_1, \theta_2], \qquad \mathbb{E}^{\theta_1}_{s,a}[r] = \mathbb{E}^{\theta_2}_{s,a}[r] + \frac{\mathbb{V}\mathrm{ar}^{\theta_3}_{s,a}(r)}{2} B^\top M_{\theta_1-\theta_2}\varphi(s,a)$$

Proof. Let's recall the gradient of the reward log partition function:

$$\left(\nabla_i Z^{\mathtt{r}}_{s,a}\right)(\theta^{\mathtt{r}}) = \mathbb{E}^{\theta^{\mathtt{r}}}_{s,a}[r]\ B^\top A_i \varphi(s,a)$$

then for all $\theta^{\mathtt{r}\prime}$ we have:

$$\mathbb{E}^{\theta^{\mathtt{r}}}_{s,a}[r] = \frac{1}{B^\top M_{\theta^{\mathtt{r}\prime}}\varphi(s,a)} \nabla_i Z^{\mathtt{r}}_{s,a}(\theta^{\mathtt{r}})^\top \theta^{\mathtt{r}\prime}$$

Let $\theta_1, \theta_2 \in \mathbf{R}^d$, using Taylor-Cauchy's formula there exists $\theta_3 \in [\theta_1, \theta_2]$ such that:

$$\mathbb{E}^{\theta_1}_{s,a}[r] = \mathbb{E}^{\theta_2}_{s,a}[r] + \frac{1}{2B^\top M_{\theta^{\mathtt{r}\prime}}\varphi(s,a)} (\theta_1-\theta_2)^\top \nabla^2 Z^{\mathtt{r}}_{s,a}(\theta_3)^\top \theta^{\mathtt{r}\prime}$$

We know that $\left(\nabla^2_{i,j} Z^{\mathtt{r}}_{s,a}\right)(\theta) = \mathbb{V}\mathrm{ar}^{\theta}_{s,a}(r) \times \varphi(s,a)^\top A_i^\top B B^\top A_j \varphi(s,a)$, choosing $\theta^{\mathtt{r}\prime} = \theta_1 - \theta_2$ we find:

$$\mathbb{E}^{\theta_1}_{s,a}[r] = \mathbb{E}^{\theta_2}_{s,a}[r] + \frac{\mathbb{V}\mathrm{ar}^{\theta_3}_{s,a}(r)}{2} B^\top M_{\theta_1-\theta_2}\varphi(s,a).$$

□

C.2.4 Elliptical potentials and elliptical lemma

Elliptical lemma

Here we show a lemma that is popular for regret control in linear MDPs and linear Bandits.

First, consider the notations: $G_{s,a} := (\varphi(s,a)^\top A_i^\top A_j \varphi(s,a))_{1\le i,j\le d}$, $\bar{G}^{\mathtt{e}}_n \equiv \bar{G}^{\mathtt{e}}_{(k-1)H} := G_n + (\alpha^{\mathtt{e}})^{-1}\eta A$, and $G_n \equiv G_{(k-1)H} := \sum_{\tau=1}^{k-1}\sum_{h=1}^{H} G_{s_h^\tau, a_h^\tau}$. Where $\mathtt{e}$ represents either $\mathtt{r}$ or $\mathtt{p}$, we omit the superscript $\mathtt{e}$ w.l.o.g in the rest of this section.

Lemma C.12. (*Elliptical lemma and variant for bounded potentials*) *Let $c \in \mathbf{R}^+$, we can bound the sum of feature norms as follows*

$$\sum_{t=1}^{T} \min\{c, \sum_{h=1}^{H} \left\|\bar{G}_n^{-1/2} G_{s,a} \bar{G}_n^{-1/2}\right\|\} \le \frac{c}{\log(1+c)} d \log\left(1 + \alpha\eta^{-1} B_{\varphi,\mathbb{A}} n\right).$$

where $B_{\varphi,\mathbb{A}} := \sup_{s,a} \|\mathbb{A}^{-1} G_{s,a}\|$.

Further, we have

$$\sum_{t=1}^{T}\sum_{h=1}^{H}\left\|\bar{G}_n^{-1/2}G_{s,a}\bar{G}_n^{-1/2}\right\| \leq 2d\log\left(1+\alpha\eta^{-1}B_{\varphi,\mathbb{A}}n\right)+\frac{3dH}{\log(2)}\log\left(1+\frac{\alpha\|A\|_2^2B_{\varphi,\mathbb{A}}^2}{\eta\log(2)}\right)$$

Proof. First we have

$$\begin{aligned}\|\bar{G}_n^{-1/2}G_{s,a}\bar{G}_n^{-1/2}\| &= \sqrt{{}^\top(\bar{G}_n^{-1/2}G_{s,a}\bar{G}_n^{-1/2}\bar{G}_n^{-1/2}G_{s,a}\bar{G}_n^{-1/2})}\\ &\leq^\top (\bar{G}_n^{-1/2}G_{s,a}\bar{G}_n^{-1/2}) =^\top (\bar{G}_n^{-1}G_{s,a}) =^\top (\boldsymbol{a}_h^\top\bar{G}_n^{-1}\boldsymbol{a}_h)\end{aligned}$$

the last line is because $G_{s,a}=\boldsymbol{a}_h\boldsymbol{a}_h^\top$, where $\boldsymbol{a}_h=(A_i\varphi(s_h,a_h))_{i\in[d]}$.

First result. Consider $h\in[H]$, denote $(\lambda_{h,i})i\in[d]$ the eigenvalues of $\boldsymbol{a}_h^\top\bar{G}_n^{-1}\boldsymbol{a}_h$. $\bar{G}_n$ is positive definite hence $\lambda_{h,i}>0,\forall h,i$, then

$$\begin{aligned}\min\{c,\sum_{h=1}^{H}\mathrm{Tr}(\boldsymbol{a}_h^\top\bar{G}_n^{-1}\boldsymbol{a}_h)\} &= \min\{c,\sum_{h=1}^{H}\sum_{i=1}^{d}\lambda_{h,i}\}\\ &\leq \frac{c}{\log(1+c)}\sum_{h=1}^{H}\sum_{i=1}^{d}\log(1+\lambda_{h,i}) \qquad (\log\text{ is concave})\\ &\leq \frac{c}{\log(1+c)}\sum_{h=1}^{H}\log(\prod_{i=1}^{d}1+\lambda_{h,i})\\ &\leq \frac{c}{\log(1+c)}\sum_{h=1}^{H}\log\det(I+\boldsymbol{a}_h^\top\bar{G}_n^{-1}\boldsymbol{a}_h)\\ &\leq \frac{c}{\log(1+c)}\log\left(\frac{\det(\bar{G}_n+\sum_{h=1}^{H}G_{s_h,a_h})}{\det(\bar{G}_n)}\right)\end{aligned}$$

where the last line follows from the matrix determinant lemma:

$$\det\left(\bar{G}_n+\boldsymbol{a}_h\boldsymbol{a}_h^\top\right)=\det(I+\boldsymbol{a}_h^\top\bar{G}_n^{-1}\boldsymbol{a}_h)\det(\bar{G}_n)$$

Therefore:

$$\sum_{t=1}^{T}\min\{c,\sum_{h=1}^{H}\left\|\bar{G}_n^{-1}G_{s_h^t,a_h^t}\right\|\}\leq\frac{c}{\log(1+c)}\sum_{t=1}^{T}\log\frac{\det\left(\bar{G}_{n+H}\right)}{\det\left(\bar{G}_n\right)},$$

We can now control the R.H.S. of the above equation, as

$$\begin{aligned}\sum_{t=1}^{T}\log\frac{\det\left(\bar{G}_{n+H}\right)}{\det\left(\bar{G}_n\right)} &= \sum_{t=1}^{T}\log\frac{\det\left(\bar{G}_{tH}\right)}{\det\left(\bar{G}_{(t-1)H}\right)}=\log\frac{\det\left(\bar{G}_{TH}\right)}{\det\left(\bar{G}_0\right)}\\ &= \log\frac{\det\left(\bar{G}_N\right)}{\det\left((\alpha^{\mathtt{p}})^{-1}\eta\mathbb{A}\right)}=\log\det\left(I+\alpha\eta^{-1}\,\mathrm{A}^{-1}G_N\right)\end{aligned}$$

$$\leq d \log\left(1 + \frac{\alpha^{\mathtt{p}}\eta^{-1}}{d} \operatorname{tr}\left(\mathbb{A}^{-1} G_n\right)\right) \qquad \text{(Trace-determinant (or AM-GM) inequality)}$$
$$\leq d \log\left(1 + \alpha^{\mathtt{p}}\eta^{-1} B_{\varphi,\mathbb{A}} n\right)$$

This concludes the proof of the first result.

