Structured Stochastic Bandits

STEFAN MAGUREANU

Licentiate Thesis
Stockholm, Sweden 2016
Abstract

In this thesis we address the multi-armed bandit (MAB) problem with stochastic rewards and correlated arms. Particularly, we investigate the case when the expected rewards are a Lipschitz function of the arm, and the learning to rank problem, as viewed from a MAB perspective. For the former, we derive a problem specific lower bound and propose both an asymptotically optimal algorithm (OSLB) and a (pareto) optimal, algorithm (POSLB). For the latter, we construct the regret lower bound and determine its closed form for some particular settings, as well as propose two asymptotically optimal algorithms PIE and PIE-C. For all algorithms mentioned above, we present performance analysis in the form of theoretical regret guarantees as well as numerical evaluation on artificial datasets as well as real-world datasets, in the case of PIE and PIE-C.
Acknowledgements

I would like to extend my deepest gratitude to my advisors Prof. Alexandre Proutiere and prof. Mikael Johansson, for taking a chance on me, despite my lack of even basic background in statistics. I would particularly like to thank Prof. Alexandre Proutiere for his support and guidance, without which I would not have made it this far. I would also like to thank Prof. Proutiere and Associate Prof. Richard Combes for having the patience to teach me how to think independently, how to tackle problems and write proofs. Your contribution to my growth cannot be overstated. Thank you for making this journey be exactly what I dreamed it would be when I decided to pursue a PhD degree.

I would like to thank all my friends, past and present colleagues, faculty and staff at the Automatic Control department for providing such a wonderful work environment. I would like thank Sadegh, Vahan, Demia, Jaeseong, Zhenhua and Riccardo for uncountable discussions, from which I always walked away having learned something new. I would like to thank the administrative staff for never allowing us to feel the true weight of bureaucracy, we are truly spoiled.

Last but not least, I would like to thank my friends and family for their continuous love and support.
# Contents

## 1 Introduction

1.1 Applications .................................................. 1
  1.1.1 Collaborative Filtering (CF) ............................. 2
  1.1.2 Learning to Rank ....................................... 2
  1.2 Thesis outline .............................................. 3
  1.3 Contributions .............................................. 3

## 2 From Unstructured to Structured Bandits

2.1 Classical Bandits ............................................ 5
  2.1.1 Regret Lower Bounds .................................. 5
  2.1.2 Upper Confidence Bound Algorithms .................. 8
  2.1.3 KL-UCB Index and Algorithm ......................... 10
  2.1.4 Thompson Sampling .................................... 12
  2.2 Structured Bandits ........................................ 13
  2.2.1 A Generic Model ...................................... 14
  2.2.2 Application to Specific Bandit Problems .......... 14
  2.2.3 Regret Lower Bounds ................................ 15

## 3 Multi-armed Bandits with Lipschitz-Continuous Rewards

3.1 Problem Statement and Related Work ...................... 17
3.2 Models ....................................................... 18
3.3 Regret Lower Bound ........................................ 19
3.4 Algorithms .................................................. 21
  3.4.1 Index of Arms ......................................... 21
  3.4.2 The OSLB Algorithm .................................. 22
  3.4.3 The POSLB Algorithm ................................ 23
3.5 Regret Analysis ............................................ 24
  3.5.1 Concentration Inequalities ............................ 24
  3.5.2 Finite time analysis of OSLB ......................... 25
  3.5.3 Regret Guarantees of POSLB .......................... 26
## Contents

3.5.4 Numerical Evaluation ............................................. 27
3.6 Towards Optimal Continuous Lipschitz Bandits .................... 27
3.6.1 Numerical Evaluation .............................................. 28
3.7 Contextual Bandit with Similarities .................................. 29
3.7.1 Model .............................................................. 29
3.7.2 Regret Lower Bound ............................................... 30
3.7.3 Algorithms ......................................................... 31
3.8 Conclusions ............................................................ 32

4 Learning to Rank using Multi-Armed Bandits .......................... 33
4.1 Problem Introduction .................................................... 33
4.1.1 Motivation .......................................................... 33
4.1.2 Related work ....................................................... 35
4.2 System Model .......................................................... 36
4.2.1 Users, Items, and Side-information .............................. 36
4.2.2 Rewards and Regret ............................................... 37
4.3 A Single Group of Items and Users ................................... 38
4.3.1 Regret Lower Bound .............................................. 39
4.3.2 Optimal Algorithms ............................................... 41
4.3.3 Proofs: Lower Bounds ............................................ 43
4.3.4 Proof: Regret Upper bound for PIE(l) ........................... 45
4.4 Known Topic ............................................................ 49
4.5 Known User-Class and Unknown Topic ............................... 50
4.5.1 Optimal Algorithms ............................................... 51
4.5.2 Proof: Regret Upper Bound for PIE-C(l, d) .................... 53
4.6 Numerical Experiments ................................................. 54
4.6.1 Artificial Data ....................................................... 54
4.6.2 Real-world Data .................................................... 56
4.7 Conclusions ............................................................. 57

5 Conclusions and Future Work ............................................ 59

Bibliography ............................................................... 61

6 Appendix ....................................................................... 65
6.1 Proof of Theorem 2.2.1 .................................................... 65
6.2 Proof of Theorem 3.7.1 .................................................... 66
6.3 Proof of Theorem 3.5.2 - OSLB Finite Time Analysis .............. 70
6.3.1 Continuity of solutions to parametric linear programs ......... 73
6.4 Proof of Theorem 3.5.3 - Finite Time Analysis of POSLB ........ 74
6.5 Proof of Theorem 3.5.4 - POSLB Asymptotic Regret for Lipschitz Bandits 78
6.5.1 Proof of Theorem 3.5.4 in Deterministic Setting ............... 86
6.6 Supporting Lemmas for the Proof of Theorem 4.3.2 ................ 91
The Multi-Armed Bandit (MAB) problem is a fundamental problem in reinforcement learning, originating in the early 1900s, with its roots in medical trials. In the original setting of the problem, a decision maker has access to a set of $K$, new, independent treatments (or arms), each with a fixed but unknown probability $\theta_1, \theta_2, \ldots, \theta_K$ to cure an illness. Time evolves in rounds, $1, \ldots, N$, with $N$ possibly very large. At each round, the decision maker is presented with a patient and has to offer one (and only one) of the $K$ treatments. Before the beginning of the next round, they get to observe whether the patient is saved or dies. The goal of the exercise is to design an algorithm that assigns treatments to patients such that, on average, it saves the largest possible number of patients. Equivalently, the performance metric is given by the difference in expected number of survivors achieved by the algorithm relative to always assigning the best treatment. This quantity is aptly named regret in the related literature.

This original formulation of the problem was solved in 1985 by Lai and Robbins, who provide an asymptotic lower bound on the regret achievable by any allocation policy and propose an algorithm matching this performance limit. However, there are numerous variations of the problem that remain unsolved. For instance, what can be done when the decision maker is also aware of correlations between arms? If two treatments $A$ and $B$ are based largely on the same drug, then surely their chances of curing a patient are, while not identical, at least similar. Therefore, we can obtain information about the performance of treatment $A$ based on the observations of treatment $B$ and vice-versa. However, how much information one can obtain in such scenarios and what reduction in the number of deaths can be expected when this information is used is still unclear. Similarly, what can be said when the number of treatments is infinite or very large compared to the number of patients and therefore we cannot explore all treatments (for example choosing the best dose for a drug)?

1.1 Applications

Multi-armed bandits have seen a substantial growth in popularity throughout the research community as they provide a very natural model for data acquisition tasks, particularly
in the monetarily rewarding field of web advertising. Since it is inefficient to first collect vast amounts of data in order to discover interesting features of a population of users, MABs provide a very natural framework for discovering and exploiting these features in an online manner. This increases revenue by providing not only much lower response time on learning tasks but also minimal cost of exploration. In contrast, supervised learning algorithms require collecting (possibly very large amounts of) learning data before being able to exploit any detected features.

1.1.1 Collaborative Filtering (CF)

CF recommender systems attempt to estimate feature vectors of users and items, in \(d\) dimensions, from a large, usually sparse, rating matrix \(A \in \mathbb{R}^{m \times n}\) (where \(m\) is the number of users and \(n\) the number of items). Conventionally, this is done by assuming \(A\) is low rank and decomposing \(A\) into a product of two matrices \(M \in \mathbb{R}^{m \times d}\) and \(F \in \mathbb{R}^{d \times n}\). Now \(M\) and \(F\) are said to contain the feature vectors of users and items, respectively, and the preference of a user \(u\) for item \(i\) is simply the scalar product of their feature vectors. The goal of the recommender system is to find, for each user, the item with the highest preference. This problem is particularly difficult when trying to display recommendations for new users / items - the so called cold start problem. However, we can think of this problem as a MAB where the arms are items and the expected rewards of an arm is the scalar product between the feature vector of the user and that of the item. Now solving this problem will not only yield and algorithm that minimizes the expected exploration cost, but also provides online estimates of all feature vectors (rather than having to periodically compute the factorization of \(A\)).

Another approach to the design of CF recommenders would be to exploit the Lipschitz structure of features. Assuming a service recommending music to a user base, and considering age to be one of the features. It is intuitive that the average tastes of people in two neighboring age brackets would exhibit a form of Lipschitz continuity. Thus, if a genre of music does not benefit from popularity from users of age 25, it will most likely not be popular with users of age 26 or 24 either.

1.1.2 Learning to Rank

Ranking of results is one of the key features of any search engines. In this problem, a search engine receives a query from a user, it then has to present an order list of results, sorted in order of relevance. For example, if a user searches for the term "typhoon" the search engine must discover whether the user is interested in the hurricane or the military aircraft. To this end, the engine must discover the more popular meaning of the term. The same can be said about the more informative articles that match the query. If we consider the user parses the presented list in order and the feedback is given by the position in the list that is clicked, this becomes a MAB problem again. We will see in the chapter dedicated to Learning to Rank, how using efficient MAB algorithms can greatly reduce the cost of exploration.
1.2 Thesis outline

The Multi-Armed Bandit family of problems is fundamental to the field of reinforcement learning and countless applications can be identified (clinical trials, radio channel selection, A/B testing etc). The goal of this thesis to advance the understanding of what represent efficient algorithms in settings where arms are correlated. To this end we will first introduce the most popular techniques use to solve the classic MAB with independent arms and proceed to detail the state of the art in the correlated-arm setting. Finally we will present the solutions to two popular problems fitting this setting: the Lipschitz Bandit and Learning to Rank.

We now proceed to describe the structure of the thesis. The following chapter, Chapter 2, presents the state of the art in MAB problems without structures. Here we will present the lower bound result of Lai and Robbins [1] and its proof. We will also introduce the idea behind two of the most popular bandit algorithms, UCB [2] and Thompson Sampling [3].

In Chapter 3, we introduce the Lipschitz bandit problem. We present the regret lower bound under this kind of correlation and propose two algorithms OSLB and POSLB. We prove OSLB is asymptotically optimal, but complex, and POSLB is (pareto) optimal asymptotically but computationally light. We then evaluate POSLB’s numerical performance and present the extensions of the algorithms and regret lower bound to Contextual and Continuous Lipschitz bandits.

Chapter 4, presents our MAB approach to the well known problem of learning to rank. Again, we derive lower bounds and propose two algorithms, PIE and PIE-C, which are provably asymptotically optimal in two widely applicable settings of the learning to rank problem.

Finally, in Chapter 5, we present the conclusion and future research directions.

1.3 Contributions

During my studies so far, I have worked on the following papers:


Additionally, Section 2.2 and Sections 3.4.3 and 3.5.3 (the POSLB algorithm and its analysis) are based on yet unpublished work.
In this chapter, I will present a short summary of the basic theoretical tools used to solve the unstructured MAB problem. First, I will introduce a short proof of the lower bounds on the asymptotic regret achievable by any algorithm in this setting. Second, I will present the driving principles behind the UCB and Thompson Sampling algorithms, with a focus on the former.

### 2.1 Classical Bandits

#### 2.1.1 Regret Lower Bounds

Consider a MAB setting with $K$ arms, having expected rewards given by the parameter $\theta = \{\theta_1, \theta_2, \ldots, \theta_K\}$. Each play of an arm $k$ yields i.i.d. rewards drawn from a Bernoulli distribution of mean $\theta_k$. Without loss of generality, assume $\theta_1 > \theta_2 > \cdots > \theta_K$. We denote by $X_\pi(n) \in \{0, 1\}$ the reward observed by an algorithm $\pi$ at round $n$ and by $t_k^\pi(n)$ the number of times $\pi$ selects arm $k$ up to time $n$. Let $\Pi$ be the set of all possible uniformly good sequential allocation rules $\pi$ defined as rules that satisfy $\forall k > 1$, $t_k^\pi(T) = o(T^\alpha)$ for all $\theta \in (0, 1)^K$ and $\alpha > 0$. We are interested in a lower bound on the regret of any algorithm $\pi \in \Pi$, defined as follows:

$$R^\pi(T) = T\theta_1 - \mathbb{E}\left[\sum_{n=1}^{T} X^\pi(n)\right]$$

This bound was obtained by Lai and Robbins in their landmark paper [1], as presented in the following theorem:

**Theorem 2.1.1** For all $\pi \in \Pi$ and for all $\theta \in (0, 1)^K$ we have:

$$\lim_{T \to \infty} \inf \frac{R^\pi(T)}{\log(T)} \geq \sum_{k=2}^{K} \frac{\theta_1 - \theta_k}{I(\theta_k, \theta_1)} \text{ a.s.}$$
Here, \( I(x, y) \) is the Kullback-Leibler divergence between two probability distributions of mean \( x \) and \( y \). Since we assume the rewards are drawn from Bernoulli distributions, we have, \( I(x, y) = x \log(x/y) + (1 - x) \log((1 - x)/(1 - y)) \). The requirement that \( \pi \) is \textit{uniformly good} stems from the fact that we are interested in algorithms that perform well for all parameters \( \theta \). Otherwise, an algorithm that always picks arm \( k \) will generate no regret when \( k \) is optimal, but will have linear regret in all other problem instances. \textit{Uniformly good} algorithms always exist, for example UCB and Thompson Sampling algorithms are \textit{uniformly good}, as we will see in the following sections.

**Proof.** Fix an arm \( k > 1 \) and constants \( \rho > 0 \) and \( \varepsilon > 0 \) such that \( I(\theta_k, \theta_1 + \varepsilon) - I(\theta_k, \theta_1) < \rho I(\theta_k, \theta_1) \). Consider a parameter \( \lambda = \{\lambda_1, \ldots, \lambda_K\} \) such that \( \lambda_i = \theta_i \) for all \( i \neq k \) and \( \lambda_k = \theta_1 + \varepsilon \). \( \lambda \) is therefore identical to \( \theta \) but with \( \theta_k \) replaced by \( \theta_1 + \varepsilon \). \( \lambda \) is a confusing parameter in the sense that it is the parameter under which arm \( k \) is optimal (rather than arm 1) and is most likely to generate samples resembling those of \( \theta \). Thus, under \( \lambda \) a \textit{uniformly good} algorithm will play arm \( k \) most of the time, while under \( \theta \) it would choose arm 1. Note that even if the true parameter is \( \theta \), if the observed rewards appear to be generated by \( \lambda \), a uniformly good algorithm would still play \( k \) most of the time. The idea behind the proof is to show that the probability of observing such rewards under a \textit{uniformly good} algorithm tends to 0 when \( T \) goes to infinity.

Let us make the following slight abuse of notation and denote by \( X_n \) the reward observed on the \( n \)-th play of arm \( k \) under parameter \( \theta \). Define \( f(X_n, \theta) = \theta_k \) if \( X_n = 1 \) and \( 1 - \theta_k \) otherwise. We further define for all \( a > 1 \):

\[
L_a = \sum_{n=1}^{a} \log \left( \frac{f(X_n, \theta)}{f(X_n, \lambda)} \right).
\]

Letting \( a \to \infty \), we obtain:

\[
\lim_{a \to \infty} \frac{L_a}{a} = \theta_k \log \left( \frac{\theta_k}{\lambda_k} \right) + (1 - \theta_k) \log \left( \frac{1 - \theta_k}{1 - \lambda_k} \right) \text{ a.s.}
\]

where the first term in the sum corresponds to the rounds producing reward \( X_n = 1 \) (from the LLN we know the number of such rounds, when \( a \to \infty \), tends to \( a\theta_k \) a.s., under parameter \( \theta \)) and second to those producing reward 0. Consequently we have \( \lim_{a \to \infty} L_a/a = I(\theta_k, \lambda_k) \) a.s., by the definition of the KL divergence between two Bernoulli distributions. For any constant \( b > 0 \) we have:

\[
\lim_{T \to \infty} \mathbb{P}_\theta[L_a > b(1 + \rho)I(\theta_k, \lambda_k) \log(T) \text{ for some } a < b \log(T)] = 0 \quad (2.1)
\]

since, for all \( a < b \log(T) \):

\[
\lim_{T \to \infty} L_a/(b \log(T)) \leq \lim_{T \to \infty} L_{b \log(T)}/(b \log(T)) = I(\theta_k, \lambda_k) \text{ a.s.}
\]

and therefore, for all \( a < b \log(T) \):

\[
b(1 + \rho)I(\theta_k, \lambda_k) \log(T) > b \log(T)I(\theta_k, \lambda_k) \geq L_a \text{ a.s.}
\]
Define the following event, for all \( a \in \mathbb{N} \):
\[
C_a = \{ t_k(T) = a, L_a \leq b(1 + \rho)I(\theta_k, \lambda_k) \log(T) \},
\]
and consider the decomposition of the following event:
\[
\{ t_k(T) < b \log(T), L_{t_k(T)} \leq b(1 + \rho)I(\theta_k, \lambda_k) \log(T) \} = \bigcup_{a<b \log(T)} C_a
\]
Next, we describe the probability of each of the events in the decomposition under \( \lambda \):
\[
\mathbb{P}_\lambda[C_a] = \int_{C_a} d\mathbb{P}_\lambda = \int_{C_a} \prod_{i=1}^{a} \frac{f(X_i, \lambda)}{f(X_i, \theta)} d\mathbb{P}_\theta
\]  
(2.2)
where the last equality is due to the change of measure from \( d\mathbb{P}_\lambda \) to \( d\mathbb{P}_\theta \). Notice the following equality holds:
\[
L_a = \sum_{i=1}^{a} \log \left( \frac{f(X_i, \theta)}{f(X_i, \lambda)} \right) = -\sum_{i=1}^{a} \log \left( \prod_{i=1}^{a} \frac{f(X_i, \lambda)}{f(X_i, \theta)} \right)
\]
Consequently, from (2.2), we have, \( \forall a < b \log(T) \):
\[
\mathbb{P}_\lambda[C_a] = e^{-L_a} \mathbb{P}_\theta[C_a] \geq e^{-b(1+\rho)I(\theta_k, \lambda_k) \log(T)} \mathbb{P}_\theta[C_a]
\]
and hence,
\[
\mathbb{P}_\lambda[t_k(T) < b, L_{t_k(T)} \leq b(1 + \rho)I(\theta_k, \lambda_k) \log(T)] \geq T^{-b(1+\rho)I(\theta_k, \lambda_k)} \mathbb{P}_\theta[t_k(T) < b, L_{t_k(T)} \leq b(1 + \rho)I(\theta_k, \lambda_k) \log(T)].
\]  
(2.3)
(2.4)
Since the plays of arm \( k \) are governed by a \textit{uniformly good} algorithm, and \( k \) is optimal under \( \lambda \) we have:
\[
\mathbb{E}_\lambda[T - t_k(T)] = o(T^\alpha), \forall \alpha > 0
\]
and therefore
\[
(T - b \log(T)) \mathbb{P}_\lambda[t_k(j) < b \log(T)] = o(T^\alpha).
\]
Dividing both sides by \( T \), we further obtain:
\[
\mathbb{P}_\lambda[t_k(j) < b \log(T)] = o(T^{\alpha-1})
\]
Since the event \( \{ t_k(j) < b \log(T) \} \subset \bigcup_a C_a \), we have
\[
\mathbb{P}_\lambda[t_k(j) < b \log(T)] \geq \mathbb{P}_\lambda[t_k(T) < b \log(T), L_{t_k(T)} \leq b(1 + \rho)I(\theta_k, \lambda_k) \log(T)]
\]
which, together with (2.3) implies:
\[
o(T^\alpha - 1) = \mathbb{P}_\lambda [t_k(T) < b, L_{t_k}(T)] \geq T^{-b(1+\rho)I(\theta_k, \lambda_k)} \mathbb{P}_\theta[t_k(T) < b, L_{t_k}(T) \leq (1 + \rho)I(\theta_k, \lambda_k) \log(T)].
\]

Remembering Equation 2.1, and taking the limit when \( T \to \infty \) we then have:
\[
\lim_{T \to \infty} o(T^\alpha - 1) \geq \lim_{T \to \infty} T^{-b(1+\rho)I(\theta_k, \lambda_k)} \mathbb{P}_\theta[t_k(T) < b \log(T)]
\]
and hence:
\[
\lim_{T \to \infty} o(T^\alpha - 1 + b(1+\rho)I(\theta_k, \lambda_k)) \geq \lim_{T \to \infty} \mathbb{P}_\theta[t_k(T) < b \log(T)].
\]
Since we want the probability on the right hand side of the above equation to go to 0 when \( T \to \infty \), we can choose \( b = \lfloor (1 + 2\rho)I(\theta_k, \lambda_k) \rfloor^{-1} \) and consequently:
\[
\lim_{T \to \infty} o(T^\alpha - 1 + b(1 + (1+2\rho)) \log(T)) \geq \lim_{T \to \infty} \mathbb{P}_\theta[t_k(T) < (1 + 2\rho)I(\theta_k, \lambda_k)^{-1}\log(T)].
\]
Letting \( \alpha \to 0 \), we then have:
\[
\lim_{T \to \infty} \mathbb{P}_\theta[t_k(T) < (1 + 2\rho)I(\theta_k, \lambda_k)^{-1}\log(T)] = 0.
\]
Finally, letting \( \rho \to 0 \), and therefore \( \varepsilon \to 0 \) and \( \lambda_k \to \theta_1 \), concludes the proof:
\[
\lim_{T \to \infty} \mathbb{P}_\theta \left[ t_k(T) < \frac{\log(T)}{I(\theta_k, \theta_1)} \right] = 0.
\]

The general idea of this proof can be applied to structured bandits and even more general problems like control of Markov chains, as one can find in the results of Graves and Lai in [4]. In [4], the authors use the same change-of-measure argument but on more complex probability measures to extend the result in Theorem 2.1.1 to MDPs. The proof, however, is significantly more involved and will not be presented in this thesis.