Second result. First, we have $\sup_{s,a} \|G_{s,a}\|_2 \leq \|A\|_2 B_{\varphi,\mathbb{A}}$.

Fix an episode $k \in [K], n = (k-1)H$, using Lemma C.14, we know that the number of times $h \in [h]$ such that $\left\|\bar{G}_n^{-1} G_{s_h,a_h}\right\| \geq 1$ is smaller than $\frac{3d}{\log(2)} \log\left(1 + \frac{\alpha(\|A\|_2 B_{\varphi,\mathbb{A}})^2}{\eta \log(2)}\right)$. Let us call $\mathcal{T}_k := \{h \in [H] \left\|\bar{G}_{(k-1)h}^{-1} G_{s_h,a_h}\right\| \leq 1\}$, then

$$\sum_{t=1}^{T}\sum_{h=1}^{H} \left\|\bar{G}_n^{-1} G_{s_h^t,a_h^t}\right\| \leq \frac{3d}{\log(2)} \log\left(1 + \frac{\alpha\|A\|_2^2 B_{\varphi,\mathbb{A}}^2}{\eta \log(2)}\right) + \sum_{h \in \mathcal{T}_k} \min\{1, \left\|\bar{G}_n^{-1} G_{s_h^t,a_h^t}\right\|\}$$

the sum of the right hand side is similar to the first result. Although the sum is not contiguous, the previous bound holds since if $h_1 < h_2, \det(\bar{G}_{n+h_1}) \leq \det(\bar{G}_{n+h_2})$, this concludes the proof. □

Remark C.13. *We can also write from the lemma in terms of* $\|(A_i\varphi(\tilde{s}_h, \pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^r)^{-1}}$ *by skipping the norm upper bound at the beginning of the proof:*

$$\sum_{t=1}^{T} \min\{c, \sum_{h=1}^{H} \|(A_i\varphi(\tilde{s}_h, \pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^r)^{-1}}\} \leq \frac{c}{\log(1+c)} d \log\left(1 + \alpha\eta^{-1} B_{\varphi,\mathbb{A}} n\right).$$

and

$$\sum_{t=1}^{T}\sum_{h=1}^{H} \|(A_i\varphi(\tilde{s}_h, \pi(\tilde{s}_h)))_{1\leq i\leq d}\|_{(\bar{G}_k^r)^{-1}} \leq 2d \log\left(1 + \alpha\eta^{-1} B_{\varphi,\mathbb{A}} n\right) + \frac{3dH}{\log(2)} \log\left(1 + \frac{\alpha\|A\|_2^2 B_{\varphi,\mathbb{A}}^2}{\eta \log(2)}\right)$$

Elliptical potentials: finite number of large feature norms (contribution)

Lemma C.14. (*Worst case elliptical potentials, adaptation of Exercise 19.3 (Lattimore and Szepesvári, 2020) for matrices*) *Let* $V_0 = \lambda I$ *and* $a_1, \ldots, a_n \in \mathbb{R}^{d\times p}$ *be a sequence of matrices with* $\|a_t\|_2 \leq L$ *for all* $t \in [n]$. *Let* $V_t = V_0 + \sum_{s=1}^{t} a_s a_s^\top$, *then*

$$\left|\{t \in \mathbb{N}^*, \|a_t\|_{V_{t-1}^{-1}} \geq 1\}\right| \leq \frac{3d}{\log(2)} \log\left(1 + \frac{L^2}{\lambda \log(2)}\right)$$

Proof. Let $\mathcal{T}$ be the set of rounds t when $\|a_t\|_{V_{t-1}^{-1}} \geq 1$ and $G_t = V_0 + \sum_{s=1}^{t} \mathbb{I}_{\mathcal{T}}(s) a_s a_s^\top$. Then

$$\begin{aligned}
\left(\frac{d\lambda + |\mathcal{T}|L^2}{d}\right)^d &\geq \left(\frac{\operatorname{trace}(G_n)}{d}\right)^d \\
&\geq \det(G_n) && \text{(Trace-determinant inequality)} \\
&= \det(V_0) \prod_{t \in T} \left(1 + \|a_t\|^2_{G_{t-1}^{-1}}\right) \\
&\geq \det(V_0) \prod_{t \in T} \left(1 + \|a_t\|^2_{V_{t-1}^{-1}}\right) \\
&\geq \lambda^d 2^{|\mathcal{T}|}
\end{aligned}$$

where the third line follows from the matrix determinant lemma:

$$\det\left(\bar{G}_n + \boldsymbol{a}_h \boldsymbol{a}_h^\top\right) = \det(I + \boldsymbol{a}_h^\top \bar{G}_n^{-1} \boldsymbol{a}_h) \det(\bar{G}_n).$$

Rearranging and taking the logarithm shows that

$$|\mathcal{T}| \leq \frac{d}{\log(2)} \log\left(1 + \frac{|\mathcal{T}|L^2}{d\lambda}\right)$$

Abbreviate $x = d/\log(2)$ and $y = L^2/d\lambda$, which are both positive. Then

$$x \log(1 + y(3x \log(1 + xy))) \leq x \log\left(1 + 3x^2 y^2\right) \leq x \log(1 + xy)^3 = 3x \log(1 + xy).$$

Since $z - x \log(1 + yz)$ is decreasing for $z \geq 3x \log(1 + xy)$ it follows that

$$|\mathcal{T}| \leq 3x \log(1 + xy) = \frac{3d}{\log(2)} \log\left(1 + \frac{L^2}{\lambda \log(2)}\right).$$

□

Appendix D

Complements on Chapter 6

Contents

D.1 Experiment Details

In all experiments we choose to use the same hyper-parameter values for all tasks as the best-performing ones reported in the literature or in their respective open source implementation documentation. In Tables D.1, D.2 and D.3, we report the list of hyper-parameters for all continuous control tasks.

Table D.1 – Hyperparameters used both in SAC and AVEC-SAC.

Parameter	Value
Adam stepsize	$3 \cdot 10^{-4}$
Discount (γ)	0.99
Replay buffer size	10^6
Batch size	256
Nb. hidden layers	2
Nb. hidden units per layer	256
Nonlinearity	ReLU
Target smoothing coefficient (τ)	0.01
Target update interval	1
Gradient steps	1

Table D.2 – Hyperparameters used both in PPO and AVEC-PPO.

Parameter	Value
Horizon (T)	2048
Adam stepsize	$2.5 \cdot 10^{-4}$
Nb. epochs	10
Nb. minibatches	32
Nb. hidden layers	2
Nb. hidden units per layer	64
Nonlinearity	tanh
Discount (γ)	0.99
GAE parameter (λ)	0.95
Clipping parameter (ε)	0.2

Overall, we choose the hyper-parameters in a way to ensure the best performance for the conventional actor-critic framework. In other words, since we are interested in evaluating the impact of this new critic, everything else is kept as is. This experimental protocol may not benefit AVEC.

Table D.3 – Hyperparameters used both in TRPO and AVEC-TRPO.