Having established the theoretical limit of the performance of any uniformly good algorithm in the classical bandit setting, let us explore the two most popular classes of optimal algorithms.

### 2.1.2 Upper Confidence Bound Algorithms

Upper Confidence Bounds (UCB) algorithms work by computing a confidence interval, of increasing accuracy over time, on the empirical average reward of each arm and select the arm with highest upper bound of this interval. This upper bound is then used as the index of an arm, as per Whittle’s result. This family of algorithms was first introduced in 1987 by the same Tze-Leung Lai in [5]. Later in 2002, finite time regret guarantees are introduced for the UCB algorithm in [6] and its asymptotically optimal version KL-UCB.
2.1. Classical Bandits

(the same algorithm originally as in [5]) analyzed in [2] in 2011. These algorithms are referred in the literature as optimistic since they consider each arm’s expected reward to be the highest that could have generated the observations up to time \( n \) with high probability. Here, the threshold for high probability varies from one algorithm to another but is usually between \( 1 - n^{-2} \) and \( 1 - (n \log(n))^{-1} \). In this section we will briefly examine the intuition behind the design of confidence bounds in the UCB algorithm in [6]. Next, we will see how using more powerful concentration bounds yields the asymptotically optimal KL-UCB algorithm.

Gittins Index Theorem

Let us first start by introducing a surprising result of Gittins in [7]. The Gittins index is defined as a real scalar value characterizing the reward that can be achieved by a stochastic process given its state, the reward function and a probability of termination. In the seminal paper [7], it is proven that, in the case of the classical MAB problem, an algorithm that always plays the arm with the largest Gittins index is optimal and the index of an arm \( k \) is computed using only the observations of \( k \) alone. The Upper Confidence Bounds algorithms presented in this section are similar in the sense that they also compute an index for each arm and play the arm with the greatest index - KL-UCB being asymptotically optimal. Note that Gittins’ result no longer holds once we add structure to the parameter \( \theta \).

UCB Index and Algorithm

Denote by \( X(n) \) the reward obtained at round \( n \) and by \( k(n) \) the arm selected at round \( n \). Let us introduce the following notation for the empirical mean of an arm \( k \) at round \( n \), \( \hat{\theta}_k(n) = 1/t_k(n) \sum_{i=1}^{n} 1\{ k(n) = k \} X(n) \), and the best arm, \( k^* = \arg \max_k \theta_k \). At a round \( n \) the UCB algorithm ([6]) selects the arm maximizing the following index:

\[
b_k^{UCB}(n) = \hat{\theta}_k(n) + \sqrt{\frac{\log(n)}{t_k(n)}}
\]

where the second term of the right hand size is casually referred to as the exploration bonus in related literature, and is supposed to account for uncertainty in the empirical estimate \( \hat{\theta}_k(n) \). Indeed, the more plays of arm \( k \) one has before time \( n \), the lower this bonus becomes. The goal of this bonus is to ensure the true average reward of \( k \) is below \( b_k^{UCB}(n) \), with probability: \( \mathbb{P}[\theta_k < b_k^{UCB}(n)] = 1 - p(n) \leq 1 - 1/n \). Why \( p(n) = 1/n \)?

Consider a generic risk level \( p(n) \) and the associated exploration bonus \( \delta(n) \) such that:

\[
\exp(-t_k(n)\delta(n)^2) = p(n)
\]

and consequently, by taking the logarithm of the above expression and solving for \( \delta(n) \):

\[
\sqrt{\frac{\log(1/p)}{t_k(n)}} = \delta(n).
\]
Note that from Hoeffding’s inequality, we have:
\[ P[|\theta_k - \hat{\theta}_k(n)| > \delta(n)] \leq \exp(-t_k(n)\delta(n)^2) = p(n). \]

Let us consider the following sets of rounds: \( B = \{n \leq T : \theta_{k^*} > b_{k^*}^{UCB}(n)\} \), when the index of the optimal arm is below its true average reward (i.e. the reward of \( k^* \) is underestimated) and \( \overline{B} \), its complement. In order to have regret scaling as \( O(\log(T)) \) we must therefore have that the regret caused by rounds in \( B \) is in \( O(\log(T)) \) and also, the regret caused by rounds in \( \overline{B} \) is in \( O(\log(T)) \). Let us first consider the rounds in \( B \). Then, if a suboptimal arm \( k \) is chosen at round \( n \in B \), we must have \( b_k^{UCB}(n) \geq b_{k^*}^{UCB}(n) \geq \theta_{k^*} \), hence \( \hat{\theta}_k(n) + \delta(n) \geq \theta_{k^*} \) and therefore \( \delta(n) \geq \theta_{k^*} - \hat{\theta}_k(n) \). Replacing in Equation (2.5) and solving for \( t_k(n) \) proves the number of plays of \( k \) in rounds in \( \overline{B} \) is bounded as follows:
\[ t_k(n) \leq \frac{\log(1/p(n))}{(\theta_{k^*} - \hat{\theta}_k(n))^2} \] (2.6)

Since \( \hat{\theta}_k(n) \to \theta_k \) when \( t_k(n) \to \infty \), it holds that \( \liminf_{n \to \infty} \theta_{k^*} - \hat{\theta}_k(n) > 0 \) w.p. 1.

Now, in order to have \( t_k(T) = O(\log(T)) \) we must have \( \log(p(T)) \geq O(\log(1/T)) \).

We now turn our attention to the regret of rounds in \( \overline{B} \). The easiest way of bounding the regret of rounds in \( B \) is simply bounding the set’s expected cardinality \( E[|B|] \). Thus:
\[ E[|B|] = \sum_{n=1}^{T} P[1 \{n \in B\}] \leq \sum_{n=1}^{T} p(n) \leq O(\log(n)). \] (2.7)

Choosing \( p(n) = 1/n \) then results in the regret of both rounds in \( B \) and \( \overline{B} \) will belong to \( O(\log(T)) \). Using \( p(n) = 1/n \), the exploration bonus \( \delta(n) \) becomes:
\[ \delta(n) = \sqrt{\frac{\log(n)}{t_k(n)}} \]

### 2.1.3 KL-UCB Index and Algorithm

KL-UCB ([5], [2]) follows the same principle as UCB [6], but aims to be asymptotically optimal. Hence, in order to reduce the expected cardinality of \( B = \{n < T : b_{k^*}^{KLUCB}(n) < \theta_k(n)\} \) to \( o(\log(T)) \) and we now choose \( p(n) = (n \log(n))^{-1} \). Furthermore, we also need to use a more powerful concentration inequality, proposed by Garivier in [8], which we state here for completeness:

**Theorem 2.1.2** For all \( \delta > 0 \) and \( n \geq 1 \):
\[ P[\exists t \in \{1, \ldots, n\} : t_k(n)I(\hat{\theta}_k(n), \theta_k) \geq \delta] \leq 2e[\delta \log(n)] \exp(-\delta) \]

As before, we solve for \( \delta \), and we must have:
\[ 2e\delta \exp(-\delta) = (n \log(n)^2)^{-1}, \]
Taking $\delta = \log(n) + 3 \log \log(n)$ yields the desired property $p(n) < (n \log(n))^{-1}$. Consequently, the index of KL-UCB is derived to be:

$$b_{KLUCB}^k(n) = b_{KL}^k(n) = \inf \{ q \geq \hat{\theta}_k(n) : t_k(n) I(\hat{\theta}_k(n), q) \geq \log(n) + 3 \log \log(n) \},$$

exactly matching that in [2]. Note that the index of KL-UCB, represents the lowest value $q > \hat{\theta}_k(n)$ for which a likelihood ratio test of risk $(n \log(n))^2 - 1$ rejects the hypothesis $H_0 = \{ \theta_k > q \}$. We present how to derive the KL-UCB index from this likelihood ratio test in Section 3.4.1.

**KL-UCB: Proof of Optimality**

In this section, we will parse a short proof of optimality of KL-UCB. The proof revolves around splitting the rounds in two categories: *good* and *bad*. Intuitively, *bad* rounds are rounds in which $\hat{\theta}_k(n)$ is poorly estimated or $\theta_1 > b_{KL}^1(n)$, meaning the index of the optimal arm underestimates its true distribution. We will formally define what *good* and *bad* rounds are later. The proof relies on bounding the expected cardinalities of these sets. In the following proof, we will make use of the following remarkable result, introduced by Combes and Proutiere in their paper [9]. We re-state the lemma here for completeness:

**Lemma 2.1.1** Let $k \in K$, and $\epsilon > 0$. Define $\mathcal{F}_n$ the $\sigma$-algebra generated by $(X(t))_{1 \leq t \leq n}$, $\Lambda \subset \mathbb{N}$ be a (random) set of instants. Assume that there exists a sequence of (random) sets $(\Lambda(s))_{s \geq 1}$ such that (i) $\Lambda \subset \bigcup_{s \geq 1} \Lambda(s)$, (ii) for all $s \geq 1$ and all $n \in \Lambda(s)$, $t_k(n) \geq \epsilon s$, (iii) $|\Lambda(s)| \leq 1$, and (iv) the event $n \in \Lambda(s)$ is $\mathcal{F}_n$-measurable. Then for all $\delta > 0$:

$$\mathbb{E} \left[ \sum_{n \geq 1} \mathbb{I} \{ n \in \Lambda, |\hat{\theta}_k(n) - \theta_k| > \delta \} \right] \leq \frac{1}{\epsilon \delta^2}. \quad (2.8)$$

The following theorem implies the asymptotic optimality of KL-UCB:

**Theorem 2.1.3** For all $\theta$, $\delta > 0$, $n \in \mathbb{N}$, under KL-UCB, we have there exists a constant $C > 0$:

$$\mathbb{E}[t_k(T)] \leq \frac{\log(n) + 3 \log \log(n)}{I(\theta_k + \delta, \theta_1)} + C + \delta^{-2}$$

Taking the limit when $T \to \infty$, and obtain:

$$\lim_{T \to \infty} \sup_{T \to \infty} \frac{\mathbb{E}[t_k(T)]}{\log(T)} \leq \frac{1}{I(\theta_k + \delta, \theta_1)}.$$

Letting $\delta \to 0$, proves KL-UCB is asymptotically optimal.

**Proof.** Let us define $B = \{ n \in \mathbb{N} : b_{KL}^1(n) < \theta_1 \}$, the set of rounds when the index of the optimal arm underestimates its true value $\theta_1$. Fix the constant $\delta > 0$. Further, define $F_k = \{ n : k(n) = k, |\hat{\theta}_k(n) - \theta_k| \geq \delta \}$. Let $k \neq 1$ be a suboptimal arm and let $n \notin B \cup F_k$
such that \( k(n) = k \) and therefore, \( n \) is a good round. Then, since \( k(n) = k \), we must have \( b_k^1(n) \geq \theta_1^k(n) \geq \theta_1 \) and consequently:

\[
\log(n) + 3\log\log(n) = t_k(n)I(\hat{\theta}_k(n), b_k(n)) \geq t_k(n)I(\hat{\theta}_k(n), \theta_1) \geq t_k(n)I(\theta_k + \delta, \theta_1)
\]

Consequently:

\[
t_k(T) \leq \frac{\log(n) + 3\log\log(n)}{I(\theta_k + \delta, \theta_1)} + |B| + |F_k|.
\]

It remains to bound \( \mathbb{E}[|B|] \) and \( \mathbb{E}[|F_k|] \). To bound \( \mathbb{E}[|B|] \), we invoke Theorem 2.1.2 and hence:

\[
\mathbb{E}[|B|] = \mathbb{E}\left[ \sum_{n=1}^{T} \mathbb{1}[b_k^1(n) < \theta_1] \right] = \sum_{n=1}^{T} \mathbb{P}[b_k^1(n) < \theta_1] \leq \sum_{n=1}^{T} \frac{1}{n\log(n)^{3}} = O(1).
\]

To bound \( \mathbb{E}[|F_k|] \) we apply Lemma 6.3.1 with \( \Lambda = \{ n : k(n) = k \} \) and \( \epsilon = 1 \) immediately yields:

\[
\mathbb{E}[|F_k|] = \frac{1}{\delta^2}.
\]

Putting everything together, we have that there exists a constant \( C > 0 \):

\[
\mathbb{E}[t_k(T)] \leq \frac{\log(n) + 3\log\log(n)}{I(\theta_k + \delta, \theta_1)} + C + \delta^{-2}
\]

which proves the statement in the theorem.

\[\square\]

### 2.1.4 Thompson Sampling

Thompson Sampling (TS) was first proposed in 1933 by Thompson in [3]. However, the algorithm was only proven asymptotically optimal in 2012 by Kaufmann et al. in [10]. TS works by drawing, at each round \( n \), a belief \( \theta(n) \) from the posterior distribution of \( \theta \) given the observations up to time \( n \). It then plays the arm with the highest belief. Therefore, since we assume Bernoulli rewards, at time \( n \) we will sample the belief on arm \( k \) according to the distribution with the following probability density function:

\[
\mathbb{P}[^\lambda|X_1, \ldots, X_n] = \frac{\lambda^{n\hat{\theta}_k(n)} \times (1 - \lambda)^{n(1 - \hat{\theta}_k(n))}}{\alpha}
\]

where \( \alpha \) is a normalizing constant, depending on \( \hat{\theta}_k(n) \), meant to ensure that \( \sum_{\lambda \in \Theta} \mathbb{P}[^\lambda|X_1, \ldots, X_n] = 1 \). Note that this implies that the belief of each arm \( k \) is drawn from a Beta distribution of parameters \( t_k(n)\hat{\theta}_k(n) \) and \( t_k(n)(1 - \hat{\theta}_k(n)) \). This implies that the samples are more and more concentrated around the empirical mean as the number of plays of an arm increases (as seen in Figure 2.1) such that the regret scales logarithmically with the time horizon \( T \). More precisely, the optimality of TS in this setting is given by the following Theorem presented and proved in [10]:
2.2. Structured Bandits

Figure 2.1. Posterior distribution for an arm with empirical mean 0.5 and 6, 20 and 60 plays, respectively.

Theorem 2.1.4 For every $\epsilon > 0$ and for all $T \geq 1$, there exists a problem dependent constant $C > 0$ such that the regret of Thompson Sampling satisfies:

$$R(T) \leq (1 + \epsilon) \sum_{k=2}^{K} \frac{\theta_1 - \theta_k}{I(\theta_k, \theta_1)} (\log(T) + \log \log(T)) + C$$

While KL-UCB was proven optimal before TS, TS shows more promising numerical performance. Indeed, in [11], Guha and Munagala show that, in some specific bandit settings (when the prior on $\theta$ is known), the regret of TS at any time $t$ is at most 2 times larger than that of an optimal algorithm which is aware of $t$.

2.2 Structured Bandits

So far we have discussed settings where the rewards of arms are independent of one another. In this section we address bandit problems where knowledge of the reward of an arm offers information about the rewards of other arms. In other words, if we fix the reward of one arm, we limit the domain in which the expected rewards of other arms can lie. We will see some examples of structured bandit problems in Section 2.2.2. In the following section we can see a generic model for a wide variety of multi-armed bandit problems where arms are correlated. We then use the results of Graves and Lai in [4] to present a lower bound on the regret achievable by any uniformly good algorithm in these settings.
2.2.1 A Generic Model

We consider a stochastic multi-armed bandit problem with a finite set of arms $\mathcal{K} = \{1, 2, ..., K\}$ and a corresponding parameter vector $\theta = \{\theta_1, \theta_2, \ldots, \theta_K\}$ with $\theta \in \Theta$, $\Theta \subset (0, 1)^K$ being the set of all allowed parameters. Note that, unlike in the unstructured setting, $\Theta$ is a subset of $(0, 1)^K$. Pulls of an arm $k$ yield i.i.d. random values drawn from a Bernoulli distribution of mean $\theta_k$. We define a superarm as a set $S \subset \mathcal{K}$. At each round $n$ the decision maker is allowed to choose a superarm $S$ from a fixed set $\mathcal{S}$ and simultaneously play all arms in $S$ and collect the reward $X_S(n)$, where $X_S(n)$ is a random vector containing the observed rewards of the arms in $S$ at round $n$ and $\eta : \{0, 1\}^{|S|} \rightarrow \mathbb{R}^+$ is a known, fixed reward function. Note that after playing all arms in $S(n)$, the vector of observations $X_S(n)$ is revealed to the decision maker.

Denote $\mu_\theta(S) = \mathbb{E}[\eta(X_S)]$ where $X_S = \{X_i, i \in S\}$ with $X_i$ being a random variable drawn from the Bernoulli distribution of mean $\theta_i$. Define $\mu_\theta^* = \max_{S \in \mathcal{S}} \mu_\theta(S)$ and $S_\theta^* = \arg\max_{S \in \mathcal{S}} \mu_\theta(S)$. Throughout the rest of this thesis we will refer to $\mu_\theta^*$ and $S_\theta^*$ simply as $\mu^*$ and $S^*$. Let $\Pi$ denote the set of all sequential selection algorithms. The goal of the decision maker is to determine, without prior knowledge of the vector $\theta$, a sequential policy $\pi \in \Pi$ that minimizes the expected regret:

$$\mathbb{E}[R^\pi(T)] = T \mu_\theta^* - \mathbb{E} \left[ \sum_{n=1}^{T} \eta(X_{S^\pi(n)}(n)) \right]$$

2.2.2 Application to Specific Bandit Problems

Let us exemplify how our model can be applied for a few classes of bandit problems. In the applications studied below, unless stated otherwise, superarms only contain one arm so $S = \{\{k\}, k \in \mathcal{K}\}$, $\eta(X_S(n)) = ||X_S(n)||_1$ and the feedback of a trial of arm $k$ at time $n$ is a random variable $X(n)$ drawn from the Bernoulli distribution of mean $\theta_k$.

**Combinatorial Bandits:** In this instance we have $\Theta = (0, 1)^K$ and the decision maker can choose a superarm from the set $\mathcal{S}$. The feedback of a superarm $S = \{s_1, s_2, \ldots\} \subset \mathcal{K}$, chosen at round $n$ under parameter $\theta$, consists of a vector $X_S(n) \in \{0, 1\}^{|S|}$ of Bernoulli random variables of respective means $\theta_{s_1}, \theta_{s_2}, \ldots$. After observing the outcome of a pull, the algorithm collects the reward $\eta(X_S(n))$. Note that we do not concern ourselves with the scaling of regret with the cardinality of $\mathcal{S}$, as this is outside the scope of this thesis.

**Lipschitz Bandits:** Here, the problem is parametrized by the Lipschitz constant (assumed known) $L$. Each arm $k$ is mapped to a corresponding point $x_k \in [0, 1]$ (known and fixed over time) and the set of allowed parameters is defined as $\Theta = \{\theta \in (0, 1)^K : \forall k, k' \in \mathcal{K} \text{ such that } |\theta_k - \theta_{k'}| \leq L|x_k' - x_k|\}$.

**Linear Bandits:** In the $d$-dimensional Linear Bandit setting, for each arm $k \in \mathcal{K}$, there exists a corresponding vector $x_k \in [0, 1]^s$, $s \geq d$, $\dim(\text{span}\{x_k : k \in \mathcal{K}\}) = d$ and, for clarity, assume that for all $k$, $||x_k|| = l$, $l < 1$. The problem takes a vector $z \in [0, 1]^s$, $||z|| \in (0, 1)$ as input (which is unknown to the decision maker) and $\theta_k = <x_k, z>$, $\forall k \in \mathcal{K}$. The set of possible parameters is given by $\Theta = \{\lambda \in (0, 1)^K : \exists y \in [0, 1]^d : \forall k \in \mathcal{K} \text{ we have } <x_k, y> = \lambda_k\}$. 

2.2.3 Regret Lower Bounds

In this section, we derive an asymptotic (when $T$ grows large) regret lower bound satisfied by any algorithm $\pi \in \Pi$. Fix the parameter vector $\theta = (\theta_1, \ldots, \theta_K)$ and the set of playable superarms $S = \{S_1, S_2, \ldots\}$. Let $S^- = S \setminus \{S^\star\}$ be the set of suboptimal superarms. For all $S \in S^-$, we define

$$\Lambda^S(\theta) = \{\lambda \in \Theta : \mu_\lambda(S) = \mu^\star_{\theta} \text{ and } \forall k \in S^\star, I(\theta_k, \lambda_k) = 0\}.$$ 

Without loss of generality, we restrict our attention to uniformly good algorithms, as defined in [1]. $\pi \in \Pi$ is uniformly good if for all $\theta \in \Theta$, $R^\pi(T) = o(T^a)$ for all $a > 0$.