Parameter	Value
Horizon (T)	2048
Adam stepsize	$1 \cdot 10^{-4}$
Nb. hidden layers	2
Nb. hidden units per layer	64
Nonlinearity	tanh
Discount (γ)	0.99
GAE parameter (λ)	0.95
Stepsize KL	0.01
Nb. iterations for the conjugate gradient	15

D.2 Details about the environments

Table D.4 – Environments details.

Environment	Description
Ant-v2	Make a four-legged creature walk forward as fast as possible.
AntBulletEnv-v0	Idem. Ant is heavier, encouraging it to typically have two or more legs on the ground (source: PyBullet Guide - url).
HalfCheetah-v2	Make a 2D cheetah robot run.
HalfCheetahBulletEnv-v0	Idem.
Humanoid-v2	Make a three-dimensional bipedal robot walk forward as fast as possible, without falling over.
Reacher-v2	Make a 2D robot reach to a randomly located target.
Walker2d-v2	Make a 2D robot walk forward as fast as possible.
Acrobot-v1	Swing the end of a two-joint acrobot up to a given height.
MountainCar-v0	Get an under powered car to the top of a hill.

D.3 Dimensions of Studied Tasks

Table D.5 – Actions and observations dimensions.

Task	$\mathcal{S}$	$\mathcal{A}$
Ant	$\mathbb{R}^{111}$	$\mathbb{R}^{8}$
AntBullet	$\mathbb{R}^{28}$	$\mathbb{R}^{8}$
HalfCheetah	$\mathbb{R}^{17}$	$\mathbb{R}^{6}$
HalfCheetahBullet	$\mathbb{R}^{26}$	$\mathbb{R}^{6}$
Humanoid	$\mathbb{R}^{376}$	$\mathbb{R}^{17}$
Reacher	$\mathbb{R}^{11}$	$\mathbb{R}^{2}$
Walker2d	$\mathbb{R}^{17}$	$\mathbb{R}^{6}$
Acrobot	$\mathbb{R}^{6}$	3
MountainCar	$\mathbb{R}^{2}$	3

List of Figures

List of Algorithms

List of Tables

List of References

[1] Yasin Abbasi-Yadkori, David Pal, and Csaba Szepesvari. Online-to-confidence-set conversions and application to sparse stochastic bandits. In *Artificial Intelligence and Statistics*. PMLR. 2012, pp. 1–9.

[2] Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems*. 2011, pp. 2312–2320.

[3] Yasin Abbasi-Yadkori and Csaba Szepesvári. Regret bounds for the adaptive control of linear quadratic systems. In *Proceedings of the 24th Annual Conference on Learning Theory*. JMLR Workshop and Conference Proceedings. 2011, pp. 1–26.

[4] Naoki Abe and Philip M Long. Associative reinforcement learning using linear probabilistic concepts. In *ICML*. Citeseer. 1999, pp. 3–11.

[5] Marc Abeille and Alessandro Lazaric. Linear thompson sampling revisited. In *Artificial Intelligence and Statistics*. PMLR. 2017, pp. 176–184.

[6] Jacob D Abernethy, Kareem Amin, and Ruihao Zhu. Threshold bandits, with and without censored feedback. *Advances In Neural Information Processing Systems* 29 (2016).

[7] Alekh Agarwal, Dean P Foster, Daniel J Hsu, Sham M Kakade, and Alexander Rakhlin. Stochastic convex optimization with bandit feedback. In *Advances in Neural Information Processing Systems 24*. Ed. by J. Shawe-Taylor, R. S. Zemel, P. L. Bartlett, F. Pereira, and K. Q. Weinberger. Curran Associates, Inc., 2011, pp. 1035–1043.

[8] Alekh Agarwal, Daniel Hsu, Satyen Kale, John Langford, Lihong Li, and Robert Schapire. Taming the monster: A fast and simple algorithm for contextual bandits. In *International Conference on Machine Learning*. PMLR. 2014, pp. 1638–1646.

[9] Rajeev Agrawal. Sample mean based index policies by o (log n) regret for the multi-armed bandit problem. *Advances in Applied Probability* 27.4 (1995), pp. 1054–1078.

[10] Shipra Agrawal and Nikhil R. Devanur. Bandits with Concave Rewards and Convex Knapsacks. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*. EC '14. Palo Alto, California, USA: ACM, 2014, pp. 989–1006.

[11] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on learning theory*. JMLR Workshop and Conference Proceedings. 2012, pp. 39–1.

[12] Shipra Agrawal and Randy Jia. Optimistic posterior sampling for reinforcement learning: worst-case regret bounds. *Advances in Neural Information Processing Systems* 30 (2017).

[13] Aymen Al Marjani and Alexandre Proutiere. Adaptive sampling for best policy identification in markov decision processes. In *International Conference on Machine Learning*. PMLR. 2021, pp. 7459–7468.

[14] Christopher Amato, Girish Chowdhary, Alborz Geramifard, N Kemal Üre, and Mykel J Kochenderfer. Decentralized control of partially observable Markov decision processes. In *52nd IEEE Conference on Decision and Control*. IEEE. 2013, pp. 2398–2405.

[15] Venkatachalam Anantharam, Pravin Varaiya, and Jean Walrand. Asymptotically efficient allocation rules for the multiarmed bandit problem with multiple plays-Part I: IID rewards. *IEEE Transactions on Automatic Control* 32.11 (1987), pp. 968–976.

[16] Kenneth J Arrow, Theodore Harris, and Jacob Marschak. Optimal inventory policy. *Econometrica: Journal of the Econometric Society* (1951), pp. 250–272.

[17] Kenneth Joseph Arrow, Samuel Karlin, Herbert E Scarf, et al. Studies in the mathematical theory of inventory and production (1958).

[18] Jean-Yves Audibert, Sébastien Bubeck, and Rémi Munos. Best arm identification in multi-armed bandits. In *COLT*. Citeseer. 2010, pp. 41–53.

[19] Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning* 47.2 (2002), pp. 235–256.

[20] Orly Avner and Shie Mannor. Multi-user lax communications: a multi-armed bandit approach. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*. IEEE. 2016, pp. 1–9.

[21] Alex Ayoub, Zeyu Jia, Csaba Szepesvari, Mengdi Wang, and Lin Yang. Model-based reinforcement learning with value-targeted regression. In *International Conference on Machine Learning*. PMLR. 2020, pp. 463–474.

[22] Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*. PMLR. 2017, pp. 263–272.

[23] Maryam Aziz, Emilie Kaufmann, and Marie-Karelle Riviere. On multi-armed bandit designs for dose-finding clinical trials. *Journal of Machine Learning Research* 22.1-38 (2021), p. 4.

[24] Katy S Azoury and Manfred K Warmuth. Relative loss bounds for on-line density estimation with the exponential family of distributions. *Machine Learning* 43.3 (2001), pp. 211–246.

[25] Elias Bareinboim, Andrew Forney, and Judea Pearl. Bandits with unobserved confounders: A causal approach. *Advances in Neural Information Processing Systems* 28 (2015).

[26] Peter L Bartlett, Wouter M Koolen, Alan Malek, Eiji Takimoto, and Manfred K Warmuth. Minimax fixed-design linear regression. In *Conference on Learning Theory*. 2015, pp. 226–239.

[27] Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. In *Advances in Neural Information Processing Systems*. 2016, pp. 1471–1479.

[28] Richard Bellman. A problem in the sequential design of experiments. *Sankhya: The Indian Journal of Statistics (1933-1960)* 16.3/4 (1956), pp. 221–229.

[29] Richard Bellman. Dynamic programming. *Science* 153.3731 (1966), pp. 34–37.

[30] Donald A Berry. Modified two-armed bandit strategies for certain clinical trials. *Journal of the American Statistical Association* 73.362 (1978), pp. 339–345.

[31] Quentin Berthet and Vianney Perchet. Fast Rates for Bandit Optimization with Upper-Confidence Frank-Wolfe. In *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, et al. 2017, pp. 2225–2234.

[32] Dimitri P. Bertsekas. *Dynamic Programming and Optimal Control*. 2nd. Athena Scientific, 2000.

[33] Alina Beygelzimer, John Langford, Lihong Li, Lev Reyzin, and Robert Schapire. Contextual bandit algorithms with supervised learning guarantees. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings. 2011, pp. 19–26.

[34] David M Blei, Alp Kucukelbir, and Jon D McAuliffe. Variational inference: A review for statisticians. *Journal of the American statistical Association* 112.518 (2017), pp. 859–877.

[35] Stéphane Boucheron, Gábor Lugosi, and Pascal Massart. *Concentration inequalities: A nonasymptotic theory of independence*. Oxford university press, 2013.

[36] Russell N Bradt, SM Johnson, and Samuel Karlin. On sequential designs for maximizing the sum of n observations. *The Annals of Mathematical Statistics* 27.4 (1956), pp. 1060–1074.

[37] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, et al. OpenAI Gym. *arXiv preprint arXiv:1606.01540* (2016).

[38] George W. Brown. On Small-Sample Estimation. *Annals of Mathematical Statistics* 18.4 (Dec. 1947), pp. 582–585.

[39] Sébastien Bubeck, Nicolò Cesa-Bianchi, et al. Regret Analysis of Stochastic and Nonstochastic Multi-armed Bandit Problems. *Foundations and Trends® in Machine Learning* 5.1 (2012), pp. 1–122.

[40] Sébastien Bubeck, Rémi Munos, and Gilles Stoltz. Pure exploration in multi-armed bandits problems. In *International conference on Algorithmic learning theory*. Springer. 2009, pp. 23–37.

[41] Kechao Cai, Xutong Liu, Yu-Zhen Janice Chen, and John CS Lui. An online learning approach to network application optimization with guarantee. In *IEEE INFOCOM 2018-IEEE Conference on Computer Communications*. IEEE. 2018, pp. 2006–2014.

[42] Felipe Caro and Jérémie Gallien. Dynamic assortment with demand learning for seasonal consumer goods. *Management science* 53.2 (2007), pp. 276–292.

[43] Miguel A Carreira-Perpinan and Geoffrey Hinton. On contrastive divergence learning. In *International workshop on artificial intelligence and statistics*. PMLR. 2005, pp. 33–40.

[44] Leonardo Cella, Alessandro Lazaric, and Massimiliano Pontil. Meta-learning with stochastic linear bandits. In *International Conference on Machine Learning*. PMLR. 2020, pp. 1360–1370.

[45] Nicolo Cesa-Bianchi, Yoav Freund, David Haussler, David P Helmbold, Robert E Schapire, and Manfred K Warmuth. How to use expert advice. *Journal of the ACM (JACM)* 44.3 (1997), pp. 427–485.

[46] Nicolo Cesa-Bianchi, Philip M Long, and Manfred K Warmuth. Worst-case quadratic loss bounds for prediction using linear functions and gradient descent. *IEEE Transactions on Neural Networks* 7.3 (1996), pp. 604–619.

[47] Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.

[48] Krzysztof Chalupka, Frederick Eberhardt, and Pietro Perona. Causal feature learning: an overview. *Behaviormetrika* 44.1 (2017), pp. 137–164.

[49] Olivier Chapelle and Lihong Li. An empirical evaluation of thompson sampling. *Advances in neural information processing systems* 24 (2011).

[50] Wei Chen, Yajun Wang, and Yang Yuan. Combinatorial multi-armed bandit: General framework and applications. In *International conference on machine learning*. PMLR. 2013, pp. 151–159.

[51] Xi Chen, Qihang Lin, and Dengyong Zhou. Statistical decision making for optimal budget allocation in crowd labeling. *The Journal of Machine Learning Research* 16.1 (2015), pp. 1–46.

[52] James Cheshire, Pierre Menard, and Alexandra Carpentier. The Influence of Shape Constraints on the Thresholding Bandit Problem. In *Conference on Learning Theory*. PMLR. 2020, pp. 1228–1275.

[53] Sayak Ray Chowdhury and Aditya Gopalan. Online learning in kernelized markov decision processes. In *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR. 2019, pp. 3197–3205.

[54] Sayak Ray Chowdhury, Aditya Gopalan, and Odalric-Ambrym Maillard. Reinforcement Learning in Parametric MDPs with Exponential Families. In *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 1855–1863.

[55] Richard Combes, Stefan Magureanu, and Alexandre Proutiere. Minimal exploration in structured stochastic bandits. In vol. 30. 2017.

[56] Corinna Cortes and Mehryar Mohri. On transductive regression. In *Advances in Neural Information Processing Systems*. 2007, pp. 305–312.

[57] Erwin Coumans and Yunfei Bai. *PyBullet, a Python module for physics simulation for games, robotics and machine learning*. 2016.

[58] Bo Dai, Hanjun Dai, Arthur Gretton, Le Song, Dale Schuurmans, and Niao He. Kernel exponential family estimation via doubly dual embedding. In *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR. 2019, pp. 2321–2330.

[59] Bo Dai, Albert Shaw, Lihong Li, Lin Xiao, Niao He, Zhen Liu, et al. Sbeed: Convergent reinforcement learning with nonlinear function approximation. In *International Conference on Machine Learning*. PMLR. 2018, pp. 1125–1134.

[60] Varsha Dani, Thomas P Hayes, and Sham M Kakade. Stochastic linear optimization under bandit feedback (2008).

[61] Christoph Dann, Tor Lattimore, and Emma Brunskill. Unifying PAC and regret: Uniform PAC bounds for episodic reinforcement learning. *Advances in Neural Information Processing Systems* 30 (2017).

[62] Christoph Dann, Yishay Mansour, Mehryar Mohri, Ayush Sekhari, and Karthik Sridharan. Reinforcement learning with feedback graphs. *Advances in Neural Information Processing Systems* 33 (2020), pp. 16868–16878.

[63] Rémy Degenne and Wouter M Koolen. Pure exploration with multiple correct answers. *arXiv preprint arXiv:1902.03475* (2019).

[64] Rémy Degenne, Wouter M Koolen, and Pierre Ménard. Non-asymptotic pure exploration by solving games. *arXiv preprint arXiv:1906.10431* (2019).

[65] Rémy Degenne, Han Shao, and Wouter Koolen. Structure adaptive algorithms for stochastic bandits. In *International Conference on Machine Learning*. PMLR. 2020, pp. 2443–2452.

[66] Nikhil R Devanur, Kamal Jain, Balasubramanian Sivan, and Christopher A Wilkens. Near optimal online algorithms and fast approximation algorithms for resource allocation problems. *Journal of the ACM (JACM)* 66.1 (2019), p. 7.

[67] Harold French Dodge and Harry G Romig. A method of sampling inspection. *The Bell System Technical Journal* 8.4 (1929), pp. 613–631.

[68] Carlos Domingo, Ricard Gavalda, and Osamu Watanabe. Adaptive sampling methods for scaling up knowledge discovery algorithms. *Data Mining and Knowledge Discovery* 6.2 (2002), pp. 131–152.

[69] Omar Darwiche Domingues, Pierre Ménard, Emilie Kaufmann, and Michal Valko. Episodic reinforcement learning in finite mdps: Minimax lower bounds revisited. In *Algorithmic Learning Theory*. PMLR. 2021, pp. 578–598.

[70] Kefan Dong, Jian Peng, Yining Wang, and Yuan Zhou. Root-n-regret for learning in markov decision processes with function approximation and low bellman rank. In *Conference on Learning Theory*. PMLR. 2020, pp. 1554–1557.