**Theorem 2.2.1** Let $\pi \in \Pi$ be a uniformly good algorithm. For any $\theta \in \Theta$, we have:

$$\liminf_{T \to \infty} \frac{R^\pi(T)}{\log(T)} \geq C(\theta),$$

where $C(\theta)$ is the minimal value of the following optimization problem:

$$\min_{c_S \geq 0, \forall S \in S^-} \sum_{S \in S^-} c_S \times (\mu^\star_{\theta} - \mu_{\theta}(S))$$

subject to

$$\forall S \in S^-, \inf_{\lambda \in \Lambda^S(\theta)} \sum_{k \in K} I(\theta_k, \lambda_k) \sum_{S' \in S} 1_{[k \in S']} c_{S'} \geq 1.$$  

The regret lower bound is a consequence of results in optimal control of Markov chains, see [4]. The proof of Theorem 2.2.1 is presented in the appendix. As in classical bandits, the minimal regret scales logarithmically with the time horizon. Note that for any $S \in S^-$, the variable $c_S$ corresponding to a solution of (4.3) characterizes the number of times superarm $S$ should be played under an optimal algorithm: superarm $S$ should be roughly played $c_S \log(n)$ times up to round $n$, when $n$ is large. It should be also observed that our lower bound is problem specific (it depends on $\theta$).
Chapter 3

Multi-armed Bandits with Lipschitz-Continuous Rewards

We have so far examined the classical MAB problem where the arms are independent of each other. Let us proceed to analyze the Lipschitz bandit problem. This chapter is based on the paper *Lipschitz Bandits: Regret Lower Bounds and Optimal Algorithms*, by S. Magureanu, R. Combes and A. Proutiere ([12]).

3.1 Problem Statement and Related Work

When the expected rewards of the various arms are not related as in Chapter 1, we have seen that the regret of the best algorithm scales as $O(K \log(T))$ where $K$ denotes the number of arms, and $T$ is the time horizon. When $K$ is very large or even infinite, MAB problems become more challenging. Fortunately, in such scenarios, the expected rewards often exhibit some structural properties that the decision maker can exploit to design efficient algorithms. Various structures have been investigated in the literature, e.g., Lipschitz [13], [14], [15], linear [16], and convex [17].

In this chapter, we revisit bandit problems where the expected reward is a Lipschitz function of the arm. The set of arms is a subset of $[0, 1]$ and we address both discrete Lipschitz bandits where this set is finite, and continuous Lipschitz bandits where this set is $[0,1]$. For discrete Lipschitz bandits, we derive problem specific regret lower bounds, and propose OSLB (Optimal Sampling for Lipschitz Bandits), an algorithm whose regret matches our lower bound. Most previous work on Lipschitz bandit problems address the case where the set of arms is the interval $[0, 1]$, see [13], [14], [15]. For these problems, there is no known problem specific regret lower bound. In [14], a regret lower bound is derived for the worst Lipschitz structure. The challenge in the design of efficient algorithms for continuous Lipschitz bandits stems from the facts that such algorithms should adaptively select a subset of arms to sample from, and based on the observed samples, establish tight confidence intervals and construct arm selection rules that optimally exploit the Lipschitz structure. The algorithms proposed in [13], [14], [15] adaptively define the set of arms to play, but used simplistic UCB indexes to sequentially select arms. In turn, these
algorithms fail at exploiting the problem structure revealed by the past observed samples. For continuous bandits, we propose to first discretize the set of arms (as in [14]), and then apply OSLB, an algorithm that optimally exploits past observations and hence the problem specific structure. As it turns out, this approach outperforms algorithms directly dealing with continuous sets of arms.

**Main contributions.**

(a) For discrete Lipschitz bandit problems, we derive an asymptotic regret lower bound satisfied by any algorithm. This bound is problem specific in the sense that it depends in an explicit manner on the expected rewards of the various arms (this contrasts with existing lower bounds for continuous Lipschitz bandits).

(b) We propose OSLB (Optimal Sampling for Lipschitz Bandits), an algorithm whose regret matches our lower bound. We further present POSLB (Pareto OSLB), an algorithm that exhibits lower computational complexity than that of OSLB, and that is yet able to exploit the Lipschitz structure.

(c) We provide a finite time analysis of the regret achieved under OSLB and POSLB. The analysis relies on a new concentration inequality for a weighted sum of KL divergences between the empirical distributions of rewards and their true distributions. We believe that this inequality can be instrumental for various bandit problems with structure.

(d) We evaluate our algorithms using numerical experiments for both discrete and continuous sets of arms. We compare their performance to that obtained using existing algorithms for continuous bandits.

(e) We extend our results and algorithms to the case of contextual bandits with similarities as investigated in [18].

### 3.2 Models

We consider a stochastic multi-armed bandit problem where the set of arms is a subset \( \{x_1, \ldots, x_K\} \) of the interval [0, 1]. Results can be easily extended to the case where the set of arms is a subset of a metric space as considered in [14]. The set of arms is of finite cardinality, possibly large, and we assume without loss of generality that \( x_1 < x_2 < \ldots < x_K \). Problems with continuous sets of arms are discussed in Section 3.6. Time proceeds in rounds indexed by \( n = 1, 2, \ldots \). At each round, the decision maker selects an arm, and observes the corresponding random reward. Arm \( x_k \) is referred to as arm \( k \) for simplicity. For any \( k \), the reward of arm \( k \) in round \( n \) is denoted by \( X_k(n) \), and the sequence of rewards \( (X_k(n))_{n \geq 1} \) is i.i.d. with Bernoulli distribution of mean \( \theta_k \) (the results can be generalized to distributions belonging to a certain parametrized family of distributions, but to simplify the presentation, we restrict our attention to Bernoulli rewards). The vector \( \theta = (\theta_1, \ldots, \theta_K) \) represents the expected rewards of the various arms. Let \( \mathcal{K} = \{1, \ldots, K\} \). We denote by \( \theta^* = \max_{k \in \mathcal{K}} \theta_k \) the expected reward of the best arm. A sequential selection algorithm \( \pi \) selects in round \( n \) an arm \( k^{\pi}(n) \in \mathcal{K} \) that depends on the past observations. In other words, for any \( n \geq 1 \), if \( \mathcal{F}_n^\pi \) denotes the \( \sigma \)-algebra generated by \((k^{\pi}(t), X_{k^{\pi}(t)}(t))_{1 \leq t \leq n}\), then \( k^{\pi}(n+1) \) is \( \mathcal{F}_n^\pi \)-measurable. Let \( \Pi \) denote the set of all
possible sequential selection algorithms.

We assume that the expected reward is a Lipschitz function of the arm, and this structure is known to the decision maker. More precisely, there exists a positive constant $L$ such that for all pairs of arms $(k, k') \in \mathcal{K}$,

$$|\theta_k - \theta_{k'}| \leq L \times |x_k - x_{k'}|. \tag{3.1}$$

We assume that $L$ is also known. We denote by $\Theta_L$ the set of vectors in $[0, 1]^K$ satisfying (3.1). The objective is to devise an algorithm $\pi \in \Pi$ that maximizes the average cumulative reward up to a certain round $T$ referred to as the time horizon ($T$ is typically large). Such an algorithm should optimally exploit the Lipschitz structure of the problem. As always in bandit optimization, it is convenient to quantify the performance of an algorithm $\pi \in \Pi$ through its expected regret (or regret for short) defined by:

$$R^\pi(T) = T\theta^* - E\left[\sum_{n=1}^{T} X_{k^\pi(n)}(n)\right].$$

### 3.3 Regret Lower Bound

In this section, we derive an asymptotic (when $T$ grows large) regret lower bound satisfied by any algorithm $\pi \in \Pi$. We denote by $I(x, y) = x \log \left( \frac{x}{y} \right) + (1-x) \log \left( \frac{1-x}{1-y} \right)$ the KL-divergence between two Bernoulli distributions with respective means $x$ and $y$. Fix the average reward vector $\theta = (\theta_1, \ldots, \theta_K)$. Let $\mathcal{K}^- = \{k \in \mathcal{K} : \theta_k < \theta^*\}$ be the set of sub-optimal arms. For any $k \in \mathcal{K}^-$, we define $\lambda_k = (\lambda_1, \ldots, \lambda_K)$ as: $\forall i \in \mathcal{K}$, $\lambda_i^k = \max\{\theta_i, \theta^* - L|x_k - x_i|\}$. The expected reward vector $\lambda_k$ is illustrated in Figure 3.1, and may be interpreted as the most confusing reward vector among vectors in $\Theta_L$ such that arm $k$ (which is sub-optimal under $\theta$) is optimal under $\lambda_k$. This interpretation will be made clear in the proof of the following theorem. Without loss of generality, we restrict our attention to so-called uniformly good algorithms, as defined in [1]. $\pi \in \Pi$ is uniformly good if for all $\theta \in \Theta_L$, $R^\pi(T) = o(T^a)$ for all $a > 0$. Uniformly good algorithms exist – for example, the UCB algorithm is uniformly good.

Theorem 3.3.1 Let $\pi \in \Pi$ be a uniformly good algorithm. For any $\theta \in \Theta_L$, we have:

$$\liminf_{T \to \infty} \frac{R^\pi(T)}{\log(T)} \geq C(\theta), \tag{3.2}$$

where $C(\theta)$ is the minimal value of the following optimization problem:

$$\min_{c_k \geq 0, \forall k \in \mathcal{K}^-} \sum_{k \in \mathcal{K}^-} c_k \times (\theta^* - \theta_k) \tag{3.3}$$

$$\text{s.t. } \forall k \in \mathcal{K}^-, \sum_{i \in \mathcal{K}} c_i I(\theta_i, \lambda_i^k) \geq 1. \tag{3.4}$$
The regret lower bound is a straightforward application of Theorem 2.2.1, when \( S = \mathcal{K}, \Theta = \Theta_L \) and \( \eta \) is the identity function. As in classical bandits, the minimal regret scales logarithmically with the time horizon. Observe that the lower bound (3.2) is smaller than the lower bound derived in [1] when the various average rewards \((\theta_k, k \in \mathcal{K})\) are not related (i.e., in absence of the Lipschitz structure). Hence (3.2) quantifies the gain one may expect by designing algorithms optimally exploiting the structure of the problem. Note that for any \( k \in \mathcal{K}^- \), the variable \( c_k \) corresponding to a solution of (3.3) characterizes the number of times arm \( k \) should be played under an optimal algorithm: arm \( k \) should be roughly played \( c_k \log(n) \) times up to round \( n \).

It should be also observed that our lower bound is problem specific (it depends on \( \theta \)), which contrasts with existing lower bounds for continuous Lipschitz bandits, see e.g. [14]. The latter are typically derived by selecting the problems that yield maximum regret. However, our lower bound is only valid for bandits with a finite set of arms, and cannot easily be generalized to problems with continuous sets of arms.

In the following section, we present two algorithms for discrete Lipschitz bandit problems. The first, referred to as OSLB (Optimal Sampling for Lipschitz Bandits), has a regret that matches the lower bound derived in Theorem 3.3.1, i.e., it is asymptotically optimal. OSLB requires that in each round, one solves an LP similar to (3.3). The second algorithm, POSLB (Pareto Optimal Sampling for Lipschitz Bandits) is much simpler to implement, but has weaker theoretical performance guarantees, although it provably exploits the Lipschitz structure.
3.4 Algorithms

In this section we will first present a novel index which we will subsequently use as the basic building block for two algorithms that efficiently exploit the Lipschitz structure. We will present the derivation of the index from the likelihood ratio test and present the algorithms and the intuition behind their use of new index.

3.4.1 Index of Arms

We have seen that using the Upper Confidence Bounds as an index of arms allows for the construction of an optimal algorithm. When arms are correlated however, this no longer holds and, moreover, Gittins’ result also breaks down. Since in these more complex settings samples observed from an arm can offer information about the rewards of other arms, it is no longer wise to build the index of an arm \( k \) based only on the observed rewards of \( k \). Particularly, in the case of Lipschitz continuous arms, we know that if the observed rewards of arms close to \( k \) are poor, with high probability, then the reward of \( k \) is also poor with high probability. Let use define \( f(n) = \log(n) + (3K + 1) \log \log(n) \). In order to exploit this structure, we use the index \( b_k(n) \) of arm \( k \) for round \( n \), defined by:

\[
b_k(n) = \sup\{ q \in [\hat{\theta}_k(n), 1] : \sum_{k' = 1}^K t_{k'}(n) I^+(\hat{\theta}_{k'}(n), \lambda_{q,k'}) \leq f(n) \}.
\]

Note that the index \( b_k(n) \) is always well defined, even for small values of \( n \), e.g. \( n = 1 \) (we have for all \( x > 0, I^+(0, x) = -\log(1 - x) \)). In order to understand the intuition behind the index let us note the following important fact: \( b_k(n) \) (when \( \hat{\theta}(n) \in \Theta_L \)) represents the lowest value \( q \) for which the likelihood ratio test rejects the hypothesis \( H_0 = \{ \theta \in \Theta_L : \theta_k > q \} \) with risk \( 1/(n \log(n)) \). Note that this means \( b_k(n) \) represents an Upper Confidence Bound index in the Lipschitz bandit setting. In contrast, KL-UCB does not exploit structure and the index used by the Zooming algorithm in [14] does not efficiently exploit neither the structure nor the information available from observed samples. In this regard, our index represents an important step towards constructing relevant indexes in more complex bandit settings.

To see why the index resembles a likelihood ratio test, let us assume \( \hat{\theta}(n) \in \Theta_L \) then the likelihood ratio test parametrized by \( \Lambda \), rejects \( H_0 = \{ \theta \in \Theta_L : \theta_k > q \} \) if:

\[
\max_{\lambda \in H_0} L(\hat{\theta}(n), \lambda) > \Lambda
\]

where \( L(\hat{\theta}(n), \lambda) \) represents the likelihood of observing \( \hat{\theta}(n) \) under \( \lambda \). The quantity \( \Lambda \) determines the probability of Type I errors of the test. Thus, with \( \lambda^k \) as defined in section 3.3:

\[
\frac{\max_{\lambda \in H_0} L(\hat{\theta}(n), \lambda)}{\max_{\lambda \in \Theta_L} L(\hat{\theta}(n), \lambda)} = \frac{L(\hat{\theta}(n), \lambda^k)}{L(\hat{\theta}(n), \hat{\theta}(n))} = \frac{\prod_{j \in \mathcal{K}} (\lambda_j^k t_j(n)\hat{\theta}_j(n) (1 - \lambda_j^k t_j(n)(1 - \hat{\theta}_k(n))))}{\prod_{j \in \mathcal{K}} (\hat{\theta}_j(n) t_j(n)(1 - \hat{\theta}_j(n))(1 - \theta_k(n)))}
\]
which, after taking the logarithm, further becomes:
\[
\log \frac{\max_{\lambda \in H_0} L(\hat{\theta}(n), \lambda)}{\max_{\lambda \in \Theta_L} L(\hat{\theta}(n), \lambda)} = \sum_{j \in \mathcal{K}} t_j(n) \left[ \hat{\theta}_j(n) \log \frac{\lambda^k_j}{\hat{\theta}_j(n)} + (1 - \hat{\theta}_j(n)) \log \frac{1 - \lambda^k_j}{1 - \hat{\theta}_j(n)} \right]
\]

Consequently, $H_0$ is rejected if:
\[
\sum_{j \in \mathcal{K}} t_j(n) I(\theta_j(n), \lambda^k_j) > \log(1/\Lambda)
\]

From the concentration inequality in Theorem 3.5.1 we obtain that, in order for the test to have risk of Type I errors of at most $1/(n \log(n))$ we must use $\Lambda = 1/(n \log(n)^{3K+1}$ and hence $H_0$ is rejected when:
\[
\sum_{j \in \mathcal{K}} t_j(n) I(\theta_j(n), \lambda^k_j) > \log(n) + (3K + 1) \log \log(n),
\]

which results in exactly the expression in the index $b_k(n)$.

An important observation is that, since in the classical setting we always have $\hat{\theta}(n) \in \Theta$, the index of KL-UCB is exactly the lowest value for which the likelihood ratio test of risk $1/(n \log(n)^3)$ rejects the hypothesis that $\theta_k > b_k(n)$. One can follow the above derivation to confirm this is indeed true in the classical setting. Furthermore, the index for the Contextual Lipschitz bandit setting follows the same principle.

### 3.4.2 The OSLB Algorithm

For any $n \geq 1$, let $k(n)$ be the arm selected under OSLB in round $n$. $t_k(n)$ denotes the number of times arm $k$ has been selected up to round $n - 1$. By convention, $t_k(1) = 0$. The empirical reward of arm $k$ at the end of round $(n - 1)$ is
\[
\hat{\theta}_k(n) = \frac{1}{t_k(n)} \sum_{t=1}^{n-1} \mathbb{1}\{k(t) = k\} X_k(t),
\]

if $t_k(n) > 0$ and $\hat{\theta}_k(n) = 0$ otherwise. We define the leader at round $n$ as $L(n) = \arg \max_{k \in \mathcal{K}} \hat{\theta}_k(n)$ the arm with the highest empirical reward (ties are broken arbitrarily) at the end of round $n - 1$. We also define $\hat{\theta}^*(n) = \hat{\theta}_{L(n)}(n)$ as the empirical reward of the leader at the end of round $n - 1$. Further define, for all $q \geq 0$ and $k$, the Lipschitz vector $\lambda^{q,k}$ such that for any $k'$, $\lambda^{q,k'}_k = q - L|x_k - x_{k'}|$. The sequential decisions made under OSLB are based on the indexes $b_k(n)$ of the various arms $k \in \mathcal{K}$. For any $\theta \in \Theta_L$, let $C(\theta)$ denote the minimal value of the optimization problem (3.3), and let $(c_k(\theta), k \in \mathcal{K}^-)$ be the values of the variables $(c_k, k \in \mathcal{K}^-)$ in (3.3) yielding $C(\theta)$. For simplicity, we define $\hat{C}(n) = C(\hat{\theta}(n))$, and $c_k(n) = c_k(\hat{\theta}(n))$ for any $k \in \mathcal{K}^-(n)$ where $\mathcal{K}^-(n) = \{k : \hat{\theta}_k(n) < \hat{\theta}^*(n)\}$. The design of OSLB stems from the observation that an optimal algorithm should satisfy $\lim_{n \to \infty} t_k(n)/(c_k(\theta) \log(n)) = 1$, almost surely, for all $k \in$
\section*{3.4. Algorithms}

**Algorithm 1 OSLB(ε)**

For all $n \geq 1$, select arm $k(n)$ such that:
- If $\hat{\theta}^*(n) \geq \max_{k \neq L(n)} b_k(n)$, then $k(n) = L(n)$;
- Else if $t_{\hat{k}(n)}(n) < \frac{\epsilon}{\bar{K}} t_{\hat{k}(n)}(n)$, then $k(n) = \hat{k}(n)$;
- Else $k(n) = \bar{k}(n)$.

$K^-$. Hence we should force the exploration of arm $k \in K^-(n)$ in round $n$ if $t_k(n) < \hat{c}_k(n) \log(n)$. We define the arm $\bar{k}(n)$ to explore as $\bar{k}(n) = \arg \min_{k \in K_e(n)} t_k(n)$ where $K_e(n) = \{k \in K^- : t_k(n) \leq \hat{c}_k(n) \log(n)\}$. If $K_e(n) = \emptyset$, $\bar{k}(n) = -1$ (a dummy arm). Finally we define the least played arm as $\hat{k}(n) = \arg \min_k t_k(n)$. In the definitions of $\bar{k}(n)$ and $\hat{k}(n)$, ties are broken arbitrarily. We are now ready to describe OSLB. Its pseudo-code is presented in Algorithm 1.

Under OSLB, the leader is selected if its empirical average exceeds the index of other arms. If this is not the case, OSLB selects the least played arm $\bar{k}(n)$, if the latter has not been played enough, and arm $\bar{k}(n)$ otherwise. Note that the description of OSLB is valid in the sense that $\bar{k}(n) \neq -1$ if $\hat{\theta}^*(n) < \max_{k \neq L(n)} b_k(n)$. After each round, all variables are updated, and in particular $\hat{c}_k(n)$ for any $k \in K^-(n)$, which means that at each round we solve an LP, similar to (3.3). In order to understand the intuition behind POSB, it is important to notice that the LP that is computed in each round can be translated as follows:

$$
\min_{t_k > 0, k \in K \setminus L(n)} \sum_{k \in K \setminus \{L(n)\}} t_k(\hat{\theta}_{L(n)}(n) - \hat{\theta}_k(n)) \text{ subject to }
\forall k \in K \setminus \{L(n)\}, b_k(n) < \hat{\theta}_{L(n)}(n).
$$

### 3.4.3 The POSLB Algorithm

The pseudocode can be found in Algorithm 2. We first define the index of the leader as the KL-UCB index:

$$
b_k^{kl}(n) = \sup \{q : t_k(n) I(\theta_k(n), q) < \log(n) + 3 \log \log(n)\}.
$$

When $n \to \infty$, in the case of the leader $k^*$, $b_k^{kl}(n) \to b_{k^*}(n)$ (see Lemma 6.4.2 in Appendix). Assigning this index to the leader eases the finite time analysis. The only difference between the indexes $b_k^{kl}(n)$ and $b_k(n)$ is that the former only considers the number of plays of $k$.

The POSLB (Pareto Optimal Sampling for Lipschitz Bandits) algorithm takes its name from the result stating that under POSLB would converge to a configuration of plays such that $\forall k : K$ we have $\lim_{n \to \infty} b_k(n) \to \theta^*$. Therefore, the number of plays of an arm $k^*$ cannot be reduced (in proportion to $\log(T)$) without increasing the number of plays of another. This result is presented in Theorem 3.5.4 and proved in the appendix.