[71] Simon Du, Sham Kakade, Jason Lee, Shachar Lovett, Gaurav Mahajan, Wen Sun, et al. Bilinear classes: A structural framework for provable generalization in rl. In *International Conference on Machine Learning*. PMLR. 2021, pp. 2826–2836.

[72] Simon Du, Akshay Krishnamurthy, Nan Jiang, Alekh Agarwal, Miroslav Dudik, and John Langford. Provably efficient RL with rich observations via latent state decoding. In *International Conference on Machine Learning*. PMLR. 2019, pp. 1665–1674.

[73] Simon S Du, Sham M Kakade, Ruosong Wang, and Lin F Yang. Is a good representation sufficient for sample efficient reinforcement learning? *arXiv preprint arXiv:1910.03016* (2019).

[74] Audrey Durand, Odalric-Ambrym Maillard, and Joelle Pineau. Streaming kernel regression with provably adaptive mean, variance, and regularization. *The Journal of Machine Learning Research* 19.1 (2018), pp. 650–683.

[75] David Duvenaud, James Lloyd, Roger Grosse, Joshua Tenenbaum, and Ghahramani Zoubin. Structure discovery in nonparametric regression through compositional kernel search. In *International Conference on Machine Learning*. PMLR. 2013, pp. 1166–1174.

[76] Aryeh Dvoretzky, Jack Kiefer, and Jacob Wolfowitz. The inventory problem: I. Case of known distributions of demand. *Econometrica: Journal of the Econometric Society* (1952), pp. 187–222.

[77] Eyal Even-Dar, Shie Mannor, Yishay Mansour, and Sridhar Mahadevan. Action Elimination and Stopping Conditions for the Multi-Armed Bandit and Reinforcement Learning Problems. *Journal of machine learning research* 7.6 (2006).

[78] Claude-Nicolas Fiechter. Efficient reinforcement learning. In *Proceedings of the seventh annual conference on Computational learning theory*. 1994, pp. 88–97.

[79] Sarah Filippi, Olivier Cappe, Aurélien Garivier, and Csaba Szepesvári. Parametric bandits: The generalized linear case. *Advances in Neural Information Processing Systems* 23 (2010).

[80] Yannis Flet-Berliac, Johan Ferret, Olivier Pietquin, Philippe Preux, and Matthieu Geist. Adversarially Guided Actor-Critic. In *International Conference on Learning Representations*. 2021.

[81] Yannis Flet-Berliac, Reda Ouhamma, Odalric-Ambrym Maillard, and Philippe Preux. Learning value functions in deep policy gradients using residual variance. *International Conference on Learning Representations* (2021).

[82] Yannis Flet-Berliac and Philippe Preux. MERL: Multi-Head Reinforcement Learning. In *Deep Reinforcement Learning Workshop, NeurIPS*. 2019.

[83] Yannis Flet-Berliac and Philippe Preux. Only Relevant Information Matters: Filtering Out Noisy Samples To Boost RL. In *International Joint Conference on Artificial Intelligence*. 2020, pp. 2711–2717.

[84] Xavier Fontaine, Quentin Berthet, and Vianney Perchet. Regularized contextual bandits. In *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR. 2019, pp. 2144–2153.

[85] Xavier Fontaine, Shie Mannor, and Vianney Perchet. An adaptive stochastic optimization algorithm for resource allocation. In *Algorithmic Learning Theory*. PMLR. 2020, pp. 319–363.

[86] Dean P Foster. Prediction in the worst case. *The Annals of Statistics* (1991), pp. 1084–1090.

[87] Dylan J Foster, Sham M Kakade, Jian Qian, and Alexander Rakhlin. The Statistical Complexity of Interactive Decision Making. *arXiv preprint arXiv:2112.13487* (2021).

[88] Marguerite Frank and Philip Wolfe. An algorithm for quadratic programming. *Naval research logistics quarterly* 3.1-2 (1956), pp. 95–110.

[89] David A Freedman. On tail probabilities for martingales. *the Annals of Probability* (1975), pp. 100–118.

[90] Yi Gai and Bhaskar Krishnamachari. Decentralized online learning algorithms for opportunistic spectrum access. In *2011 IEEE Global Telecommunications Conference-GLOBECOM 2011*. IEEE. 2011, pp. 1–6.

[91] Yi Gai, Bhaskar Krishnamachari, and Rahul Jain. Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation. In *2010 IEEE Symposium on New Frontiers in Dynamic Spectrum (DySPAN)*. IEEE. 2010, pp. 1–9.

[92] Pierre Gaillard, Sébastien Gerchinovitz, Malo Huard, and Gilles Stoltz. Uniform regret bounds over $\mathbb{R}^d$ for the sequential linear regression problem with the square loss. In *Algorithmic Learning Theory*. 2019, pp. 404–432.

[93] Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In *Conference on Learning Theory*. PMLR. 2016, pp. 998–1027.

[94] Aurélien Garivier, Pierre Ménard, Laurent Rossi, and Pierre Menard. Thresholding bandit for dose-ranging: The impact of monotonicity. *arXiv preprint arXiv:1711.04454* (2017).

[95] John C Gittins and David M Jones. A dynamic allocation index for the discounted multiarmed bandit problem. *Biometrika* 66.3 (1979), pp. 561–565.

[96] Aditya Gopalan and Shie Mannor. Thompson sampling for learning parameterized markov decision processes. In *Conference on Learning Theory*. PMLR. 2015, pp. 861–898.

[97] Ole-Christoffer Granmo. Solving two-armed Bernoulli bandit problems using a Bayesian learning automaton. *International Journal of Intelligent Computing and Cybernetics* (2010).

[98] Evan Greensmith, Peter L Bartlett, and Jonathan Baxter. Variance reduction techniques for gradient estimates in reinforcement learning. *Journal of Machine Learning Research* 5 (2004), pp. 1471–1530.

[99] Shixiang Gu, Timothy Lillicrap, Ilya Sutskever, and Sergey Levine. Continuous deep q-learning with model-based acceleration. In *International Conference on Machine Learning*. 2016, pp. 2829–2838.

[100] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft Actor-Critic: Off-Policy Maximum Entropy Deep Reinforcement Learning with a Stochastic Actor. In *International Conference on Machine Learning*. 2018, pp. 1856–1865.

[101] Mance E Harmon and Leemon C Baird III. Multi-player residual advantage learning with general function approximation ().

[102] Junya Honda and Akimichi Takemura. Non-asymptotic analysis of a new bandit algorithm for semi-bounded rewards. *J. Mach. Learn. Res.* 16 (2015), pp. 3721–3756.

[103] Abdolhossein Hoorfar and Mehdi Hassani. Inequalities on the Lambert W function and hyperpower function. *J. Inequal. Pure and Appl. Math* 9.2 (2008), pp. 5–9.

[104] Timothy Hospedales, Antreas Antoniou, Paul Micaelli, and Amos Storkey. Meta-learning in neural networks: A survey. *arXiv preprint arXiv:2004.05439* (2020).

[105] Kaixuan Huang, Sham M Kakade, Jason D Lee, and Qi Lei. A short note on the relationship of information gain and eluder dimension. *arXiv preprint arXiv:2107.02377* (2021).

[106] Andrew Ilyas, Logan Engstrom, Shibani Santurkar, Dimitris Tsipras, Firdaus Janoos, Larry Rudolph, et al. A Closer Look at Deep Policy Gradients. In *International Conference on Learning Representations*. 2020.

[107] JR Isbell. On a problem of Robbins. *The Annals of Mathematical Statistics* 30.2 (1959), pp. 606–610.

[108] Haque Ishfaq, Qiwen Cui, Viet Nguyen, Alex Ayoub, Zhuoran Yang, Zhaoran Wang, et al. Randomized Exploration in Reinforcement Learning with General Value Function Approximation. In *International Conference on Machine Learning*. PMLR. 2021, pp. 4607–4616.

[109] Max Jaderberg, Volodymyr Mnih, Wojciech Marian Czarnecki, Tom Schaul, Joel Z Leibo, David Silver, et al. Reinforcement learning with unsupervised auxiliary tasks. *arXiv preprint arXiv:1611.05397* (2016).

[110] Lalit Jain and Kevin G Jamieson. A New Perspective on Pool-Based Active Classification and False-Discovery Control. In *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc., 2019.

[111] Kevin Jamieson and Lalit Jain. A bandit approach to multiple testing with false discovery control. *arXiv preprint arXiv:1809.02235* (2018).

[112] Svante Janson. Tail bounds for sums of geometric and exponential variables. *Statistics & Probability Letters* 135 (2018), pp. 1–6.

[113] Robert G Jeroslow. The polynomial hierarchy and a simple model for competitive analysis. *Mathematical programming* 32.2 (1985), pp. 146–164.

[114] Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*. PMLR. 2017, pp. 1704–1713.

[115] Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*. PMLR. 2020, pp. 2137–2143.

[116] Rie Johnson and Tong Zhang. Accelerating stochastic gradient descent using predictive variance reduction. In *Advances in Neural Information Processing Systems*. 2013, pp. 315–323.

[117] Bent Jørgensen. Maximum likelihood estimation and large-sample inference for generalized linear and nonlinear regression models. *Biometrika* 70.1 (1983), pp. 19–28.

[118] Kwang-Sung Jun, Aniruddha Bhargava, Robert Nowak, and Rebecca Willett. Scalable generalized linear bandits: Online computation and hashing. *arXiv preprint arXiv:1706.00136* (2017).

[119] Leslie Pack Kaelbling. Learning to achieve goals. In *IJCAI*. Citeseer. 1993, pp. 1094–1099.

[120] Leslie Pack Kaelbling. Associative reinforcement learning: A generate and test algorithm. *Machine Learning* 15.3 (1994), pp. 299–319.

[121] Sham Kakade and John Langford. Approximately Optimal Approximate Reinforcement Learning. In *International Conference on Machine Learning*. 2002, pp. 267–274.

[122] Gautam Kamath. Bounds on the expectation of the maximum of samples from a gaussian. *URL http://www. gautamkamath. com/writings/gaussian max. pdf* (2015).

[123] Hideaki Kano, Junya Honda, Kentaro Sakamaki, Kentaro Matsuura, Atsuyoshi Nakamura, and Masashi Sugiyama. Good arm identification via bandit feedback. *Machine Learning* 108.5 (2019), pp. 721–745.

[124] Bilal Kartal, Pablo Hernandez-Leal, and Matthew E Taylor. Terminal prediction as an auxiliary task for deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*. Vol. 15. 1. 2019, pp. 38–44.

[125] Sumeet Katariya, Ardhendu Tripathy, and Robert Nowak. MaxGap Bandit: Adaptive Algorithms for Approximate Ranking. In *Advances in Neural Information Processing Systems*. Ed. by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett. Vol. 32. Curran Associates, Inc., 2019.

[126] Julian Katz-Samuels and Clay Scott. Feasible arm identification. In *International Conference on Machine Learning*. PMLR. 2018, pp. 2535–2543.

[127] Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. *The Journal of Machine Learning Research* 17.1 (2016), pp. 1–42.

[128] Emilie Kaufmann, Wouter Koolen, and Aurélien Garivier. Sequential test for the lowest mean: From Thompson to Murphy sampling. *arXiv preprint arXiv:1806.00973* (2018).

[129] Michael Kearns and Daphne Koller. Efficient reinforcement learning in factored MDPs. In *IJCAI*. Vol. 16. 1999, pp. 740–747.

[130] Michael Kearns, Yishay Mansour, and Satinder Singh. Fast planning in stochastic games. *arXiv preprint arXiv:1301.3867* (2013).

[131] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* (2013).

[132] Jyrki Kivinen and Manfred K Warmuth. Exponentiated gradient versus gradient descent for linear predictors. *information and computation* 132.1 (1997), pp. 1–63.

[133] Bernard O Koopman. The optimum distribution of effort. *Journal of the Operations Research Society of America* 1.2 (1953), pp. 52–63.

[134] Nathaniel Korda, Emilie Kaufmann, and Remi Munos. Thompson sampling for 1-dimensional exponential family bandits. *Advances in neural information processing systems* 26 (2013).

[135] Branislav Kveton and Milos Hauskrecht. Solving Factored MDPs with Exponential-Family Transition Models. In *ICAPS*. 2006, pp. 114–120.

[136] Tze Leung Lai, Herbert Robbins, et al. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics* 6.1 (1985), pp. 4–22.

[137] Tor Lattimore, Koby Crammer, and Csaba Szepesvári. Linear multi-resource allocation with semi-bandit feedback. *Advances in Neural Information Processing Systems* 28 (2015).

[138] Tor Lattimore, Csaba Szepesvari, and Gellert Weisz. Learning with good feature representations in bandits and in rl with a generative model. In *International Conference on Machine Learning*. PMLR. 2020, pp. 5662–5670.

[139] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

[140] Elizaveta Levina and Peter Bickel. Maximum likelihood estimation of intrinsic dimension. *Advances in neural information processing systems* 17 (2004).

[141] Bowen Li, Panlong Yang, Jinlong Wang, Qihui Wu, Shaojie Tang, Xiang-Yang Li, et al. Almost optimal dynamically-ordered channel sensing and accessing for cognitive networks. *IEEE Transactions on Mobile Computing* 13.10 (2013), pp. 2215–2228.

[142] Gene Li, Junbo Li, Nathan Srebro, Zhaoran Wang, and Zhuoran Yang. Exponential Family Model-Based Reinforcement Learning via Score Matching. *arXiv preprint arXiv:2112.14195* (2021).

[143] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, et al. Continuous control with deep reinforcement learning. In *International Conference on Learning Representations*. 2016.

[144] Kaixiang Lin and Jiayu Zhou. Ranking Policy Gradient. In *International Conference on Learning Representations*. 2020.

[145] Nicholas Littlestone, Philip M Long, and Manfred K Warmuth. On-line learning of linear functions. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing*. 1991, pp. 465–475.

[146] Hao Liu, Yihao Feng, Yi Mao, Dengyong Zhou, Jian Peng, and Qiang Liu. Action-dependent control variates for policy optimization via stein identity. In *International Conference on Learning Representations*. 2018.

[147] Andrea Locatelli, Maurilio Gutzeit, and Alexandra Carpentier. An optimal algorithm for the thresholding bandit problem. In *International Conference on Machine Learning*. PMLR. 2016, pp. 1690–1698.

[148] Yangyi Lu, Amirhossein Meisami, and Ambuj Tewari. Low-rank generalized linear bandit problems. In *International Conference on Artificial Intelligence and Statistics*. PMLR. 2021, pp. 460–468.

[149] Yangyi Lu, Amirhossein Meisami, and Ambuj Tewari. Efficient reinforcement learning with prior causal knowledge. In *Conference on Causal Learning and Reasoning*. PMLR. 2022, pp. 526–541.

[150] Odalric-Ambrym Maillard. Self-normalization techniques for streaming confident regression (2016).

[151] Alan Malek and Peter L Bartlett. Horizon-independent minimax linear regression. In *Advances in Neural Information Processing Systems*. 2018, pp. 5259–5268.

[152] Gustavo Malkomes, Charles Schaff, and Roman Garnett. Bayesian optimization for automated model selection. *Advances in Neural Information Processing Systems* 29 (2016).

[153] Shie Mannor, Vianney Perchet, and Gilles Stoltz. Approachability in unknown games: Online learning meets multi-objective optimization. In *Conference on Learning Theory*. PMLR. 2014, pp. 339–355.

[154] Shie Mannor and Ohad Shamir. From bandits to experts: On the value of side-observations. *Advances in Neural Information Processing Systems* 24 (2011).