The POSLB algorithm does not concern itself with finding the optimal solution but instead only aims at satisfying the constraints of the optimization problem in Theorem
Algorithm 2 POSLB

For all \( n \geq 1 \), select arm \( k(n) \) such that:
\[
q(n) = b_{L(n)}^{U}(n);
\]
\[
k(n) = \arg \max_{k} f(n) - f_k(n, q(n)) \text{ (ties are broken arbitrarily)}
\]
where
\[
f_k(n, q(n)) = \begin{cases} 
\sum_{j \in \mathcal{K}} t_j(n) I(\hat{\theta}_j(n), \lambda_{j,k,q}(n)) & \text{if } k \neq L(n) \\
t_k(n) I(\hat{\theta}_k(n), q(n)) & \text{if } k = L(n) 
\end{cases}
\]
and \( \lambda_{j,k,q}(n) = \max(q - |k - j|L, \hat{\theta}_j(n)) \).

3.3.1. To this end, apparently suboptimal arms are explored based on how "far" they are from satisfying their corresponding constraint in Equation (4.3). In intuition, this is roughly similar to playing the arm with the highest index \( b_k(n) \). The algorithm also follows the optimism in the face of uncertainty principle, since, as stated earlier, \( b_k(n) \) is an upper confidence bound for arm \( k \).

Note that the algorithm does not need to solve the optimization problem in Theorem 3.3.1 with every iteration, unlike OSLB, which might perform poorly when the solutions to the optimization problem in Theorem 3.3.1 are sensitive to noise in the parameter estimate. POSLB only computes the KL-UCB index for the leader, and only requires a simple comparison for each suboptimal arm. This makes it significantly faster and more robust than OSLB. Furthermore, in the classical setting (when the Lipschitz constant \( L = \infty \)), it is faster than KL-UCB.

3.5 Regret Analysis

In this section, we provide finite time upper bounds for the regret achieved under OSLB and POSLB.

3.5.1 Concentration Inequalities

To analyze the regret of algorithms for bandit optimization problems, one often has to leverage results related to the concentration-of-measure phenomenon. More precisely, here, in view of the definition of the indexes \( b_k(n) \), we need to establish a concentration inequality for a weighted sum of KL divergences between the empirical distributions of rewards and their true distributions. We derive such an inequality. The latter extends to the multi-dimensional case the concentration inequality derived in [8] for a single KL divergence. We believe that this inequality can be instrumental in the analysis of general structured bandit problems, as well as for statistical tests involving vectors whose components have distributions in a one-parameter exponential family (such as Bernoulli or Gaussian distributions). For simplicity, the inequality is stated for Bernoulli random variables only.

We use the following notations. For \( k \in \mathcal{K} \), \( \{X_k(n)\}_{n \in \mathbb{N}} \) is a sequence of i.i.d. Bernoulli random variables with expectation \( \theta_k \) and \( X(n) = (X_k(n), k \in \mathcal{K}) \). We rep-
resent the history up to round \( n \) using the \( \sigma \)-algebra \( \mathcal{F}_n = \sigma(X(1), \ldots, X(n)) \), and define the natural filtration \( \mathcal{F} = \{\mathcal{F}_n\}_{n \geq 1} \). We consider a generic sampling rule \( B(n) = (B_k(n), k \in K) \) where \( B_k(n) \in \{0, 1\} \) for all \( k \in K \). The sampling rule is assumed to be predictable in the sense that \( B(n) \in \mathcal{F}_{n-1} \).

We define the number of times that \( k \) was sampled up to round \( n - 1 \) by \( t_k(n) = \sum_{t=1}^{n-1} B_k(t) \) and the sum \( S_k(n) = \sum_{t=1}^{n-1} B_k(t) X_k(t) \). The empirical average for \( k \) is \( \bar{\theta}_k(n) = S_k(n)/t_k(n) \) if \( t_k(n) > 0 \) and \( \bar{\theta}_k(n) = 0 \) otherwise. Finally, we define the vectors \( \hat{\theta}(n) = (\hat{\theta}_1(n), \ldots, \hat{\theta}_K(n)) \) and \( t(n) = (t_1(n), \ldots, t_K(n)) \). When comparing vectors in \( \mathbb{R}^K \), we use the component-by-component order unless otherwise specified.

**Theorem 3.5.1** For all \( \delta \geq (K + 1) \) and \( n \in \mathbb{N} \) we have:

\[
\mathbb{P} \left[ \sum_{k=1}^{K} t_k(n) I^+ (\bar{\theta}_k(n), \theta_k) \geq \delta \right] \leq e^{-\delta} \left( \frac{[\delta \log(n)]^{\delta}}{K} \right)^K e^{K+1}. \tag{3.5}
\]

The proof of Theorem 3.5.1 can be found in [12] and involves tools that are classically used in the derivation of concentration inequalities, but also requires the use of stochastic ordering techniques, see e.g. [19].

### 3.5.2 Finite time analysis of OSLB

Next we provide a finite time analysis of the regret achieved under OSLB, under the following mild assumption. This assumption greatly simplifies the analysis.

**Assumption 1** The solution of the LP (3.3) is unique.

It should be observed that the set of parameters \( \theta \in \Theta_L \) such that Assumption 1 is satisfied constitutes a dense subset of \( \Theta_L \).

**Theorem 3.5.2** For all \( \epsilon > 0 \), under Assumption 1, the regret achieved under \( \pi = \text{OSLB}(\epsilon) \) satisfies: for all \( \theta \in \Theta_L \), for all \( \delta > 0 \) and \( T \geq 1 \),

\[
R^\pi(T) \leq C^\delta(\theta)(1 + \epsilon) \log(T) + C_1 \log \log(T) + K^3 \epsilon^{-1} \delta^{-2} + 3K \delta^{-2}, \tag{3.6}
\]

where \( C^\delta(\theta) \to C(\theta) \), as \( \delta \to 0^+ \), and \( C_1 > 0 \).

In view of the above theorem, when \( \epsilon \) is small enough, OSLB(\( \epsilon \)) approaches the fundamental performance limit derived in Theorem 3.3.1. More precisely, we have for all \( \epsilon > 0 \) and \( \delta > 0 \):

\[
\lim_{T \to \infty} \sup \frac{R^\pi(T)}{\log(T)} \leq C^\delta(\theta)(1 + \epsilon).
\]

In particular, for any \( \zeta > 0 \), one can find \( \epsilon > 0 \) and \( \delta > 0 \) such that \( C^\delta(\theta)(1 + \epsilon) \leq (1 + \zeta)C(\theta) \), and hence, under \( \pi = \text{OSLB}(\epsilon) \),

\[
\lim_{T \to \infty} \sup \frac{R^\pi(T)}{\log(T)} \leq C(\theta)(1 + \zeta).
\]
### 3.5.3 Regret Guarantees of POSLB

The following theorem shows that POSLB dominates KL-UCB (is at least as good in finite time and strictly better asymptotically), thus proving our algorithm exploits the Lipschitz structure of the problem.

**Theorem 3.5.3** Under POSLB, for all \( \theta \in \Theta_L, \) all \( T \geq 1, \) all \( 0 < \delta < (\theta^* - \max_{k \neq k^*} \theta_k)/2, \) and any suboptimal arm \( k \in K^{-}, \)

(i) we have:

\[
\mathbb{E}[t_k(T)] \leq \frac{f(T)}{I(\theta_k + \delta, \theta^* - \delta)} + C_1 \log(\log(T)) + 2\delta^{-2}.
\]

with \( C_1 \geq 0 \) a constant.

Furthermore:

**Theorem 3.5.4** Under POSLB, for all \( \theta \in \Theta_L \) and \( k \in K^{-}, \) we have that:

\[
\lim_{T \to \infty} \mathbb{E} \left[ \frac{\sum_{i \in K^{-}} t_i(T) I^+(\theta_i, \gamma_i^{\theta^*-k})}{f(T)} \right] = 1.
\]

Intuitively, Theorem 3.5.4 implies that \( b_k(n) \to \theta^*, \) for all \( k \) when \( n \to \infty. \) This result is somewhat surprising since it implies non-trivial properties of the coefficient matrix of the LP in Equation (3.3). Defining the matrix of KL-divergence numbers \( A = (a_{ik})_{i,k \in K} \) with \( a_{ik} = I^+(\theta_i, \lambda_{ik}^{\theta^*-k}), \) a direct consequence of the above theorem is that \( A \) is invertible and the linear system of equations \( Ax = 1 \) has strictly positive solutions. Furthermore:

**Corollary 3.5.1** Consider a square positive matrix \( A \in \mathbb{R}^{K \times K}_{+}. \) If there exists a set \( \{x_k \geq 0 : k \in \{1, \ldots, K\} \) and \( x_k < x_{k+1} \} \) such that \( \forall i \in \{1, \ldots, K\}, \forall j, k, l \in \mathbb{N} \) such that \( i \leq j < k < l \leq K \) or \( 0 < j < k < l \leq i \) we have \( \forall y \neq i, A(i, i) > A(i, y) \) and:

\[
\frac{A(i,j)(x_i - x_k) + A(i,l)(x_k - x_j)}{x_i - x_j} > A(i, k)
\]

then \( A \) is invertible and the solution to \( Ax = 1 \) has strictly positive components.

To prove Corollary 3.5.1, one must simply consider the proof of Theorem 3.5.4 in a deterministic setting and using \( I(\theta_j, \lambda_j^{\theta^*-k}) = A(j, k). \) Intuitively, the corollary states that if, for some square, positive matrix \( A, \) the values of a column decrease away from the diagonal as a strictly convex function of the row, then the matrix is invertible and the solution to the linear system of equations \( Ax = 1 \) has strictly positive components.

It is also important to notice that, in this setting, POSLB is Pareto optimal in the sense that we cannot decrease the number of plays of one arm without having to increase the number of plays of another. Also, note that the algorithm is suboptimal, as the solutions of the optimization problem in (3.3) do not always saturate all constraints. One can easily generate parameters \( \theta \in \Theta_L \) such that POSLB converges to something other than the optimal solution.
3.5.4 Numerical Evaluation

We first consider discrete bandit problems with 46 arms, and with time horizons less than $T = 5 \times 10^5$ rounds. The regret is averaged over 150 runs. In Figure 3.2, we compare the performance of KL-UCB and POSLB. For improved numerical performance, in the case of both algorithms we ignore the $\log \log(n)$ terms in the indexes (i.e. $f(n) = \log(n)$). On the left, we plot the expected reward as a function of the arm, as well as the (scaled) amount of times $E[t_k(n)]/\log(n)$ sub-optimal arm $k$ is played under both algorithms, as function of time. Under KL-UCB, the amount of times for arm $k$ approaches $1/I(\theta_k, \theta^*)$, whereas under POSLB, $E[t_k(n)]$ satisfy the upper bounds derived in Theorems 3.5.4 and 3.5.3. POSLB explores suboptimal arms less often than KL-UCB, as it is designed to exploit the Lipschitz structure. On the right, we plot the expected regret as a function of time under both algorithms. The regret under POSLB is always smaller than that under KL-UCB (the regret under KL-UCB is typically twice as large as that under POSLB in this example). This illustrates the significant gains that one may achieve by efficiently exploiting the structure of the problem.

3.6 Towards Optimal Continuous Lipschitz Bandits

We now turn our attention to continuous Lipschitz bandits where the set of arms is $[0,1]$. We postulate that running an algorithm designed for MABs with discrete arms using a sensible discretization of the action space will offer good performance in the continuous setting, since it will optimally exploit the statistical structure of the rewards. Note that using a uniform discretization of the set of arms, with step $\delta^{-1} = \lceil \sqrt{T/\log(T)} \rceil$ arms yields order-optimal regret scaling in the continuous setting for functions which are regular around their maximum [20]. Furthermore, algorithms such as POSLB and OSLB are proven to efficiently exploit the statistical structure of observed samples. This approach
Figure 3.3. Expected regret of different algorithms as function of time for a triangular reward function (left) and a quadratic reward function (right).

however is not without its problems. For example, when $T \to \infty$, arms become arbitrarily close and the index of an arm $k$, $b_k(n)$, shows discontinuities when we have noise in the estimates $\hat{\theta}_j(n)$, $j \in K$. Now if we consider each of the $t_k(n)$ to be a unique play of $t_k(n)$ arms instead of $t_k(n)$ plays of a unique arm, it is sensible that a good index should produce the same value $b_k(n)$ in both scenarios. Unfortunately, the index in Equation (3.5) does not exhibit this property.

It is worth noting that the regret of the algorithm depends on how close the true optimal arm is from the chosen discretization. It is also worth noting that, while discretizing uniformly guarantees correct scaling of regret, it is not necessarily optimal in terms of multiplicative constants. Ideally, an optimal algorithm should be able to play any point in the arm space, without being limited to the discretization. In other words, a good algorithm should be able to control the discretization. This point is proven by the fact that Zooming and HOO (which use adaptive discretizations) outperform KL-UCB using a uniform discretization. The comparison is fair since all three algorithms use indexes that depend on only the plays of a single arm, meaning their upper confidence intervals are of roughly the same tightness (with KL-UCB having a slight edge).

3.6.1 Numerical Evaluation

In our numerical simulations, we consider two reward functions that behave differently around their maximum: (1) $\theta(x) = 0.8 - 0.5|0.5 - x|$ (triangle) and (2) $\theta(x) = \max(0.1, 0.9 - 3.2 \times (0.7 - x)^2)$ (quadratic function). To adapt KL-UCB and POSLB to this continuous setting, we use a uniform discretization of the set of arms, with $\delta^{-1} = \lceil \sqrt{T/\log(T)} \rceil$ arms. As mentioned earlier, this discretization is known to be order-optimal for functions which are regular around their maximum [20]. In order not to give a positive bias to KL-UCB and POSLB, we make sure that the maximum of the reward functions is not achieved in one of the arms in the discretization: the maximum is placed at a distance of at least $\delta/4$ from any arm in the discretization. We compare the performance of KL-UCB and POSLB to that of the algorithm HOO introduced in [15], and the Zooming algorithm proposed in
[14]. The two latter algorithms have performance guarantees (they are order-optimal). We also compare KL-UCB and POSLB to HOO+ and Zooming+, two improved versions of HOO and Zooming, respectively. In these tuned versions, the confidence radius (see [15] and [14] for details) is set equal to $\sqrt{\log(n)/(2 \times t_k(n))}$ in round $n$. HOO+ and Zooming+ exhibit better performance than their initial versions, but their regrets have not been analytically studied. In the experiments, we limit the time horizon to $T = 25000$ rounds, and the expected regret is calculated by averaging over 100 independent runs.

Figure 3.3 presents the expected regret of the various algorithms for the triangular reward function (left) and for the quadratic reward function (right). First note that surprisingly, KL-UCB, an algorithm that does not leverage the Lipschitz structure, outperforms some of the algorithms designed to exploit the structure. This highlights the importance of having efficient indexes to the performance of the algorithm. Observe that POSLB clearly outperforms KL-UCB and all other algorithms in both problem instances. For quadratic reward functions, it is known that the optimal discretization of the set of arms should roughly have $(\log(T)/T)^{1/4}$ arms, [9]. We also plot the regret achieved under POSLB using this optimized discretization, and we observe that this indeed further reduces the regret.

It is worth noting that in the case of POSLB most of the regret is caused by not discretizing enough around the top arm. In contrast, in the case of Zooming and HOO, most of the regret is caused by loose confidence bounds. Therefore, in future work we will explore the possibility of combining the adaptive discretization scheme of Zooming and HOO with efficient confidence bounds as used by POSLB.

### 3.7 Contextual Bandit with Similarities

The algorithms and results presented above can be extended to the case of contextual bandit problems with similarities as studied in [18]. In such problems, in each round, the decision maker observes a context, and then decides which arm to select. The expected reward of the various arms depends on the context, and is assumed to be Lipschitz in the arm and context. We assume that contexts arrive according to an i.i.d. process whose distribution is not known to the decision maker. This contrasts with most of the work in contextual bandits, where the context process is adversarial.

#### 3.7.1 Model

Let $\{y_1, \ldots, y_J\}$ denote the set of possible contexts, assumed to be a subset of $[0, 1]$. We assume that $y_1 < \ldots < y_J$. For simplicity, context $y_j$ is referred to as context $j$. For each context $j \in \mathcal{J} = \{1, \ldots, J\}$, the expected rewards of the various arms are represented by a vector $\theta(j) = (\theta_k(j), k \in \mathcal{K})$ ($\theta_k(j)$ is the expected reward of arm $k$ when the context is $j$). We consider a general scenario where the reward is a Lipschitz function in both the arm and the context. There exists $L$ (known to the decision maker) such that for all $(i, k), (j, l) \in \mathcal{J} \times \mathcal{K}$,

$$|\theta_k(i) - \theta_l(j)| \leq L \times D((i, k), (j, l)), \quad (3.7)$$
Multi-armed Bandits with Lipschitz-Continuous Rewards

where \( D \) refers to some metric over \( J \times K \). The choice of this metric is free, and allows us to consider different scenarios. For example, we may assume that the Lipschitz structure is stronger in terms of arms than in terms of contexts. In this case, we may choose, for some \( \beta > 1 \),

\[
D((i, k), (j, l)) = \sqrt{(\beta(y_i - y_j)^2 + (x_k - x_l)^2)}.
\]

The set of parameters \( \theta = (\theta_k(j), k \in K, j \in J) \) satisfying (3.7) is denoted by \( \Theta_{L,2} \).

The context process is i.i.d.. The distribution of the observed context \( j(n) \) in round \( n \) is \( \psi \), i.e., \( \psi(j) = \mathbb{P}[j(n) = j] \). Without loss of generality, we assume that for any \( j \in J \), \( \psi(j) > 0 \). \( \psi \) is unknown to the decision maker. Let \( X_{j,k}(n) \) denote the reward of arm \( k \) obtained in round \( n \) when the context is \( j \). For contextual bandits, we define the regret of algorithm \( \pi \) as follows:

\[
R^\pi(T) = T \sum_{j \in J} \psi(j) \theta^*(j) - \sum_{n=1}^{T} \mathbb{E}[X_{j(n), k^*(n)}(n)].
\]

where \( \theta^*(j) \) denotes the reward of the best arm under context \( j \), and as earlier \( k^*(n) \) denotes the arm selected under \( \pi \) in round \( n \).

### 3.7.2 Regret Lower Bound

To state the regret lower bound, we introduce for any context \( j \in J, K^-(j) = \{k \in K : \theta_k(j) < \theta^*(j)\} \) the set of suboptimal arms for context \( j \). We also introduce for any context \( j \in J \), and any \( k \in K \), the vector \((\lambda^{j,k}_l(i), l \in K, i \in J)\) such that

\[
\lambda^{j,k}_l(i) = \max \{\theta_l(i), \theta^*(j) - L \sqrt{D((j, k), (i, l))}\}.
\]

**Theorem 3.7.1** Let \( \pi \) be a uniformly good algorithm. Then, for any \( \theta \in \Theta_{L,2} \):

\[
\liminf_{T \to \infty} \frac{R^\pi(T)}{\log(T)} \geq C'(\theta)
\]

where \( C'(\theta) \) is the minimal value of the following optimization problem:

\[
\min_{c_{j,k} \geq 0, \forall j, \forall k} \sum_{j \in J} \sum_{k \in K^-(j)} c_{j,k} \times (\theta^*(j) - \theta_k(j))
\]

\[
\text{s.t. } \forall j, \forall k \in K^-(j), \sum_{i \in J} \sum_{l \in K} c_{i,l} I(\theta_l(i), \lambda^{j,k}_l(i)) \geq 1.
\]

Observe that our regret lower bound is problem specific, and again the values of the \( c_{j,k} \)'s solving the above optimization problem can be interpreted as follows: an asymptotically optimal algorithm plays arm \( k \) when the context is \( j \) a number of times that scales as \( c_{j,k} \log(T) \) as \( T \) grows large. Also note that, remarkably, the regret lower bound does
3.7. Contextual Bandit with Similarities

Algorithm 3 CPOSB

For all \( n \geq 1 \), observe context \( j(n) \) and select arm \( k(n) \) such that:

\[
q(n) = b_{L(n,j)}^k(n);
\]
\[
k(n) = \arg \max_k f(n) - f_k(n, q(n)) \text{ (ties are broken arbitrarily)}.
\]

where

\[
f_{k,y}(n, q) = \begin{cases} 
\sum_{i \in I} \sum_{j \in K} t_j(i, n) I(\hat{\theta}_j(i, n), \lambda_{j}^{k,q,y}(i, n)) & \text{if } k \neq L(n) \\
t_k(n) I(\hat{\theta}_k(n), q(n)) & \text{if } k = L(n)
\end{cases}
\]

and

\[
\lambda_j^{k,q,y}(i, n) = \max(q - D((y, k), (i, j)), L, \hat{\theta}_j(i, n)).
\]

not depend on the distribution \( \psi \) of the contexts. While the distribution of contexts can increase (or decrease) the total reward of the oracle policy by increasing (or decreasing) the frequency of contexts with high reward, it has no effect on the regret lower bound. Furthermore, since the constraints for an arm \( y \) and context \( i \) represent the statistical test of the hypothesis \( H_0 = \{ \theta_k(i) > \theta^*(i) \} \), and this test only depends on the plays of all other arms (and not on the frequency with which contexts appear) it is sensible that the arrival process of context will not determine the number of plays required to optimally satisfy the statistical tests.

3.7.3 Algorithms

The algorithms proposed for Lipschitz bandits can be naturally extended to the case of contextual bandits with similarities. For conciseness, we just present CPOSLB (Contextual POSLB), the extension of POSLB. Its regret analysis can be conducted as that of POSLB with minor modifications.

To describe CPOSLB, we introduce the following notations. Let \( \hat{\theta}_k(j, n) \) denote the empirical average reward of arm \( k \) for context \( j \) up to round \( n - 1 \). \( t_k(j, n) \) is the number of times context \( j \) is presented and arm \( k \) is chosen up to round \( n - 1 \). We define the index \( b_{k}^c(j, n) \) of arm \( k \) for round \( n \), when the context \( j \) is observed as:

\[
b_{k}^c(j, n) = \sup \{ q \in [\hat{\theta}_k(j, n), 1] : \sum_{i \in J} \sum_{l \in K} t_l(i, n) I^+(\hat{\theta}_l(i, n), \lambda_l^{q,k,j}(i, n)) \leq f(n) \},
\]

where \( \lambda_l^{q,k,j}(i, n) = q - LD((j, k), (i, l)) \). As for Lipschitz bandits, the indexes are built so as to match the constraints (3.10) of the optimisation problem leading to the regret lower bound. The leader for round \( n \) and context \( j \) is defined \( L(n, j) = \arg \max_k \hat{\theta}_k(j, n) \) (ties are broken arbitrarily). In round \( n \), CPOSLB plays the leader \( L(n, j(n)) \) for the current context if it has the highest index, and otherwise selects the least played arm which has an index higher than the leader \( L(n, j(n)) \).
3.8 Conclusions

In this chapter we addressed the problem of multi-armed bandits where the mean rewards are a Lipschitz function of the arm. In this section we introduce a new index, generalizing that of KL-UCB to settings with dependent arms and two asymptotically (pareto) optimal algorithms OSLB and POSLB. We show that in this setting, exploiting structure offers a substantial decrease in regret. Further, we extend our results to the infinite-armed and contextual settings. While this structure is fairly weak, it represents a stepping stone for exploiting more complex and useful correlations, such as the case of Linear Bandits.
In this chapter we present a variation of the MAB problem where the amount of feedback is stochastic. In the classical setting, the feedback consists of exactly one Bernoulli random variable. In more complex problems, like the combinatorial bandits with semi-bandit feedback, the decision maker selects sets of arms and observes one random variable for each. In this setting, the decision maker must select an ordered set of arms, but will observe only a random number of rewards. The laws dictating the feedback in this model will be explained later, in Section 4.2. This chapter is based on the paper *Learning to Rank: Regret Lower Bounds and Efficient Algorithms* [21].

### 4.1 Problem Introduction

In this chapter, we address the problem of learning to rank a set of items based on user feedback. Specifically, we consider a service, where users repeatedly issue queries (e.g. a text string). There are $N$ items, and given the query, a decision maker picks an ordered subset or list of items of size $L$ to be presented to the user. The user examines the items of the list in order, and clicks on the first item she is interested in. The goal for the decision maker is to maximize the number of clicks (over a fixed time horizon, i.e., for a fixed number of queries), and to present the most relevant items in the first slots or positions in the list. The probability for a user to click on an item is unknown to the decision maker initially, and must be learned in an on-line manner, through trial and error. The problem of learning to rank is fundamental in the design of several online services such as search engines [22], ad-display systems [23] and video-on-demand services where the presented items are webpages, ads, and movies, respectively.

#### 4.1.1 Motivation

The main challenge in the design of learning-to-rank algorithms stems from the prohibitively high number of possible decisions: there are $N!/(N - L)!$ possible lists and typically, we
may have more than 1000 items and 10 slots. Hence even trying each decision once can be too costly and inefficient. Fortunately, when selecting a list of items, the decision maker may leverage useful side-information about both the user and her query. For instance, in search engines, the decision maker may be aware of her gender, age, location, etc., and could also infer from her query the type of documents she is interested in (i.e., the topic of her query), which in turn may significantly prune the set of items to choose from. Formally, we assume that the set of items can be categorized into $K$ different disjoint groups, each group corresponding to a given topic. Similarly, users are clustered into $K$ classes, such that a class-$k$ user is interested in items in group $h(k)$ only ($h$ is a 1-to-1 mapping from the user classes to the groups of items). This structure simplifies the problem. Two main issues remain however. 1) Even though we could know the class $k$ of the user issuing the query as well as the group of items of interest $h(k)$, we still need to select in that group the $L$ most relevant items. 2) The topic of the query could remain unclear. For example, the query "jaguar" in a search engine may correspond to several topics, for instance a car manufacturer, an animal or a petascale supercomputer. In this case, it seems appropriate to select items from several groups to make sure that at least one item in the list is relevant. This feature is referred to as diversity principle in the literature [24]. Another important and final feature of the problem stems from the nature of the decisions: The reward, e.g. the probability that there exists a relevant item in the displayed list, typically exhibits diminishing returns, e.g., it can be a submodular function of the set of displayed items [25].

Our MAB problem differs from the classical bandit framework in several ways. First, the type of feedback received by the system depends on the actual relevance of the various items in the displayed list. For example, if the user clicks on the last item, we know that none of the previous items in the list are relevant. Conversely, if the user clicks on the first item, we do not get any feedback for the subsequent items in the list. Then, the rewards of two lists containing a common item are not independent.

There has recently been an important effort to tackle structured MAB problems similar to ours, refer to Section 4.1.2 for a survey of existing results. The design of previously proposed learning-to-rank algorithms has been based on heuristics, and these algorithms seem like reasonable solutions to the problem. In contrast, here, our aim is to devise algorithms with provably minimum regret. Our contributions are as follows:

(i) We first investigate the case where the topic of the user query is known. We derive problem-specific regret lower bounds satisfied by any algorithm. We also propose PIE (Parsimonious Item Exploration), an algorithm whose regret matches our lower bound, and scales as $O(N_k \log(T))$ when applied to queries of class-$k$ users. Here $N_k$ denotes the number of items in the group $h(k)$, and $T$ denotes the time horizon, i.e., the number of queries. The exploration of apparently suboptimal items under PIE is parsimonious, as these items are explored only in a single position of the list.

(ii) We then handle the case where the class of the user issuing the query is known, but the group of items she is interested in is not (i.e., the mapping $h$ is unknown). For this scenario, we propose PIE-C (where "C" stands for "Clustered"), an algorithm that efficiently learns the mapping $h$, and in turn, exhibits the same regret guarantees as PIE, i.e., as if the mapping $h$ was known initially. In fact, we establish that learning the topic of interest for each user class incurs a constant regret (i.e., that does not scale with the time
(iii) Finally, we illustrate the performance of PIE and PIE-C using numerical experiments. To this aim, we use both artificially generated data and real-world data extracted from the MovieLens dataset. In all cases, our algorithms outperform existing algorithms.

4.1.2 Related work

Learning to rank relevant contents has attracted a lot of attention in recent years with an increasing trend of modeling the problem as a MAB with semi-bandit feedback. Most of existing models for search engines do not introduce a sufficiently strong structure to allow for the design of efficient algorithms. For example, in [24, 26, 25, 27, 28], the authors hardly impose any structure in their model. Indeed, they consider scenarios where the random variables representing the relevance of the various items are arbitrarily correlated, and even sometimes depart from the stochastic setting by considering adversarial item relevances. The only important structure that these work consider relates to the diminishing return property, and they typically assume that the reward is just a submodular function of the subset of displayed items, see e.g. [25]. As a consequence, the regret guarantees that can be achieved in the corresponding MAB problems (e.g. submodular bandit problems) are weak; a regret with sublinear scaling in the time horizon cannot be achieved. For instance, in submodular bandits, and its variants, the regret has to be defined by considering, as a benchmark, the performance of the best offline polynomial-time algorithm whose approximation ratio is $1 - 1/e$ [29] unless $NP \subset DTIME(n^{\log \log(n)})$, which indeed implies that the true regret scales linearly with time. In absence of strong structure, one cannot hope to develop algorithms that learn to rank items in a reasonable time. We believe that our model by its additional and natural clustered structure is more appropriate, and in turn, allows us to devise efficient algorithms, i.e., algorithms whose regret scales as $\log(T)$ as the time horizon $T$ grows large.

In [30], Kholi et al. present an analysis close to ours. There, each user is represented by a binary vector in $\{0, 1\}^N$ indicating the relevance of the various items to that user, and users are assumed to arrive according to an i.i.d. process with unknown distribution $D$. They first assume that the relevances of the different items are independent, similar to our setting, and propose a UCB1-based algorithm whose regret provably scales as $O(NL \log(T))$. UCB1 is unfortunately suboptimal, and as we show in this chapter, one may devise algorithms with regret scaling as $O(N \log(T))$ in this setting. Then, to extend their results to more general distributions $D$ (allowing for arbitrary correlations among items), the authors of [30] leverage a recent and elegant result from [31] to establish that a regret guarantee scaling as $(1 - 1/e)T$.

In [32], Slivkins et al. investigate a scenario where items are represented by vectors in a metric space, and assume that their relevance probabilities are Lipschitz continuous. While this model captures the positive correlation between similar items, it does not account for negative correlations between topics. For example, if a user issues the query "jaguar", and if she is not interested in cars, it means that most likely her query concerns the animal.

There has been over the last decade an important research effort towards the understanding of structured MAB problems, see [33] for a recent survey. By structure, we mean
that the reward as a function of the arm has some specific properties. Various structures have been investigated in the literature, e.g., Lipschitz [13, 14, 15, 12], linear [16], convex [17]. The structure of the MAB problem corresponding to the design of learning-to-rank algorithms is different, and to our knowledge, we present here the first solution (regret lower bounds, and asymptotically optimal algorithms) to this problem.

4.2 System Model

4.2.1 Users, Items, and Side-information

Our model captures the two important properties of online services mentioned in the introduction, namely the diversity principle and the diminishing return property. Let \( \mathcal{N} = \{1, \ldots, N\} \) be a set of items (news, articles, files, etc.). Time proceeds in rounds. In each round, a user makes a query and in response to this query, the decision maker has to select from \( \mathcal{N} \) an ordered list of \( L \) items. We denote by \( \mathcal{U} = \{u \subset \mathcal{N} : u = \{u_1, \ldots, u_L\}, u_i \in \mathcal{N}, u_i \neq u_j \text{ if } i \neq j\} \) the set of all possible decisions. The user scans the selected list in order, and stops as soon as she identifies a relevant item. In round \( n \), the relevance of items to the user is captured by a random vector \( X(n) = (X_i(n), i \in \mathcal{N}) \in \{0, 1\}^N \), where for any item \( i \), \( X_i(n) = 1 \) if and only if it is relevant.

Item / User classification. We assume that the set \( \mathcal{N} \) is partitioned into \( K \) disjoint groups \( \mathcal{N}_1, \ldots, \mathcal{N}_K \) of respective cardinalities \( N_1, \ldots, N_K \). For example, in the case of a query "jaguar", we could consider three groups, corresponding to items related to the animal, the car brand, or a super-computer. This partition of the various items corresponds to the possible broad topics of user queries. Similarly, we categorize users into \( K \) different classes, and denote by \( h(k) \) the index of the topic of interest for class-\( k \) users, i.e., the query of class-\( k \) users concern items in \( \mathcal{N}_{h(k)} \). The mapping \( h \) could be known or not as discussed below. Denote by \( k(n) \) the class of the user making the query in round \( n \). \( (k(n), n \geq 1) \) are i.i.d. random variables with distribution \( \phi = (\phi_1, \ldots, \phi_K) \) where \( \phi_k = \mathbb{P}[k(n) = k] > 0 \). Now, given \( k(n) = k \), \( (X_i(n), i \in \mathcal{N}) \) are independent. Let \( \theta_{ki} = \mathbb{P}[X_i(n) = 1|k(n) = k] \) denote the probability that item \( i \) is relevant to class-\( k \) users. As already noticed in [28], the above independence assumption captures the diminishing return property. Indeed, given \( k(n) = k \), if \( u \) is the set of displayed items, the probability that the user finds at least one relevant item in \( u \) is \( 1 - \prod_{i=1}^{L}(1 - \theta_{ku_i}) \), which is a submodular function of \( u \) (hence with diminishing return).

Observe that the set of users is not specified in our model. We assume that there is an infinite pool of users, that the class of the user issuing a query in round \( n \) is drawn from distribution \( \phi \), and that this user does not issue any query in subsequent rounds. In particular, we cannot learn the class of users from observations. This contrasts with the model proposed in [34], where the set of users is finite, and hence the decision maker can learn the classes of the various users when they repeatedly place queries.

Diversity principle. To capture the diversity principle in our model, we assume that when a user makes a query, she is interested in a single topic only, i.e., in items within a
4.2. System Model

A single group $\mathcal{N}_{h(k)}$ only. More precisely, we assume that for all $k, \ell \in [K] := \{1, \ldots, K\}$:

$$\max_{i \in \mathcal{N}_\ell} \theta_{ki} \begin{cases} < \delta & \text{if } \ell \neq h(k), \\ > \Delta & \text{if } \ell = h(k), \end{cases}$$

(4.1)

for some fixed $0 < \delta < \Delta < 1$. Typically, we assume that there is an item that is highly relevant to users of a given class, so that e.g., $\Delta > 1/2$. When in round $n$, the topic $h(k(n))$ is not known, items of various types should be explored and displayed in the $L$ slots so that the chance of displaying a relevant item is maximized. In other words, (4.1) captures the diversity principle.

**Side-information and Feedback.** In round $n$, under decision rule $\pi$, an ordered list of $L$ items is displayed. This decision depends on past observations and some side information, i.e., in round $n$, the decision rule $\pi$ maps $((u^\pi(s), f(s), i(s), s < n), i(n))$ to a decision in $\mathcal{U}$, where $u^\pi(s)$, $f(s)$, and $i(s)$ denote the list selected under $\pi$, the received feedback, and the side information in round $s$, respectively.

**Feedback:** In round $n$, if the ordered list $u = (u_1, \ldots, u_L)$ is displayed, the decision maker is informed about the first relevant item in the list, i.e., $f(n) = \min\{i \leq L : X_{u_i}(n) = 1\}$. By convention, $f(n) = 0$ if none of the displayed items is relevant. This type of feedback is often referred to as semi-bandit feedback in the bandit literature.

**Side-information:** we model different types of systems depending on the information available to the decision maker about the user placing the query. For example, when a user issues a query, one could infer from her age, gender, location, and other attributes the topic of her query. In such a case, the decision maker knows, before displaying the list of items, the topic of the query, i.e., in round $n$, $i(n) = h(k(n))$. Alternatively, the decision maker could know the user class (which could be extracted from users’ interactions in a social network) but not the topic of her query, i.e., $i(n) = k(n)$. In this case, the mapping $h$ remains unknown.

### 4.2.2 Rewards and Regret

To formulate our objectives, we specify the reward of the system when presenting a given ordered list, and introduce the notion of regret which we will aim at minimizing.

The reward is assumed to be a decreasing function of the position of the first relevant item, e.g., in search engines, it is preferable to display the most relevant item first. The reward function is denoted by $r(\cdot)$, i.e., the reward is $r(\ell)$ where $\ell$ denotes the position of the first relevant item in the list. In absence of relevant item, no reward is collected. Without loss of generality, we assume that rewards are in $[0, 1]$.

In view of our assumptions, the expected reward when presenting an ordered list $u$ to a class-$k$ user is:

$$\mu_\theta(u, k) := \sum_{l=1}^{L} r(l) \theta_{ku_l} \prod_{i=1}^{l-1} (1 - \theta_{ku_i}),$$

where $\theta := (\theta_{ki}, k \in [K], i \in \mathcal{N})$ captures the statistical properties of the system.

The performance of a decision rule $\pi$ is characterized through the notion of regret which compares the performance of an Oracle algorithm aware of the parameter $\theta$ to that
Learning to Rank using Multi-Armed Bandits

of the decision rule $\pi$ up to a given time horizon $T$ (in rounds). The way regret is defined depends on the available side-information. To simplify the presentation and the results, we make the two following assumptions:

(A1) In each group of items, there are at least $L$ items that are relevant with a probability greater than $\delta$. In particular, $N_k \geq L$ for all $k \in [K]$.

(A2) The number of groups $K$ is larger than the number of slots $L$.

Under these two assumptions, the performance of an Oracle algorithm can be expressed in a simple way. It depends however on the available side-information.

Known topic: When in each round $n$, the topic $h(k(n))$ is known, the best decision in this round consists in displaying the $L$ most relevant items of group $N_{h(k(n))}$. For any user class $k$, and $\ell = h(k)$, for all $i \in [N]$, $i_\ell$ denotes the item in $N_\ell$ with the $i$-th highest relevance: $\theta_{k1_\ell} \geq \theta_{k2_\ell} \geq \ldots \geq \theta_{kN_\ell}$. The list maximizing the expected reward given that the user class is $k(n) = k$ is $u^{*,k} := (1_{h(k)}, \ldots, L_{h(k)})$. Thus, the expected reward under the Oracle algorithm is:

$$\mu_{1,\theta}^* := \sum_{k \in [K]} \phi_k \mu_\theta(u^{*,k}, k),$$

and the regret under algorithm $\pi$ up to round $T$ is defined as:

$$R_\theta^\pi(T) := T \mu_{1,\theta}^* - \mathbb{E} \left[ \sum_{n=1}^T \mu_\theta(u^\pi(n), k(n)) \right].$$

To simplify the presentation, we assume that given a user class $k$, the optimal list is unique, i.e., for any $u \neq u^{*,k}$, $\mu_\theta(u, k) < \mu_\theta(u^{*,k}, k)$.

Known user class, and unknown topic: In this case, since the Oracle algorithm is aware of the parameter $\theta$, it is also aware of the mapping $h$. Thus, the regret of algorithm $\pi$ is the same as in the previous case, i.e., up to time $T$, the regret is $R_\theta^\pi(T)$.

Our objective is to devise efficient sequential list selection algorithms in both scenarios, when the topic of successive queries are known, and when only the class of the user issuing the query is known.

4.3 A Single Group of Items and Users

In this section, we study the case where $K = 1$, i.e., there is a single class of user and a single group of item. Even with $K = 1$, our bandit problem remains challenging, due the non-linear reward structure and the reward-specific feedback. To design efficient algorithms, we need to determine how many and where apparently sub-optimal items should be included for exploration in the displayed list.

When $K = 1$, we can drop the indexes $k$ and $h(k)$. To simplify the notation, we replace $\theta_{k_i h(k)}$ by $\theta_i$ for all $i \in [N]$. More precisely, we have $N$ items, and users are statistically identical, i.e., $\theta_i$ denotes the probability that item $i$ is relevant. Let $\theta = (\theta_1, \ldots, \theta_N)$ and w.l.o.g. the items are ordered so that $\theta_1 \geq \theta_2 \geq \ldots \geq \theta_N$. We denote by $u^* = (1, \ldots, L)$ the list with maximum expected reward, $\mu_\theta^*$. The regret of policy $\pi$ up to round $T$ is then

$$R_\theta^\pi(T) = T \mu_\theta^* - \mathbb{E}[\sum_{n=1}^T \mu_\theta(u^\pi(n))],$$

where $\mu_\theta(u)$ is the expected reward of list $u$. 
4.3.1 Regret Lower Bound

We first derive a generic regret lower bound valid for any decreasing reward function \( r(\cdot) \). This lower bound will be made more explicit for particular choices of reward functions. We define uniformly good algorithms as in [1]. A uniformly good algorithm \( \pi \) satisfies \( R^\pi_\theta(T) = o(T^a) \) for all parameters \( \theta \) and all \( a > 0 \). Later, it will become clear that such algorithms exist, and therefore we only restrict our attention to the set of such algorithms.

We denote by \( I(a, b) \) the Kullback-Leibler divergence between two Bernoulli distributions of respective means \( a \) and \( b \), i.e., \( I(a, b) = a \log(a/b) + (1 - a) \log((1 - a)/(1 - b)) \). We further define \( U(i) = \{ u \in U : i \in u \} \), the set of lists in \( U \) that include the item with the \( i \)-th highest relevance. Finally, for any list \( u \), and any item \( i \in u \), we denote by \( p_i(u) \) the position of \( i \) in \( u \).

**Theorem 4.3.1** For any uniformly good algorithm \( \pi \), we have:

\[
\lim_{T \to \infty} \inf \frac{R^\pi_\theta(T)}{\log(T)} \geq c(\theta), \tag{4.2}
\]

where \( c(\theta) \) is the minimal value of the objective function in the following optimization problem \((P_\theta)\):

\[
\inf_{c_u \geq 0, u \in U} \sum_{u \in U} c_u (\mu^*_\theta - \mu_\theta(u)) \tag{4.3}
\]

s.t. \( \sum_{u \in U(i)} c_u I(\theta_i, \theta_L) \prod_{s < p_i(u)} (1 - \theta_u_s) \geq 1, \forall i > L. \)

The solution of the optimization problem \((P_\theta)\) has a natural interpretation. For any \( u \in U \), \( c_u \) represents the expected number of times the list \( u \) should be displayed using an algorithm minimizing the regret. More precisely, \( u \) should be displayed \( c_u \log(T) \) times asymptotically when the time horizon \( T \) grows large. Theorem 4.3.1 and its above interpretation are applications of the theory of controlled Markov chains with unknown transition kernel developed in [4]. Next we specify the solution of \((P_\theta)\) for two particular classes of reward functions. Define for \( i < L \), \( \Delta_i = r(i) - r(i + 1) \), and \( \Delta_L = r(L) \).

1) **Reward functions such that**: \( \Delta_i \geq \Delta_L > 0 \) for all \( i < L \). This assumption on the reward function seems natural in the context of search engines where the rewards obtained from items presented first are high and rapidly decrease as the position of the item increases.