[155] Pierre Massé. *Les réserves et la régulation de l'avenir dans la vie économique. Vol. 1, Avenir détermine*. Hermann, 1946.

[156] Benedict C May and David S Leslie. Simulation studies in optimistic Bayesian sampling in contextual-bandit problems. *Statistics Group, Department of Mathematics, University of Bristol* 11.02 (2011).

[157] Francisco S Melo, Sean P Meyn, and M Isabel Ribeiro. An analysis of reinforcement learning with function approximation. In *Proceedings of the 25th international conference on Machine learning*. 2008, pp. 664–671.

[158] Pierre Ménard, Omar Darwiche Domingues, Anders Jonsson, Emilie Kaufmann, Edouard Leurent, and Michal Valko. Fast active learning for pure exploration in reinforcement learning. In *International Conference on Machine Learning*. PMLR. 2021, pp. 7599–7608.

[159] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, et al. Asynchronous methods for deep reinforcement learning. In *International Conference on Machine Learning*. 2016, pp. 1928–1937.

[160] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, et al. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* (2013).

[161] Jaouad Mourtada. Exact minimax risk for linear least squares, and the lower tail of sample covariance matrices. *arXiv preprint arXiv:1912.10754* (2019).

[162] Subhojyoti Mukherjee, Kolar Purushothama Naveen, Nandan Sudarsanam, and Balaraman Ravindran. Thresholding bandits with augmented ucb. *arXiv preprint arXiv:1704.02281* (2017).

[163] Ofir Nachum and Mengjiao Yang. Provable representation learning for imitation with contrastive fourier features. *Advances in Neural Information Processing Systems* 34 (2021).

[164] H. Namkoong and J. C. Duchi. Variance-based Regularization with Convex Objectives. In *Advances in Neural Information Processing Systems*. 2017, pp. 2971–2980.

[165] Radford M Neal. Annealed importance sampling. *Statistics and computing* 11.2 (2001), pp. 125–139.

[166] John Ashworth Nelder and Robert WM Wedderburn. Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)* 135.3 (1972), pp. 370–384.

[167] Jungseul Ok, Alexandre Proutiere, and Damianos Tranos. Exploration in structured reinforcement learning. *Advances in Neural Information Processing Systems* 31 (2018).

[168] Francesco Orabona, Nicolo Cesa-Bianchi, and Claudio Gentile. Beyond logarithmic bounds in online learning. In *Artificial intelligence and statistics*. PMLR. 2012, pp. 823–831.

[169] Ian Osband, Daniel Russo, and Benjamin Van Roy. (More) efficient reinforcement learning via posterior sampling. *Advances in Neural Information Processing Systems* 26 (2013).

[170] Ian Osband and Benjamin Van Roy. Model-based reinforcement learning and the eluder dimension. *Advances in Neural Information Processing Systems* 27 (2014).

[171] Ian Osband, Benjamin Van Roy, and Zheng Wen. Generalization and exploration via randomized value functions. In *International Conference on Machine Learning*. PMLR. 2016, pp. 2377–2386.

[172] Reda Ouhamma, Debabrota Basu, and Odalric-Ambrym Maillard. Bilinear exponential family of mdps: Frequentist regret bound with tractable exploration and planning. *Association for the Advancement of Artificial Intelligence* (2023).

[173] Reda Ouhamma, Rémy Degenne, Pierre Gaillard, and Vianney Perchet. Online Sign Identification: Minimization of the Number of Errors in Thresholding Bandits. *Advances in Neural Information Processing Systems* 34 (2021), pp. 18577–18589.

[174] Reda Ouhamma, Odalric-Ambrym Maillard, and Vianney Perchet. Stochastic Online Linear Regression: the Forward Algorithm to Replace Ridge. *Advances in Neural Information Processing Systems* 34 (2021), pp. 24430–24441.

[175] Vianney Perchet, Philippe Rigollet, Sylvain Chassang, Erik Snowberg, et al. Batched bandit problems. *Annals of Statistics* 44.2 (2016), pp. 660–681.

[176] Jan Peters, Katharina Mulling, and Yasemin Altun. Relative entropy policy search. In *AAAI Conference on Artificial Intelligence*. 2010.

[177] T Pham-Gia and TL Hung. The mean and median absolute deviations. *Mathematical and Computer Modelling* 34.7-8 (2001), pp. 921–936.

[178] Warren B Powell. *Approximate Dynamic Programming: Solving the curses of dimensionality*. Vol. 703. John Wiley & Sons, 2007.

[179] M. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 1994.

[180] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *Advances in neural information processing systems* 20 (2007).

[181] Tongzheng Ren, Tianjun Zhang, Csaba Szepesvári, and Bo Dai. A Free Lunch from the Noise: Provable and Practical Exploration for Representation Learning. *arXiv preprint arXiv:2111.11485* (2021).

[182] Herbert Robbins. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society* 58.5 (1952), pp. 527–535.

[183] Gerald S Rogers and Dennis L Young. Explicit maximum likelihood estimators for certain patterned covariance matrices. *Communications in Statistics-Theory and Methods* 6.2 (1977), pp. 121–133.

[184] Yoan Russac, Claire Vernade, and Olivier Cappé. Weighted linear bandits for non-stationary environments. *arXiv preprint arXiv:1909.09146* (2019).

[185] Daniel Russo and Benjamin Van Roy. Eluder dimension and the sample complexity of optimistic exploration. *Advances in Neural Information Processing Systems* 26 (2013).

[186] Daniel Russo and Benjamin Van Roy. Learning to Optimize via Information-Directed Sampling. *Advances in Neural Information Processing Systems* 27 (2014).

[187] Daniel Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, and Zheng Wen. A Tutorial on Thompson Sampling. *arXiv preprint arXiv:1707.02038* (2017).

[188] Mohsen Amini Salehi, Jay Smith, Anthony A Maciejewski, Howard Jay Siegel, Edwin KP Chong, Jonathan Apodaca, et al. Stochastic-based robust dynamic resource allocation for independent tasks in a heterogeneous computing system. *Journal of Parallel and Distributed Computing* 97 (2016), pp. 96–111.

[189] Jürgen Schmidhuber. Developmental robotics, optimal artificial curiosity, creativity, music, and the fine arts. *Connection Science* 18.2 (2006), pp. 173–187.

[190] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz. Trust Region Policy Optimization. In *International Conference on Machine Learning*. 2015, pp. 1928–1937.

[191] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* (2017).

[192] John Schulman, Philipp Moritz, Sergey Levine, Michael Jordan, and Pieter Abbeel. High-dimensional continuous control using generalized advantage estimation. In *International Conference on Learning Representations*. 2016.

[193] Steven L Scott. A modern Bayesian look at the multi-armed bandit. *Applied Stochastic Models in Business and Industry* 26.6 (2010), pp. 639–658.

[194] Abhin Shah, Devavrat Shah, and Gregory Wornell. A Computationally Efficient Method for Learning Exponential Family Distributions. *Advances in Neural Information Processing Systems* 34 (2021), pp. 15841–15854.

[195] Roshan Shariff and Csaba Szepesvári. Efficient planning in large MDPs with weak linear function approximation. *arXiv preprint arXiv:2007.06184* (2020).

[196] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *International Conference on Machine Learning*. 2014.

[197] Max Simchowitz, Kevin Jamieson, and Benjamin Recht. The simulator: Understanding adaptive sampling in the moderate-confidence regime. In *Conference on Learning Theory*. PMLR. 2017, pp. 1794–1834.

[198] Ankur Sinha, Tanmay Khandait, and Raja Mohanty. A gradient-based bilevel optimization approach for tuning hyperparameters in machine learning. *arXiv preprint arXiv:2007.11022* (2020).

[199] Aleksandrs Slivkins et al. Introduction to Multi-Armed Bandits. *Foundations and Trends® in Machine Learning* 12.1-2 (2019), pp. 1–286.