**Proposition 4.3.1** Assume that \( \Delta_i \geq \Delta_L > 0 \) for \( i < L \). Then for all \( u \in U \) such that \( u \neq u^* \), the coefficient \( c_u \) corresponding to the solution of \((P_\theta)\) satisfies: If for some \( i > L \), \( u = (1, \ldots, L - 1, i) \),

\[
c_u = \frac{1}{I(\theta_i, \theta_L) \prod_{j<L}(1 - \theta_j)},
\]
else $c_u = 0$. Hence, we have:

$$c(\theta) = \Delta_L \sum_{i=L+1}^{N} \frac{\theta_L - \theta_i}{I(\theta_i, \theta_L)}.$$ 

The above proposition states that very few lists from $\mathcal{U}_k$ should be explored $\Theta(\log(T))$ times. These lists include the $(L - 1)$ most relevant items in the $(L - 1)$ first slots, and an item that is not within the $L$ most relevant items in the last slot. In other words, an optimal algorithm should include only one sub-optimal item in the list when it explores, and this item should be placed last. This observation will simplify the design of asymptotically optimal algorithms – although of course, initially, the decision maker does not know the $(L - 1)$ most relevant items. Note that the minimum regret scales as $(N - L) \log(T)$; this indicates that optimal algorithms should really exploit the reward and feedback structures.

2) Reward functions such that: $\Delta_i = 0$ for all $i < L$, and $\Delta_L > 0$. This scenario may be appropriate in the case of display ads, where the reward obtained when a user clicks does not depend on the position of the ad on the webpage.

**Proposition 4.3.2** Assume that $\Delta_i = 0$ for all $i < L$, and $\Delta_L > 0$. Then for all $u \in \mathcal{U}$ such that $u \neq u^*$, the coefficient $c_u$ corresponding to the solution of $(P_\theta)$ satisfies:

If for some $i > L$, $u = (i, 1, \ldots, L - 1)$,

$$c_u = \frac{1}{I(\theta_i, \theta_L)},$$

Else $c_u = 0$. Hence, we have:

$$c(\theta) = \Delta_L \prod_{j < L} (1 - \theta_j) \sum_{i=L+1}^{N} \frac{(\theta_L - \theta_i)}{I(\theta_i, \theta_L)}.$$ 

Again the above proposition states that very few lists should be explored $\Theta(\log(T))$ times. These lists are those containing the $(L - 1)$ most relevant items in the $(L - 1)$ last positions, and an item that is not within the $L$ most relevant items in the first position. In other words, the exploration of items is performed in the first position. Observe that as in the previous case, the minimum regret scales as $(N - L) \log(T)$.

3) General Reward Function. An explicit expression for the lower bound $c(\theta)$ for general reward function is more challenging to derive. However, we suspect that the lists $u$ such that $c_u > 0$ are $w_i^l = (1, \ldots, (l - 1), i, l, \ldots, (L - 1))$ for some $i > L$. In other words, only one suboptimal item, (i.e., for $i > L$, the $i$-th most relevant item) is explored at a time, and we should explore $i$ in the $l$-th position. To determine this position, we make the following heuristic reasoning. Let us fix the number of times $i$ is explored. Given this fixed exploration rate, we select position $l$ that induces the smallest regret. Let us assume that $i$ is explored in slot $l$. When the list $w_i^l$ is displayed, the probability $p_l$ that $i$ is actually explored is: $p_l = \prod_{j=1}^{l-1}(1 - \theta_j)$. The average number of times $i$ is actually explored when
placed in position \( l \) is proportional to \( 1/p_l \). Hence the position \( \text{oes}(i) \) where \( i \) should be placed should satisfy:
\[
\text{oes}(i) \in \text{argmin}_{i \leq L} f(\theta, w^i_l),
\]
where \( f(\theta, w^i_l) = \frac{\mu^i_l - \mu(\theta)}{p_l} \). Let \( w^i = w^i_{\text{oes}(i)} \). If the argmin in (4.4) is realized for different positions, we break ties arbitrarily. We state the following conjecture.

For any decreasing reward function \( r(\cdot) \), and for \( u \in \mathcal{U} \), the coefficient \( c_u \) in the solution of (4.4) has the following form:
\[
c_u = \begin{cases} 
\frac{1}{T(\theta, \theta_L)p_{\text{oes}(i)}} & \text{if } u = w^i \text{ for some } i > L, \\
0 & \text{otherwise}.
\end{cases}
\]

Observe that the conjecture holds in Cases 1) and 2). In the former, it is optimal to place any suboptimal item \( i \) (\( i > L \)) in the last slot, in which case \( p_{\text{oes}(i)} = \prod_{j=1}^{L-1} (1 - \theta_j) \). In the latter, it is optimal to place any suboptimal item \( i \) in the first slot, in which case \( p_{\text{oes}(i)} = 1 \).

### 4.3.2 Optimal Algorithms

Next we present asymptotically optimal sequential list selection algorithms, i.e., their limiting regret (as \( T \) grows large) matches the lower bound derived above. To describe our algorithms, we need to introduce the following definitions. Let \( u(n) \) be the list selected in round \( n \), and let \( p_i(n) \) denote the position at which item \( i \) is shown if \( i \in u(n) \), and \( p_i(n) = 0 \) otherwise. Recall that a sample of \( \theta_i \) is obtained if and only if \( i \in u(n) \) and \( X_{i'}(n) = 0 \) for all \( i' \in \{u_1(n), \ldots, u_{p_i(n)-1}\} \). Define
\[
o_i(n) := \mathbb{I}\{i \in u(n), \forall l' < p_i(n), X_{u_{i'}(n)}(n) = 0\}.
\]

Then we get a sample from \( \theta_i \) in round \( n \) iff \( o_i(n) = 1 \). Let \( t_i(n) := \sum_{n' \leq n} o_i(n') \) be the number of samples obtained for \( \theta_i \) up to round \( n \). The corresponding empirical mean is:
\[
\hat{\theta}_i(n) = \frac{1}{t_i(n)} \sum_{n' \leq n} o_i(n')X_i(n')
\]
if \( t_i(n) > 0 \) and \( \hat{\theta}_i(n) = 0 \) otherwise. We also define \( c_i(n) \) as the number of times that a list containing \( i \) has been selected up to round \( n \): \( c_i(n) := \sum_{n' \leq n} \mathbb{I}\{i \in u(n')\} \). Let \( j(n) = (j_1(n), \ldots, j_N(n)) \) be the indices of the items with empirical means sorted in decreasing order, so that:
\[
\hat{\theta}_{j_1(n)}(n) \geq \hat{\theta}_{j_2(n)}(n) \geq \ldots \geq \hat{\theta}_{j_N(n)}(n)
\]
We assume that ties are broken arbitrarily. Define the list of \( L \) "leaders" at time \( n \) as \( \mathcal{L}(n) = (j_1(n), \ldots, j_L(n)) \). The algorithms we propose use the indexes used by the KL-UCB algorithm, known to be optimal in classical MAB problems [2]. The KL-UCB index \( b_i(n) \) of item \( i \) in round \( n \) is:
\[
b_i(n) = \max\{q \in [0, 1] : t_i(n)I(\hat{\theta}_i(n), q) \leq f(n)\},
\]
Algorithm PIE(l)

Init: $B(1) = \emptyset$, $\hat{\theta}_i(1) = 0 = b_i(1)$, $\forall i$, $L(1) = \{1, \ldots, L\}$

For $n \geq 1$

If $B(n) = \emptyset$, select $L(n)$

Else w.p. 1/2, select $L(n)$, w.p. select $U^l_1(n)$, $I \in B(n)$ unif. distributed

Compute: $B(n + 1)$, $L(n + 1)$, and $\hat{\theta}_i(n + 1)$, $b_i(n + 1)$, $\forall i$


where $f(n) = \log(n) + 4 \log(\log(n))$. Let:

$B(n) := \{i \notin L(n) : b_i(n) \geq \hat{\theta}_j(n(n))\}$.

be the set of items which are not in the set of leaders, and whose index are larger than the empirical mean of item $j_L(n)$. Intuitively, $B(n)$ includes items which are potentially better than the worst current leader. For $1 \leq i \leq N$, define decision:

$$U^l_i(n) = (j_1(n), \ldots, j_{l-1}(n), i, j_l(n), \ldots, j_{L-1}(n)).$$

$U^l_i(n)$ is the list obtained by considering the $L - 1$ first items of $L(n)$, and by placing item $i$ at position $l$. We are now ready to present our algorithms. The latter, referred to as PIE(l), are parametrized by $l \in \{1, \ldots, L\}$, the position where the exploration is performed. In round $n$, PIE(l) proceeds as follows:

(i) if $B(n)$ is empty, then the leader is selected: $u(n) = L(n)$;

(ii) otherwise, we select $u(n) = L(n)$ with probability 1/2, and $u(n) = U^l_i(n(n))$ with probability 1/2, where $i(n)$ is chosen from $B(n)$ uniformly at random.

Refer to pseudo-code of PIE(l) for a formal description. Note that the PIE(l) algorithm has low computational complexity. It can be easily checked that it requires at each round $O(N + L \log(N))$ operations. In the following theorem, we provide a finite-time regret upper bound of the PIE(l) algorithm. Introduce $\eta = \prod_{i=1}^{L-1}(1 - \theta_i)^{-1}$ and recall that $p_t = \prod_{i' < (1 - \theta_i')}$ and $u^{i,l} = (1, \ldots, (l-1), i, (l+1), \ldots, (L-1))$ for all $i > l$.

**Theorem 4.3.2** Under algorithm $\pi = \text{PIE}(l)$, for all $T \geq 1$ and, all $\epsilon > 0$ and all $0 < \delta < \delta_0 = \min_{i < N}(\theta_i - \theta_{i+1})/2$, the regret under $\pi$ satisifies:

$$R^\pi(T) = f(T)c_{\text{PIE}(l)}(\theta, \delta) + C(\theta, \delta, \epsilon),$$

where

$$c_{\text{PIE}(l)}(\theta, \delta) = p_t^{-1} \sum_{i = L+1}^{N} \frac{\mu(u^*) - \mu(u^{i,l})}{I(\theta_i + \delta, \theta_L - \delta)},$$

$$C(\theta, \delta, \epsilon) = 2N\eta[(5 + 8NL)\eta + (3 + 2L)\delta^{-2}] + 15L
+(N - L)p_t^{-1} [\epsilon^{-2}p_t^{-1} + \delta^{-2}(1 - \epsilon)^{-1}].$$
As a consequence:

\[
\limsup_{T \to \infty} \frac{R^\pi(T)}{\log(T)} \leq C^{\text{PIE}(L)}(\theta) := \sum_{i=L+1}^{N} \frac{\mu(u^*) - \mu(u^*_{i,l})}{\prod_{i' < i}(1 - \theta_{i'}) I(\theta_{i}, \theta_{L})}.
\]

A direct consequence of the above theorem is that PIE(L) and PIE(1) are asymptotically optimal in Case 1) (convex decreasing reward functions) and Case 2) (constant rewards), respectively. Indeed, one can easily check that for example \(C^{\text{PIE}(L)}(\theta) = C(\theta)\) in Case 1).

### 4.3.3 Proofs: Lower Bounds

**Proof of Theorem 4.3.1**

The result is a consequence of the theory of controlled Markov chain with unknown transition rates [4]. We apply the formalism of [4] as follows. The state space of the Markov chain is \(X = \{0, 1, \ldots, L\}\), and the state will capture the feedback obtained from the previous decision, i.e., \(x = 0\) means that no item in the list is relevant, and \(x = i\) means that the first relevant item is in position \(i\). The set of control actions is the set of lists \(U\). The transition probability from state \(x\) to state \(y\) given that the chosen list is \(u\) is

\[
p(x, y; u, \theta) = \begin{cases} 
\prod_{s=1}^{L} (1 - \theta_{u,s}) & \text{if } y = 0, \\
\theta_{u,y} \prod_{s=1}^{y-1} (1 - \theta_{u,s}) & \text{if } y \in \{1, \ldots, L\}.
\end{cases}
\]

The reward associated to the state \(x\) and the control action \(u\) is denoted by \(g(x, u)\), and here we have \(g(x, u) = r(x)\). Finally, the set of control laws is \(G = U\). The expected reward under the control law \(u\) is \(\mu(\theta)\). Next we apply Theorem 1 in [4]. To this aim, we first introduce the KL divergence between two parameters \(\lambda \in [0, 1]^N\) and \(\theta\) under control law \(u\) as:

\[
I^u(\lambda, \theta) = \sum_{s=1}^{L} \theta_{u,s} \left[ \prod_{i=1}^{s-1} (1 - \theta_{u,i}) \right] \log \left( \frac{\theta_{u,s} \prod_{i=1}^{s-1} (1 - \theta_{u,i})}{\lambda_{u,s} \prod_{i=1}^{s-1} (1 - \lambda_{u,i})} \right) + \left[ \prod_{i=1}^{s} (1 - \theta_{u,i}) \right] \log \left( \frac{\prod_{i=1}^{s} (1 - \theta_{u,i})}{\prod_{i=1}^{s} (1 - \lambda_{u,i})} \right),
\]

which can be rewritten as:

\[
I^u(\theta, \lambda) = \sum_{i=1}^{L} I(\theta_{u,i}, \lambda_{u,i}) \prod_{s=1}^{i-1} (1 - \theta_{u,s}).
\]

Let us further introduce the set of bad parameters \(B(\theta)\) as:

\[
B(\theta) = \{ \lambda \in [0, 1]^N : I^u(\theta, \lambda) = 0 \text{ and } \exists u \neq u^*, \mu(\lambda(\mu_{u^*}) > \mu_{u^*}) \},
\]
where $\mu_\lambda(u)$ denotes the expected reward of decision $u$ under parameter $\lambda$. By definition, if $\lambda \in B(\theta)$, there is $i > L$ such that $\lambda_i > \theta_L$. Thus we can decompose $B(\theta)$ into the union of sets $B_i(\theta) = \{ \lambda \in B(\theta), \lambda_i > \theta_L \}$ over $i \in \{L + 1, \ldots, N\}$. By Theorem 1 in [4], we have, for any uniformly good algorithm $B$

we have, for any uniformly good algorithm $B$

where

$$c(\theta) = \inf_{c_u \geq 0, u \in U} \sum_{u \neq u^*} c_u (\mu_u^* - \mu_\theta(u))$$

$$\text{s. t. } \forall i > L, \inf_{\lambda \in B_i(\theta)} \sum_{u \neq u^*} c_u I_u(\theta, \lambda) \geq 1.$$ 

By definition of $B_i(\theta)$, if $\lambda \in B_i(\theta)$, then $\lambda_i > \theta_L$. It can easily be seen that $\inf_{\lambda \in B_i(\theta)} I_u(\theta, \lambda)$ is achieved for some parameter $\lambda^*$ such that $\lambda^*_i = \theta_L$ and $\lambda^*_j = \theta_j$ for $j \neq i$ and hence:

$$\inf_{\lambda \in B_i(\theta)} I_u(\theta, \lambda) = \sum_{u \in U(i)} c_u \prod_{s < p_i(u)} (1 - \theta_{u_s}) I(\theta_i, \theta_L) \geq 1.$$ 

This completes the proof. \qed

**Proof of Proposition 4.3.1**

For $i > L$, we define $\nu^i$ the list such that $\nu^i_j = j$ for $j < l$, and $\nu^i_L = i$. According to Proposition 4.3.1, these lists only should be explored under an optimal algorithm. Let $c = \{c_u : u \neq u^*\}$ be a solution of the LP introduced in Theorem 4.3.1. We prove by contradiction that $c_u > 0$ implies that there exists $i > l$ such that $u = \nu^i$. Assume $\exists u \neq u^*$ such that $c_u > 0$ and $u \neq \nu^i$, $\forall i > L$. We propose a new set of coefficients $c' = \{c'_u : u \neq u^*\}$ such that the value of objective function $c'(\theta)$ of the LP under $c'$ is less than the value of objective function $c(\theta)$ of the LP under $c$. We use the following notation: $c_{w,i} = c_w \prod_{s < p_i(w)} (1 - \theta_{w_s}) \prod_{s < L \theta_{w_s}}$ for any $w \in U$. Recall that $p_i(w)$ is the position of $i$ in $w$. Now introduce $c'$ such that for all $u \neq u^*$:

$$c'_w = \begin{cases} 0 & \text{if } w = u, \\ c_w + c_{u,p_i(u)} & \text{if } \exists i > L \text{ such that } w = \nu^i, \\ c_w & \text{otherwise.} \end{cases}$$

We show that $c'$ yields a strictly lower value of the objective function in the LP of Theorem 4.3.1 than $c$, a contradiction. Denote by $c(\theta)$ and $c'(\theta)$ the value of the objective function of the LP under $c$ and $c'$, respectively. We have:

$$c(\theta) - c'(\theta) = c_u (\mu_u^* - \mu_\theta(u)) - \sum_{i: u^*_i > L} c_{u,i} (\mu_u^* - \mu_\theta(\nu^i_u)).$$
4.3. A Single Group of Items and Users

It is easy to check that: \( \mu_{\theta}(u) = r(1) - \sum_{l=1}^{L} \Delta_l \prod_{s \leq l}(1 - \theta_s) \). Therefore \( \mu_{\theta}^* - \mu_{\theta}(u) = \sum_{l=1}^{L} \Delta_l(\prod_{s \leq l}(1 - \theta_s) - \prod_{s \leq l}(1 - \theta_s)) \). Since \( \Delta_L = r(L) \), we have:

\[
\mu_{\theta}^* - \mu_{\theta}(u) = \sum_{i: \mu_{\theta}(u) > L} \prod_{s \leq i}(1 - \theta_s) - \prod_{s \leq i}(1 - \theta_s) \]

Let \( i \leq L \) such that \( u_i > L \). We have:

\[
\Delta_i \prod_{s \leq i}(1 - \theta_s) - \prod_{s \leq i}(1 - \theta_s) - \Delta_L \prod_{s < i}(1 - \theta_s)(\theta_L - \theta_u)
\]

\[ \geq \Delta_i \prod_{s < i}(1 - \theta_u)(1 - \theta_L) - \prod_{s \leq i}(1 - \theta_s) > 0. \]

We deduce:

\[
c_u(\mu_{\theta}^* - \mu_{\theta}(u)) > \sum_{u_i > L} c_{u,i}(\mu_{\theta}^* - \mu_{\theta}(u)).
\]

And hence, \( c'(\theta) < c(\theta) \). We have shown that in \( c \) the solution of the LP involved in Theorem 4.3.1, \( c_u > 0 \) iff \( \exists i > l: u = v^i \). Now we can easily solve the LP in light of this result, and show that the \( c_u \)'s are of the form as stated in Proposition 1. The proof of Proposition 4.3.2 is similar.

4.3.4 Proof: Regret Upper bound for PIE(l)

Preliminaries

Before analyzing the regret of PIE(l), we state and prove Lemma 4.3.1. The latter shows that, under algorithm PIE(l), the set of rounds at which either (i) the set of leaders is different from the optimal decision, or (ii) the empirical mean of one of the leaders deviates from its expectation by more than a fixed quantity \( \delta > 0 \),

has finite size (in expectation). Note that (i) and (ii) are not mutually exclusive. The upper bound provided by Lemma 4.3.1 is explicit as a function of the parameters \( (\theta_i) \) and \( \delta \).

**Lemma 4.3.1** Define \( \delta_0 = \min_{i<N}(\theta_i - \theta_{i+1})/2 \) and \( \eta = \prod_{i=1}^{L-1}(1 - \theta_i)^{-1} \). Let \( 0 < \delta < \delta_0 \) and define the following sets of rounds:

\[
A = \{ n \geq 1 : \mathcal{L}(n) \neq u^* \},
\]

\[
D = \{ n \geq 1 : \exists i \in \mathcal{L}(n) : |\hat{\theta}_i(n) - \theta_i| \geq \delta \}.
\]
and \( C = A \cup D \). Under algorithm PIE(\( l \)), for all \( 0 < \delta < \delta_0 \) we have:

\[
\mathbb{E}[|C|] \leq 2N\eta[(5 + 8NL)\eta + (3 + 2L)\delta^{-2}] + 15L.
\]

**Proof.** Fix \( \delta < \delta_0 \) throughout the proof. Our goal is to upper bound the expected size of \( C \). To do so, we decompose \( C \) in an appropriate manner. We introduce the following sets of instants:

\[
\mathcal{E} = \{ n \geq 1 : \exists i \in \{1, ..., L \} : b_i(n) \leq \theta_i \}
\]
\[
\mathcal{G} = \{ n \geq 1 : n \in A \setminus (D \cup \mathcal{E}) , \exists i \in \{1, ..., L \} \setminus L(n) : |\hat{\theta}_i(n) - \theta_i| \geq \delta \}.
\]

We first check that \( C \subseteq D \cup \mathcal{E} \cup \mathcal{G} \). Since \( C = A \cup D \), it is sufficient to prove that \( A \subseteq (D \cup \mathcal{E} \cup \mathcal{G}) \). Let \( n \in A \setminus (D \cup \mathcal{E}) \). Let \( i, i' \in \mathcal{L}(n) \), with \( i < i' \). Since \( n \notin D \) we have \( |\hat{\theta}_i(n) - \theta_i| \leq \delta, |\hat{\theta}_{i'}(n) - \theta_{i'}| \leq \delta \), and \( \delta \leq (\theta_i - \theta_{i'})/2 \), therefore \( \hat{\theta}_i(n) \geq \hat{\theta}_{i'}(n) \). This proves that \( (j_1(n), ..., j_L(n)) \) is an increasing sequence. We have that \( j_L(n) > L \), otherwise we have \( (j_1(n), ..., j_L(n)) = (1, 2, ..., L) \) hence \( \mathcal{L}(n) = u^* \) and \( n \notin A \), a contradiction. Since \( j_L(n) > L \) there exists \( i \leq L \) such that \( i \notin \mathcal{L}(n) \). Let us now prove by contradiction that \( |\hat{\theta}_i(n) - \theta_i| \geq \delta \). Assume that \( |\hat{\theta}_i(n) - \theta_i| \leq \delta \), then we have \( |\hat{\theta}_{j_L(n)}(n) - \theta_{j_L(n)}| \leq \delta \) (since \( j_L(n) \in \mathcal{L}(n) \) and \( n \notin D \)) so that \( \hat{\theta}_i(n) \geq \hat{\theta}_{j_L(n)}(n) \). In turn this would imply that \( i \notin \mathcal{L}(n) \) which is a contradiction. Finally we have proven that \( n \in A \setminus (D \cup \mathcal{E}) \) implies \( n \in \mathcal{G} \). Hence \( C \subseteq D \cup \mathcal{E} \cup \mathcal{G} \), and by a union bound:

\[
\mathbb{E}[|C|] \leq \mathbb{E}[|D|] + \mathbb{E}[|\mathcal{E}|] + \mathbb{E}[|\mathcal{G}|].
\]

Next we prove the following inequalities:

(a) \( \mathbb{E}[|D|] \leq 2N\eta \left[ 10\eta + 3\delta^{-2} \right] \);

(b) \( \mathbb{E}[|\mathcal{E}|] \leq 15L \);

(c) \( \mathbb{E}[|\mathcal{G}|] \leq 4NL\eta \left[ 4\eta + \delta^{-2} \right] \).

**Inequality (a):** We further decompose \( D \) as \( D = \bigcup_{i=1}^{N} (D_{i,1} \cup D_{i,2}) \), with:

\[
D_{i,1} = \{ n \geq 1 : i \in \mathcal{L}(n), j_L(n) \neq i, |\hat{\theta}_i(n) - \theta_i| \geq \delta \}
\]
\[
D_{i,2} = \{ n \geq 1 : i \in \mathcal{L}(n), j_L(n) = i, |\hat{\theta}_i(n) - \theta_i| \geq \delta \}
\]

In other words, \( D_{i,1} \) is the set of rounds at which \( i \) is not the \( L \)-th leader, so that if \( n \in D_{i,1} \) then \( i \) will be included in \( u(n) \). \( D_{i,2} \) is the set of instants at which \( i \) is the \( L \)-th leader, so that if \( n \in D_{i,2} \), then either \( i \) or \( i(n) \) will be included in \( u(n) \).

First let \( n \in D_{i,1} \). Then we have \( i \in u(n) \) by definition of the algorithm. Hence \( \mathbb{E}[o_i(n)|n \in D_{i,1}] \geq \eta^{-1} \). Furthermore, for all \( n, 1 \{ n \in D_{i,1} \} \in \mathcal{F}_{n-1} \) measurable \( (\mathcal{F}_{n-1} \sigma \text{-algebra generated by } u(s) \text{ and the corresponding feedback for } s \leq n - 1) \). Therefore we can apply the second statement of Lemma 6.6.3, presented in Appendix (with \( H := D_{i,1}, c := \eta^{-1} \)) to obtain: \( \mathbb{E}[|D_{i,1}|] \leq 2\eta \left[ 2\eta + \delta^{-2} \right] \).

Next let \( n \in D_{i,2} \). Then we have that \( i \in u(n) \) with probability at least \( 1/2 \) by definition of the algorithm, so that \( \mathbb{E}[o_i(n)|n \in D_{i,2}] \geq \eta^{-1}/2 \). Also \( 1 \{ n \in D_{i,2} \} \) is
4.3. A Single Group of Items and Users

\( \mathcal{F}_{n-1} \) measurable. Hence applying the second statement of Lemma 6.6.3 (with \( H \equiv D_{i,2} \) and \( c \equiv n^{-1}/2 \)) we obtain: 
\[ \mathbb{E}[|D_{i,2}|] \leq 4n \left[ 4n + \delta^{-2} \right]. \]

Applying a union bound over \( 1 \leq i \leq N \), we get:
\[ \mathbb{E}[|D|] \leq \sum_{i=1}^{N} \mathbb{E}[|D_{i,1}|] + \mathbb{E}[|D_{i,2}|] \leq 2Nn \left[ 10n + 3\delta^{-2} \right]. \]

**Inequality (b):** Decompose \( \mathcal{E} \) as \( \mathcal{E} = \bigcup_{i=1}^{L} \mathcal{E}_i \) where
\[ \mathcal{E}_i = \{ n \geq 1 : b_i(n) \leq \theta_i \}. \]
Applying Lemma 6.6.4 we obtain that \( \mathbb{E}[|\mathcal{E}_i|] \leq 15 \) for all \( i \), so that:
\[ \mathbb{E}[|\mathcal{E}|] \leq \sum_{i=1}^{L} \mathbb{E}[|\mathcal{E}_i|] \leq 15L. \]

**Inequality (c):** Decompose \( \mathcal{G} \) as \( \mathcal{G} = \bigcup_{i=1}^{L} \mathcal{G}_i \) where
\[ \mathcal{G}_i = \{ n \geq 1 : n \in A \setminus (D \cup \mathcal{E}), i \not\in \mathcal{L}(n), |\hat{\theta}_i(n) - \theta_i| \geq \delta \}. \]
For a given \( i \leq L \), \( \mathcal{G}_i \) is the set of rounds at which \( i \) is not one of the leaders, and is not accurately estimated. Let \( n \not\in \mathcal{G}_i \). Since \( i \not\in \mathcal{L}(n) \), we must have \( j_L(n) > L \). In turn, since \( n \not\in \mathcal{D} \) we have \( |\hat{\theta}_{j_L(n)}(n) - \theta_{j_L(n)}| \leq \delta \), so that
\[ \hat{\theta}_{j_L(n)}(n) \leq \theta_{j_L(n)} + \delta \leq \theta_{L+1} + \delta \leq (\theta_{L+1} + \theta_L)/2. \]
Furthermore, since \( n \not\in \mathcal{E} \) and \( 1 \leq i \leq L \), we have \( b_i(n) \geq \theta_i \geq \theta_L \geq (\theta_{L+1} + \theta_L)/2 \geq \hat{\theta}_{j_L(n)}(n) \). This implies that \( i \in \mathcal{B}(n) \). Since \( i(n) \) has uniform distribution over \( \mathcal{B}(n) \), we have that \( \bar{i}(n) = i \) with probability at least \( 1/N \). We have that for all \( n \), \( \mathbb{E}[\bar{i}(n) \in \mathcal{G}_i] \geq \eta^{-1}/(2N) \). So we can apply Lemma 6.6.3 (with \( H \equiv \mathcal{G}_i \) and \( c \equiv n^{-1}/(2N) \)) to yield:
\[ \mathbb{E}[|\mathcal{G}_i|] \leq 4Nn \left[ 4n + \delta^{-2} \right]. \]

Using a union bound over \( 1 \leq i \leq L \), we obtain:
\[ \mathbb{E}[|\mathcal{G}|] \leq \sum_{i=1}^{L} \mathbb{E}[|\mathcal{G}_i|] \leq 4NLn \left[ 4Nn + \delta^{-2} \right]. \]

Putting inequalities (a), (b) and (c) together, we obtain the announced result:
\[ \mathbb{E}[|C|] \leq \mathbb{E}[|D|] + \mathbb{E}[|\mathcal{E}|] + \mathbb{E}[|\mathcal{G}|] \leq 2Nn \left[ (5 + 8NL)n + (3 + 2L)\delta^{-2} \right] + 15L, \]
which concludes the proof.
Proof of Theorem 4.3.2

We decompose the regret by distinguishing rounds in \( C \) (as defined in the statement of Lemma 4.3.1), and other rounds. For all \( i > L \), we define the sets of instants between 1 and \( T \) at which \( n \not\in C \) and decision \( u_i \) is selected (recall that \( u_i = (1, \ldots, (l - 1), i, (l + 1), \ldots, (L - 1)) \)):

\[
K_i = \{1 \leq n \leq T : n \not\in C, \mathcal{L}(n) = u^*, u(n) = u_i\}.
\]

By design of the algorithm, when \( n \not\in C \), the leader is the optimal decision, and so the only sub-optimal decisions that can be selected are \( \{u_{L+1}, \ldots, u_N\} \). Hence the set of instants at which a suboptimal decision is selected verifies:

\[
\{1 \leq n \leq T : u(n) \neq u^*\} \subset C \cup (\cup_{i=L+1}^N K_i).
\]

Since \( \mu(u^*) - \mu(u) \leq 1 \) for all \( u \), we obtain the upper bound:

\[
R^*(T) \leq \mathbb{E}[|C|] + \sum_{i=L+1}^N [\mu(u^*) - \mu(u_i)] \mathbb{E}[|K_i|].
\]

By Lemma 4.3.1, we have:

\[
\mathbb{E}[|C|] \leq 2N\eta[(5 + 8NL)\eta + (3 + 2L)\delta^{-2}] + 15L.
\]

Hence, to complete the proof, it is sufficient to prove that, for all \( i \geq L + 1 \), all \( \epsilon > 0 \) and all \( 0 < \delta < \theta_L - \theta_{L+1} \), we have:

\[
\mathbb{E}(|K_i|) \leq p_i^{-1} \frac{f(T)}{(1 - \epsilon)I(\hat{\theta}_i + \delta, \theta_L - \delta)} + p_i^{-1} \left[p_i^{-1} \epsilon^{-2} + \delta^{-2}(1 - \epsilon)^{-1}\right]. \tag{4.5}
\]

Define the number of rounds in \( K_i \) before round \( n \):

\[
k_i(n) = \sum_{n' \leq n} I\{n' \in K_i\}.
\]

Fix \( \epsilon > 0 \), define \( t_0 = f(T)/I(\hat{\theta}_i + \delta, \theta_L - \delta) \), and define the following subsets of \( K_i \):

\[
K_{i,1} = \{n \in K_i : t_i(n) \leq p_i(1 - \epsilon)k_i(n) \text{ or } |\hat{\theta}_i(n) - \theta_i| \geq \delta\},
\]

\[
K_{i,2} = \{n \in K_i : t_0 \leq p_i(1 - \epsilon)k_i(n)\}.
\]

Namely, \( K_{i,1} \) is the set of rounds in \( K_i \) where either item \( i \) has been sampled (we recall that \( i \) is sampled iff all items presented before \( i \) where not relevant) less than \( p_i(1 - \epsilon)k_i(n) \) times or and its empirical mean deviates from its expectation by more than \( \delta \). \( K_{i,2} \) is the number of instants in \( K_i \) where \( p_i(1 - \epsilon)k_i(n) \) is smaller than \( t_0 \), i.e \( K_{i,2} \) is the set of the first \( t_0p_i^{-1}(1 - \epsilon)^{-1} \) instants of \( K_i \).

Let us prove that \( K_i \subset K_{i,1} \cup K_{i,2} \). We proceed by contradiction: Consider \( n \in K_i \setminus (K_{i,1} \cup K_{i,2}) \). We prove that we have both (a) \( t_i(n) \geq t_0 \) and (b) \( b_i(n) \geq \theta_L - \delta \).
Since \( n \notin \mathcal{K}_{i,1} \) we have that \( t_i(n) \geq p^{-1}(1 - \epsilon)k_i(n) \) and since \( n \notin \mathcal{K}_{i,2} \) we have \( p_i(1 - \epsilon)k_i(n) \geq t_0 \). So (a) holds. By definition of the algorithm, we have that \( i \in \mathcal{B}(n) \), so that \( b_i(n) \geq \hat{\theta}_{j_L(n)}(n) \). Furthermore, since \( n \in \mathcal{K}_i \) we have that \( n \notin \mathcal{C} \), so that \( j_L(n) = L \), and \( |\hat{\theta}_L(n) - \theta_L| \leq \delta \). In turn, this implies \( b_i(n) \geq \hat{\theta}_{j_L(n)}(n) = \hat{\theta}_L(n) \geq \theta_L - \delta \) so (b) holds as well. Combining (a) and (b) with the definition of \( b_i(n) \):

\[
t_0 I(\hat{\theta}_i(n), \theta_L - \delta) \leq t_i(n) I(\hat{\theta}_i(n), \theta_L - \delta) \leq f(n) \leq f(T),
\]

and thus: \( I(\hat{\theta}_i(n), \theta_L - \delta) \leq I(\theta_i - \delta, \theta_L - \delta) \), which proves that \( |\hat{\theta}_i(n) - \theta_i| \geq \delta \) using the fact that the function \( x \mapsto I(x, y) \) is decreasing for \( 0 \leq x \leq y \). Hence \( n \in \mathcal{K}_{i,1} \) which is a contradiction since we assumed that \( n \in \mathcal{K}_i \setminus (\mathcal{K}_{i,1} \cup \mathcal{K}_{i,2}) \). Hence \( \mathcal{K}_i \subset \mathcal{K}_{i,1} \cup \mathcal{K}_{i,2} \) as announced. We now provide upper bounds on the expected sizes of \( \mathcal{K}_{i,1} \) and \( \mathcal{K}_{i,2} \).

Set \( \mathcal{K}_{i,1} \): Since \( n \in \mathcal{K}_{i,1} \subset \mathcal{K}_i \) implies \( u(n) = u^{i,j} \) we have that \( \mathbb{E}[|i|(n)|n \in \mathcal{K}_{i,1}] = p_i \). Applying Corollary 6.6.1 presented in Appendix (with \( H \equiv \mathcal{K}_{i,1} \) and \( c \equiv p_i \)) we obtain:

\[
\mathbb{E}[|\mathcal{K}_{i,1}|] \leq p_i^{-1} [p_i^{-1} - 2 + \delta^{-2} (1 - \epsilon)^{-1}].
\]

Set \( \mathcal{K}_{i,2} \): Since \( n \in \mathcal{K}_{i,2} \) implies that \( k_i(n) \leq t_0 p_i^{-1} (1 - \epsilon)^{-1} \) and that \( k_i(n) \) is incremented at \( n \), we have that:

\[
\mathbb{E}[|\mathcal{K}_{i,2}|] \leq t_0 p_i^{-1} (1 - \epsilon)^{-1}.
\]

Putting it all together we obtain the desired bound (4.5) on the expected size of \( \mathcal{K}_i \), which concludes the proof of the first statement of Theorem 4.3.2. The second statement of the theorem is obtained by taking the limit \( T \rightarrow \infty \) and then \( \delta \rightarrow 0 \).

### 4.4 Known Topic

In the remaining of the chapter, we consider \( K > 1 \) groups of users and items, and switch back to the notations introduced in Section 4.2. In this section, we consider the scenario where in each round \( n \), the topic of the request is known, i.e., the decision maker is informed about \( h(k(n)) \) before selecting the items to be displayed. In such a scenario, the problem of the design of sequential list selection algorithms can be decomposed into \( K \) independent bandit problems, one for each topic. Indeed in view of Assumption (A1), when the topic of the request is \( h(k) \), any algorithm should present, in the list, items from \( \mathcal{N}_{h(k)} \). The \( K \) independent MAB problems are instances of the problems considered in the previous section. As a consequence, we can apply the analysis of Section 4.3, and immediately deduce regret lower bounds and asymptotically optimal algorithms. Optimal algorithms are obtained by just running \( K \) independent PIE(l) algorithms, one for each topic. We refer to as \( K \times \text{PIE}(l) \) the resulting global algorithm.

Define \( \mathcal{U}_k \) as the set of lists containing items from \( \mathcal{N}_{h(k)} \) only, i.e., \( \mathcal{U}_k := \{ u \in \mathcal{U} : \forall s \in [L], u_s \in \mathcal{N}_{h(k)} \} \). We denote by \( \mathcal{U}_k(i) = \{ u \in \mathcal{U}_k : i_{h(k)}(k) = u \} \), the set of lists in \( \mathcal{U}_k \) that include the item \( i_{h(k)} \) with the \( i \)-th highest relevance in \( \mathcal{N}_{h(k)} \). Finally, for \( u \in \mathcal{U}_k(i) \), we refer to as \( p_i(u) \) as the position of \( i_{h(k)} \) in the list \( u \). The following theorem is a direct consequence of Theorem 4.3.1.
Theorem 4.4.1 Let $\theta \in [0,1]^{K \times N}$. For any uniformly good algorithm $\pi$, we have:

$$\liminf_{T \to \infty} \frac{R_\pi(T)}{\log(T)} \geq \sum_{k \in [K]} c_k(\theta),$$

(4.6)

where for any $k \in [K]$, $c_k(\theta)$ is the minimal value of the objective function in the following optimization problem $(P_{\theta,k})$:

$$\inf_{c_u \geq 0, u \in U_k} \sum_{u \in U_k} c_u (\mu_{\theta}^{*,k} - \mu_\theta(u, k))$$

s.t. $\sum_{u \in U_k(i)} c_u \prod_{s < p_i(u)} (1 - \theta_{ku,s}) I(\theta_{{k}i_{h(k)}}, \theta_{{k}h(k)}) \geq 1,$

$\forall i > L.$

The LPs $(P_{\theta,k})$ are similar to $(P_\theta)$ presented in Theorem 4.3.1, and enjoy the same simplifications (see Propositions 4.3.1 and 4.3.2) when the reward function has the specific structure of Case 1) or 2). Observe that the regret lower bound does not depend on the proportions of queries made by users of the various classes (remember that we assumed that $\phi_k > 0$ for all $k \in [K]$) – this is simply due to the facts that over the time horizon $T$, we roughly have $\phi_k T$ queries generated by class-$k$ users, and that the regret incurred for class-$k$ users is $c_k(\theta) \log(\phi_k T) \approx c_k(\theta) \log(T)$.

The next theorem is a direct consequence of Theorem 4.3.2, and states that $K \times \text{PIE}(L)$ and $K \times \text{PIE}(1)$ are asymptotically optimal in Cases 1) and 2), respectively.

Theorem 4.4.2 Assume that the reward function has the specific structure described in Case 1) (resp. 2)). Under algorithm $\pi = K \times \text{PIE}(L)$ (resp. $\pi = K \times \text{PIE}(1)$), we have for all $\theta$:

$$\limsup_{T \to \infty} \frac{R_\pi(T)}{\log(T)} \leq \sum_{k \in [K]} c_k(\theta).$$

4.5 Known User-Class and Unknown Topic

In this section, we address the problem with $K > 1$ groups of users and items, and where in each round $n$, the decision maker is aware of the class of the user issuing the query, but does not know the mapping $h$, i.e., initially, the decision maker does not know which topic the users of the various classes are interested in. Of course, this scenario is more challenging than the one where, before selecting a list of items, the decision maker is informed on the topic $h(k(n))$, and hence, the regret lower bound described in Theorem 4.4.1 is still valid.

Next we devise a sequential list selection algorithm that learns the mapping $h$ very rapidly. More precisely, we prove that its asymptotic regret satisfies the same regret upper bound as those derived for $K \times \text{PIE}(l)$ when the topic is known, which means that the fact that the mapping $h$ is unknown incurs a sub-logarithmic regret. Thus, our algorithm is asymptotically optimal since its regret upper bound matches the lower bound derived in Theorem 4.4.1.
To describe our algorithms, we introduce the following notations. Let \( u(n) \) be the list selected in round \( n \), and let \( p_i(n) \) denote the position at which item \( i \) is shown if \( i \in u(n) \), and \( p_i(n) = 0 \) otherwise. Let \( X_{ki}(n) \in \{0,1\} \) denote the relevance of item \( i \) when presented to a class-\( k \) user in round \( n \). Define

\[
o_{ki}(n) := \mathbb{1} \{ k(n) = k, i \in u(n), \forall l' < p_i(n), X_{ku, l'}(n)(n) = 0 \}
\]

the event indicating whether a query of a class-\( k \) user arrives in round \( n \) and this user scans item \( i \). Then we get a sample from \( \theta_{ki} \) in round \( n \) iff \( o_i(n) = 1 \). Let \( t_{ki}(n) := \sum_{n'<n} o_{ki}(n') \) be the number of samples obtained, up to round \( n \), for \( \theta_{ki} \). The corresponding empirical mean is:

\[
\hat{\theta}_{ki}(n) = \frac{1}{t_{ki}(n)} \sum_{n' \leq n} o_{ki}(n') X_{ki}(n')
\]

if \( t_{ki}(n) > 0 \) and \( \hat{\theta}_{ki}(n) = 0 \) otherwise. The KL-UCB index \( b_{ki}(n) \) of item \( i \) when presented to a class-\( k \) user in round \( n \) is:

\[
b_{ki}(n) = \max\{ q \in [0,1] : t_{ki}(n) I(\hat{\theta}_{ki}(n), q) \leq f(n) \},
\]

where \( f(n) = \log(n) + 4 \log(\log(n)) \). Finally, for any user class \( k \) and topic \( h \), we define \( j_{kh}(n) = (j_{kh,1}(n), ..., j_{kh,N_h}(n)) \), the items of \( N_h \) with empirical means sorted in decreasing order for users of class \( k \) in round \( n \). Namely:

\[
\hat{\theta}_{k,j_{kh,1}(n)}(n) \geq \hat{\theta}_{k,j_{kh,2}(n)}(n) \geq ... \geq \hat{\theta}_{k,j_{kh,N_h}(n)}(n)
\]

and \( j_{kh,i}(n) \in N_h \) for all \( k, h, \) and \( i \).