[200] Ingo Steinwart, Don Hush, and Clint Scovel. A Classification Framework for Anomaly Detection. *Journal of Machine Learning Research* 6.2 (2005).

[201] Vladimir N Sudakov. Gaussian measures, Cauchy measures and ε-entropy. In *Soviet Math. Dokl*. Vol. 10. 1969, pp. 310–313.

[202] R.S. Sutton and A.G. Barto. *Introduction to reinforcement learning*. Cambridge: MIT Press, 1998.

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

[204] Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems* 12 (1999).

[205] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy Gradient Methods for Reinforcement Learning with Function Approximation. In *Advances in Neural Information Processing Systems*. 2000.

[206] Chao Tao, Saúl Blanco, Jian Peng, and Yuan Zhou. Thresholding bandit with optimal aggregate regret. In *Advances in Neural Information Processing Systems*. 2019, pp. 11664–11673.

[207] P. Thodoroff, A. Durand, J. Pineau, and D. Precup. Temporal Regularization for Markov Decision Process. In *Advances in Neural Information Processing Systems*. 2018.

[208] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika* 25.3-4 (1933), pp. 285–294.

[209] Sebastian Thrun. Probabilistic robotics. *Communications of the ACM* 45.3 (2002), pp. 52–57.

[210] Andrea Tirinzoni, Aymen Al-Marjani, and Emilie Kaufmann. Near Instance-Optimal PAC Reinforcement Learning for Deterministic MDPs. *arXiv preprint arXiv:2203.09251* (2022).

[211] Andrea Tirinzoni, Matteo Pirotta, Marcello Restelli, and Alessandro Lazaric. An asymptotically optimal primal-dual incremental algorithm for contextual linear bandits. *arXiv preprint arXiv:2010.12247* (2020).

[212] E. Todorov, T. Erez, and Y. Tassa. Mujoco: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2012, pp. 5026–5033.

[213] Michel Tokic. Adaptive ε-greedy exploration in reinforcement learning based on value differences. In *Annual Conference on Artificial Intelligence*. Springer. 2010, pp. 203–210.

[214] Nilesh Tripuraneni and Lester Mackey. Single Point Transductive Prediction. *arXiv* (2019), arXiv–1908.

[215] George Tucker, Surya Bhupatiraju, Shixiang Gu, Richard Turner, Zoubin Ghahramani, and Sergey Levine. The Mirage of Action-Dependent Baselines in Reinforcement Learning. In *International Conference on Machine Learning*. 2018, pp. 5015–5024.

[216] Michal Valko, Rémi Munos, Branislav Kveton, and Tomáš Kocák. Spectral bandits for smooth graph functions. In *International Conference on Machine Learning*. PMLR. 2014, pp. 46–54.

[217] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. *arXiv preprint arXiv:1509.06461* (2015).

[218] Benjamin Van Roy and Shi Dong. Comments on the du-kakade-wang-yang lower bounds. *arXiv preprint arXiv:1911.07910* (2019).

[219] Shankar Vembu, Thomas Gartner, and Mario Boley. Probabilistic structured predictors. *arXiv preprint arXiv:1205.2610* (2012).

[220] Arun Verma, Manjesh Hanawal, Arun Rajkumar, and Raman Sankaran. Censored semi-bandits: A framework for resource allocation with censored feedback. *Advances in Neural Information Processing Systems* 32 (2019).

[221] Sofía S Villar, Jack Bowden, and James Wason. Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges. *Statistical science: a review journal of the Institute of Mathematical Statistics* 30.2 (2015), p. 199.

[222] Pascal Vincent, Hugo Larochelle, Yoshua Bengio, and Pierre-Antoine Manzagol. Extracting and composing robust features with denoising autoencoders. In *Proceedings of the 25th international conference on Machine learning*. 2008, pp. 1096–1103.

[223] Walter Vogel. An asymptotic minimax theorem for the two armed bandit problem. *The Annals of Mathematical Statistics* 31.2 (1960), pp. 444–451.

[224] V Vovk. *On-line competitive linear regression*. Tech. rep. Technical Report CSD-TR-97-13, Department of Computer Science, Royal …, 1997.

[225] Volodya Vovk. Competitive on-line statistics. *International Statistical Review* 69.2 (2001), pp. 213–248.

[226] Andrew Wagenmaker and Kevin Jamieson. Instance-Dependent Near-Optimal Policy Identification in Linear MDPs via Online Experiment Design. *arXiv preprint arXiv:2207.02575* (2022).

[227] Andrew J Wagenmaker, Max Simchowitz, and Kevin Jamieson. Beyond no regret: Instance-dependent pac reinforcement learning. In *Conference on Learning Theory*. PMLR. 2022, pp. 358–418.

[228] Abraham Wald. *Sequential analysis*. John Wiley, 1947.

[229] Ruosong Wang, Russ R Salakhutdinov, and Lin Yang. Reinforcement learning with general value function approximation: Provably efficient approach via bounded eluder dimension. *Advances in Neural Information Processing Systems* 33 (2020), pp. 6123–6135.

[230] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. Dueling network architectures for deep reinforcement learning. In *International Conference on Machine Learning*. 2016, pp. 1995–2003.

[231] L. Weaver and N. Tao. The Optimal Reward Baseline for Gradient·Based Reinforcement Learning. In *Advances in Neural Information Processing Systems*. 2001.

[232] Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*. Vol. 2. 3. MIT press Cambridge, MA, 2006.

[233] R.J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning* 8.3-4 (1992), pp. 229–256.

[234] Ronald J Williams and Jing Peng. Function optimization using connectionist reinforcement learning algorithms. *Connection Science* 3.3 (1991), pp. 241–268.

[235] Cathy Wu, Aravind Rajeswaran, Yan Duan, Vikash Kumar, Alexandre M Bayen, Sham Kakade, et al. Variance reduction for policy gradient with action-dependent factorized baselines. In *International Conference on Learning Representations*. 2018.

[236] Lin Yang and Mengdi Wang. Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. In *International Conference on Machine Learning*. PMLR. 2020, pp. 10746–10756.

[237] Andrea Zanette, David Brandfonbrener, Emma Brunskill, Matteo Pirotta, and Alessandro Lazaric. Frequentist regret bounds for randomized least-squares value iteration. In *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 1954–1964.

[238] Andrea Zanette, Mykel J Kochenderfer, and Emma Brunskill. Almost horizon-free structure-aware best policy identification with a generative model. *Advances in Neural Information Processing Systems* 32 (2019).

[239] Xuezhou Zhang, Yuda Song, Masatoshi Uehara, Mengdi Wang, Alekh Agarwal, and Wen Sun. Efficient reinforcement learning in block mdps: A model-free representation learning approach. In *International Conference on Machine Learning*. PMLR. 2022, pp. 26517–26547.

[240] Tingting Zhao, Gang Niu, Ning Xie, Jucheng Yang, and Masashi Sugiyama. Regularized policy gradients: direct variance reduction in policy gradient estimation. In *Asian Conference on Machine Learning*. PMLR. 2016, pp. 333–348.

[241] Jie Zhong, Yijun Huang, and Ji Liu. Asynchronous parallel empirical variance guided algorithms for the thresholding bandit problem. *arXiv preprint arXiv:1704.04567* (2017).

[242] Yinglun Zhu, Sumeet Katariya, and Robert Nowak. Robust Outlier Arm Identification. In *Proceedings of the 37th International Conference on Machine Learning*. Ed. by Hal Daumé III and Aarti Singh. Vol. 119. Proceedings of Machine Learning Research. PMLR, 13–18 Jul 2020, pp. 11566–11575.

[243] Honglei Zhuang, Chi Wang, and Yifan Wang. Identifying outlier arms in multi-armed bandit. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*. 2017, pp. 5210–5219.