**The PIE-C\((l, d)\) Algorithm.** The algorithm is parametrized by \( l \in [L] \) which indicates the position in which apparently sub-optimal items are explored, and by \( d \), a real number chosen strictly between \( \delta \) and \( \Delta \). To implement such an algorithm, we do not need to know the maximum expected relevance \( \delta \) of items of uninteresting topics, nor the lower bound \( \Delta \) of the highest relevance of items whose topic corresponds to that of the query. We just need to know a number \( d \) in between.
In round \( n \), PIE-C\((l, d)\) maintains an estimator \( \hat{h}(n) \) of the topic \( h(k(n)) \) requested by the user, and it proceeds as follows. Given the user-class \( k(n) \), we first identify the set of \textit{admissible} topics \( C(n) \):

\[
C(n) = \{ h \in [K] : \max_{i \in N_h} \hat{\theta}_{k(n)i}(n) \geq d \}.
\]

This set corresponds to the topics that according to our observations up to round \( n \), could be the topic requested by the class-\( k(n) \) user.

(i) If \( C(n) = \emptyset \), \( \hat{h}(n) = -1 \) (we don’t know what the topic is), and we select \( u(n) \) uniformly at random over the set of possible decisions \( U \);

(ii) If \( C(n) \neq \emptyset \),

Select \( \hat{h}(n) \in C(n) \) uniformly at random;

Define leaders at time \( n \): \( \mathcal{L}(n) \) lists in order the \( L \) items in \( N_{\hat{h}(n)} \) with largest empirical means,

\[
\mathcal{L}(n) = (j_{k(n)\hat{h}(n),1}(n), ..., j_{k(n)\hat{h}(n),L}(n)));
\]

Define the possible decisions \( U_i(n) \) for all \( i \in N_{\hat{h}(n)} \setminus \mathcal{L}(n) \) obtained by replacing in \( \mathcal{L}(n) \) the \( l \)-th item by \( i \);

Define

\[
B(n) = \{i \in N_{\hat{h}(n)} \setminus \mathcal{L}(n) : b_{k(n)i}(n) \geq \hat{\theta}_{j_{k(n)\hat{h}(n),L}(n)}(n)\};
\]

(a) If \( B(n) = \emptyset \), select the list \( \mathcal{L}(n) \), and (b) If \( B(n) \neq \emptyset \), choose \( i(n) \) uniformly at random in \( B(n) \) and select either \( \mathcal{L}(n) \) with probability \( 1/2 \) or decision \( U_i(n) \) with probability \( 1/2 \).

Note that when \( \hat{h}(n) \) is believed to estimate \( h(k(n)) \) accurately (i.e., when \( C(n) \neq \emptyset \)), then the algorithm mimics the \( K \times \text{PIE}(l) \) algorithm. Refer to the pseudocode of PIE-C\((l, d)\) for a formal description. The following theorem states that PIE-C\((l, d)\) exhibits the same asymptotic regret as the optimal algorithms when the topic of each request is known.

\textbf{Theorem 4.5.1} Assume that the reward function has the specific structure described in Case 1) (resp. 2)). For all \( \delta < d < \Delta \), under the algorithm \( \pi = \text{PIE-C}(L, d) \) (resp. \( \pi = \text{PIE-C}(1, d) \)), we have for all \( \theta \):

\[
\lim_{T \to \infty} \sup \frac{R^\pi(T)}{\log(T)} \leq \sum_{k \in [K]} c_k(\theta).
\]
4.5.2 Proof: Regret Upper Bound for PIE-C(l, d)

The proof of Theorem 4.5.1 consists in showing that the set of rounds at which the estimation of the topic \( h(k(n)) \) of the request fails is finite in expectation. As already mentioned, when the estimation is correct the algorithm behaves like \( K \times \text{PIE}(l) \), and the analysis of its regret in such rounds is the same as that under \( K \times \text{PIE}(l) \). Hence, we just need to control the size of the following set of rounds:

\[
\mathcal{M} = \{ n \geq 1 : \hat{h}(n) \neq h(k(n)) \}.
\]

**Lemma 4.5.1** Under algorithm PIE-C(l,d) we have:

\[
\mathbb{E}[|\mathcal{M}|] \leq 2KN \left[ 2(N + 1) + (d - \delta)^{-2} + (\Delta - d)^{-2} \right]
\]

The above bound is minimized by setting \( d = (\Delta + \delta)/2 \), in which case:

\[
\mathbb{E}[|\mathcal{M}|] \leq 4KN \left[ N + 4(\Delta - \delta)^{-2} \right]
\]

**Proof.** For all \( k \), we define the most popular item for class-\( k \) users: \( i_k^* = \arg \max_i \theta_{ki} \). We decompose \( \mathcal{M} \) by introducing the following sets:

\[
\mathcal{M}_k = \{ n \in \mathcal{M}, k(n) = k \},
\]

\[
\mathcal{M}_{k,-1} = \{ n \in \mathcal{M}_k, \hat{h}(n) = -1, |\hat{\theta}_{ki_k^*}(n) - \theta_{ki_k^*}| \geq \Delta - d \},
\]

\[
\mathcal{M}_{k,i} = \{ n \in \mathcal{M}_k, i = u_1(n), |\hat{\theta}_{ki}(n) - \theta_{ki}| \geq d - \delta \}.
\]

\( \mathcal{M}_{k,-1} \) is the set of rounds at which a user of class \( k \) makes a request, the set of admissible topics for class \( k \) users is empty \( C(n) \), and \( \theta_{ki_k^*} \) is badly estimated. \( \mathcal{M}_{k,i} \) is the set of rounds at which a user of class \( k \) makes a request, item \( i \notin \mathcal{N}_{h(k)} \) is presented in the first slot (note that \( i \) is not interesting to that user) and \( \theta_{ki} \) is badly estimated. We have that:

\[
\mathcal{M} = \bigcup_{k=1}^{K} \mathcal{M}_k
\]

We prove that for all \( k \): \( \mathcal{M}_k \subset \mathcal{M}_{k,-1} \cup (\bigcup_{i \notin \mathcal{N}_{h(k)}} \mathcal{M}_{k,i}) \).

Consider \( n \in \mathcal{M}_k \), so that \( k(n) = k \) and \( \hat{h}(n) \neq h(k) \). We distinguish two cases:

(i) If \( \hat{h}(n) = -1 \), then \( C(n) = \emptyset \). So \( h(k) \notin C(n) \), and by definition of \( C(n) \), this implies that \( \max_{i \in \mathcal{N}_{h(k)}} \hat{\theta}_{ki}(n) \leq d \). Since \( i_k^* \in \mathcal{N}_{h(k)} \), we have \( \hat{\theta}_{ki_k^*}(n) \leq d \). Since \( i_k^* = \arg \max_i \theta_{ki} \), we have \( \theta_{ki_k^*} \geq \Delta \). Hence we have that both \( \hat{\theta}_{ki_k^*}(n) \leq d \) and \( \theta_{ki_k^*} \geq \Delta \), so we have \( |\hat{\theta}_{ki_k^*}(n) - \theta_{ki_k^*}| \geq \Delta - d \) and therefore \( n \in \mathcal{M}_{k,-1} \).

(ii) If \( \hat{h}(n) \notin \{h(k),-1\} \), then by design of the algorithm \( u(n) \subset \{1, \ldots, N\} \setminus \mathcal{N}_{h(k)} \) since \( \{N_1, \ldots, N_K\} \) forms a partition of \( \{1, \ldots, N\} \). Hence there exists \( i \notin \mathcal{N}_{h(k)} \) such that \( u_1(n) = i \). By design of the algorithm, since \( u_1(n) = i \), we have \( \hat{\theta}_{ki}(n) = \arg \max_{i' \in \mathcal{N}_{h(n)}} \hat{\theta}_{ki'}(n) \) and \( \arg \max_{i' \in \mathcal{N}_{h(n)}} \hat{\theta}_{ki'}(n) \geq d \) since \( \hat{h}(n) \in C(n) \). Therefore \( \hat{\theta}_{ki}(n) \geq d \) and we know that \( \theta_{ki} \leq \delta \) since \( i \notin \mathcal{N}_{h(k)} \), so that \( |\hat{\theta}_{ki}(n) - \theta_{ki}| \geq d - \delta \). Summarizing, \( \hat{h}(n) \notin \{h(k),-1\} \) implies that there exists \( i \notin \mathcal{N}_{h(k)} \) such that \( u_1(n) = i \) and \( |\hat{\theta}_{ki}(n) - \theta_{ki}| \geq d - \delta \), therefore \( n \in \bigcup_{i \notin \mathcal{N}_{h(k)}} \mathcal{M}_{k,i} \).
Hence we have proven, as announced, that \( M_k \subset M_{k,-1} \cup_{i \notin N_{h(k)}} M_{k,i} \). We now upper bound the expected sizes of sets \( M_{k,-1} \) and \( M_{k,i} \).

Set \( M_{k,-1} \): When \( n \in M_{k,-1} \), \( u(n) \) is uniformly distributed over the set of possible decisions \( \mathcal{U} \), so that \( \mathbb{P}[u_1(n) = i^*_k | n \in M_{k,-1}] = 1/N \). In turn, this implies that \( \mathbb{E}[o_{k,i}(n)|n \in M_{k,-1}] = 1/N \). Applying Lemma 6.6.3, second statement (with \( H \equiv M_{k,-1}, c \equiv 1 \) and \( \delta \equiv \Delta - d \)), we obtain:

\[
\mathbb{E}[|M_{k,-1}|] \leq 2N \left[ 2N + (\Delta - d)^{-2} \right].
\]

Set \( M_{k,i} \): When \( n \in M_{k,i} \), we have that \( u_1(n) = i \) and \( k(n) = k \) so that \( \mathbb{E}[o_{k,i}(n)|n \in M_{k,i}] = 1 \). Applying Lemma 6.6.3, second statement (with \( H \equiv M_{k,i}, c \equiv 1 \) and \( \delta \equiv d - \delta \)), we obtain:

\[
\mathbb{E}[|M_{k,i}|] \leq 2 \left[ 2 + (d - \delta)^{-2} \right].
\]

Using a union bound we have:

\[
\mathbb{E}[|M_{k,-1}|] \leq \mathbb{E}[|M_{k,-1}|] + \sum_{i \notin N_{h(k)}} \mathbb{E}[|M_{k,i}|]
\leq 2N \left[ 2N + (\Delta - d)^{-2} \right] + 2N \left[ 2 + (d - \delta)^{-2} \right]
= 2N \left[ 2(N + 1) + (d - \delta)^{-2} + (\Delta - d)^{-2} \right],
\]

and summing over \( k \in \{1, ..., K\} \) we obtain the announced result:

\[
\mathbb{E}[|M|] = \sum_{k=1}^{K} \mathbb{E}[|M_k|]
\leq 2KN \left[ 2(N + 1) + (d - \delta)^{-2} + (\Delta - d)^{-2} \right],
\]

which concludes the proof.

4.6 Numerical Experiments

In this section, we evaluate the practical performance of our algorithms using both artificially generated and real-world data\(^1\).

4.6.1 Artificial Data

We first evaluate the PIE and PIE-C algorithms in the scenarios presented in Sections 4.3, 4.4, and 4.5. In these scenarios, the algorithms are optimal and hence they should outperform any other algorithm.

\(^1\)We use the Movielens10M dataset, available at http://grouplens.org/datasets/movielens/
A Single group of users / items. First we assume there exists only one relevant topic ($K = 1$) consisting of $N = 800$ items. We consider $L = 10$ and evaluate the performance of the algorithms over the arrival of $T = 8 \times 10^4$ user queries. The parameter $\theta$ is artificially generated as follows:

$$\theta_i = 0.55 \times (1 - (i - 1)/(N - 1)).$$

In Figure 4.1, in the left plot, we consider the reward to be $r(l) = 1$ for $l \in \{1, ..., L\}$ while in the right plot, we assume the reward decreases geometrically with the slot ($r(l) = 2^{1-l}$, $l \in \{1, ..., L\}$). Under these assumptions, PIE(1) and PIE(L) respectively are asymptotically optimal according to Theorem 2. We compare their performance to that of Slotted UCB, Slotted KL-UCB algorithms, and RBA (Ranked Bandit Algorithm) proposed in [26] and [30]. In Slotted UCB (resp. KL-UCB), the $L$ items with the largest UCB (resp. KL-UCB) indexes are displayed, whereas RBA runs $L$ independent bandit algorithms, one for each slot. In particular, for all items $k$, the bandit algorithm assigned to slot $l$ can only access the observations obtained from $k$ when $k$ was played in slot $l$ (RBA attempts to learn an item’s so-called marginal utility for each slot). Observe that PIE significantly outperforms all other algorithms.

Multiple groups of users / items. Next, we consider $K = 5$ groups of users and items, and $N = 4,000$ items. We assume all groups are of equal size so that $\phi_k = 1/K$ for all $k$. There are $N/K$ items in each group. We define $j(i,k) = (i - h(k)N/K)$, and generate the parameter $\theta$ as follows:

$$\theta_{ki} = \begin{cases} 0.55 \times (1 - (j(i,k) - 1)/(N - 1)) & \text{if } i \in \mathcal{N}_{h(k)}, \\ 0.05 & \text{otherwise.} \end{cases}$$

Figure 4.2 presents the performance of $5 \times$ PIE(1) (referred to as PIE(1) in the figure) when the decision maker knows the mapping between user classes and topics $h(\cdot)$, and that of PIE-C(1,0.5) when $h(\cdot)$ is unknown. Figure 4.2 corroborates the theoretical result of
Theorem 4.5.1: The performance loss due to the need to learn the mapping \( h(\cdot) \) is rather limited, especially the time horizon grows large.

4.6.2 Real-world Data

We further investigate the performance of our algorithms on real-world systems. We use the Movielens dataset which contains the ratings given by users to a large set of movies. The dataset is a large matrix \( X = (X_{a,m}) \) where \( X_{a,m} \in \{0, 1, ..., 5\} \) is the rating given by user \( a \) to movie \( m \). The highest rating is 5, the lowest is 1, and 0 denotes an absence of rating, as most users did not watch the whole set of movies. From matrix \( X \), we created a binary matrix \( Y \) such that \( Y_{a,m} = 0 \) if \( X_{a,m} < 4 \) and \( Y_{a,m} = 1 \) otherwise. We say that movie \( m \) is \textit{interesting} to user \( a \) iff \( Y_{a,m} = 1 \).

We first selected the 100 most popular movies with less than 13,000 ratings (to avoid
movies with good ratings for a large majority of users) and the 61,357 users who rated at least one of those movies. We extracted the corresponding sub-matrix of $Y$. To cluster the users and the movies, we use the classical spectral method. We extracted the 4 largest singular values of $Y$ and their corresponding singular vectors $\gamma_i, i \in \{1, 2, 3, 4\}$. We then assigned each user $a$ to the cluster $k = \arg\max_i Y_a \cdot \gamma_i$, where $Y_a$ is the $a$-th line of matrix $Y$. We performed a similar classification of movies.

In Figure 4.3, on the left, we consider class-1 users, and compare the performance of algorithms already considered in Subsection 4.6.1, Scenario 1. The simulation proceeds as follows: in round $n$, we draw a class-1 user, denoted by $a(n)$, uniformly at random. The considered algorithm chooses an action $u(n)$, if $u(n)$ contains an interesting movie $i$ for user $a(n)$, i.e., $Y_{a(n)i} = 1$, the system collects a unit reward, otherwise the reward is 0. We emulate the semi-bandit feedback by assuming the algorithm is informed about uninteresting movies $j$, i.e., $Y_{a(n)j} = 0$, placed above $i$ in the list of $L = 10$ movies. Here the performance of the algorithm is quantified through the notion of abandonment, as introduced in [26]. The abandonment is the number of rounds in which no interesting movies are displayed. As a benchmark, we use an Oracle policy that displays the $L$ most popular movies for users of class 1 in every round. Note that we use abandonment as a performance metric rather than the regret, because the optimal policy is hard to compute given the fact that the ratings offered by a user to different movies are not always independent in our data set. Again, PIE outperforms the Slotted variants of UCB and KL-UCB which in turn significantly outperform RBA(KL-UCB). In fact, the cost of learning in PIE (compared to the Oracle policy is limited): the abandonment under PIE does not exceed twice that of the Oracle policy. Note that the performance gain under PIE compared to Slotted-KL-UCB is much higher in our artificial data simulations. We believe that this may be firstly due to the inaccuracy of our model when used against this particular data-set, and secondly due to the fact that the gain under PIE increases with the number of items $N$.

On the right of Figure 4.3, we consider $K = 4$ groups, each topic consisting of 25 items. Again, the performance of PIE-C algorithm is not too far from that of the Oracle policy. PIE-C is compared to Slotted KL-UCB and a Slotted KL-UCB aware of the groups and of the mapping $h(\cdot)$. The former just ignores the group structure and runs as if there were a single group only, whereas the latter consists in $K$ parallel and independent instances of Slotted KL-UCB, one for each user class $k$ and item group $h(k)$. PIE-C outperforms Slotted KL-UCB, and its performance is similar to that of Slotted KL-UCB with known mapping $h(\cdot)$. Again this indicates that PIE-C rapidly learns the mapping $h(\cdot)$.

4.7 Conclusions

In this chapter we presented the online learning to rank problem from a multi-armed bandit perspective. We proposed two algorithms PIE and PIE-C which rapidly learn the preferences of users and are proven asymptotically optimal for two broad families of reward functions. These algorithms are designed under the assumption that users and items are clustered and the decision maker is aware of the class of the user making queries. An interesting continuation on this topic would be extending our algorithms to settings when...
the class of the user making the query is not revealed to the decision maker and further investigating their performance in real-world scenarios and using real traces from search engines such as Google, Bing or Spotify.
In this thesis we tackled the problem of Multi-Armed Bandits with correlated arms. We presented the basic principles behind MAB theory and examined two varieties of structured bandits with practical applications: the Lipschitz Bandits and Learning to Rank from a MAB perspective. We derived a regret lower bound for generic bandit settings with correlated arms as an application of the results of Graves and Lai in [4]. We showed how to develop a statistically efficient index for the Lipschitz Bandit that combines the samples of all arms to generate a tight upper confidence bound on the empirical mean of any arm. We used this index to develop and analyze two (pareto) optimal algorithms: OSLB and POSLB. We showed how these two algorithms numerically outperform the state of the art in both discrete and continuous settings. In the case of learning to rank, we propose and analyze two algorithms PIE and PIE-C. We show they are optimal for specific families of reward functions and exhibit their numerical performance on both artificial and real-world datasets.

We believe the insight from this thesis can be used to further tackle MAB problems with even more complex correlations between arms. We believe efficient indexes for any bandit problem where the arms are correlated can be devised following the same principles used to derive the index used by POSLB and OSLB. Furthermore, we are interested in pursuing the extension of these techniques to problems with infinitely many arms. In these cases, we do not have problem specific lower bounds and while the index presented here work well numerically, they do not exhibit all the properties one would expect from a tight confidence bound on the empirical rewards (continuity when arms are arbitrarily close).

While the Continuous Lipschitz Bandit problem provides a great stepping stone to a better understanding of correlated bandits, we would like to focus on more complex and useful structures in the future. A great example being the Linear Bandit, as portrayed in Section 1.1.1. While this type of structure is harder to exploit, it implies a much stronger correlation between the rewards of arms compared to the Lipschitz Bandit. Therefore, it offers substantially more potential to reduce regret relative to algorithms that do not exploit structure. Determining an algorithm that is efficient both in terms of regret and computation in this setting would represent an important step forward for the cold start problem in recommender systems.
Since the infinite armed are useful in practice when the time horizon is not very large relative to the number of arms, it is worth investigating finite time optimality in MAB. More precisely, what is the best possible expected regret at any time \( n < \infty \). The results of Lai and Robbins [1] (and later Graves and Lai [4]) only hold asymptotically, when the time horizon \( T \to \infty \). Therefore, the question what is the best that can be done in finite time still stands. Since KL-UCB uses likelihood ratio testing (which is known to be the most powerful statistical test between two point-hypotheses), it is not far fetched to imagine an optimal finite time algorithm would use the same techniques. Particularly, using the same sequential statistical testing approach but moving away from the *optimism in the face of uncertainty* principle. While this paradigm yields an optimal algorithm when \( T \to \infty \), its numerical inferiority to TS in finite time indicates it is still exploring too much when \( T \) is small. Another indication of this problem is the fact that one can, in practice, ignore the \( \log\log(T) \) terms in the index and still increase the performance of the algorithm. We postulate that an optimal finite time algorithm would not allow such *shortcuts* to increase its performance.


Chapter 6

Appendix

6.1 Proof of Theorem 2.2.1

To establish the asymptotic lower bound, we apply the techniques used in [4] to investigate efficient adaptive decision rules in controlled Markov chains. We recall here their general framework. Consider a controlled Markov chain \((X_t)_{t \geq 0}\) on a finite state space \(V\) with a control set \(U\). The transition probabilities given control \(u \in U\) are parametrized by \(\theta\) taking values in a compact metric space \(\Theta\): the probability to move from state \(x\) to state \(y\) given the control \(u\) and the parameter \(\theta\) is \(p(x, y; u, \theta)\). The parameter \(\theta\) is not known. The decision maker is provided with a finite set of stationary control laws \(G = \{g_1, \ldots, g_K\}\) where each control law \(g_j\) is a mapping from \(V\) to \(U\): when control law \(g_j\) is applied in state \(x\), the applied control is \(u = g_j(x)\). It is assumed that if the decision maker always selects the same control law \(g\), the Markov chain is irreducible with stationary distribution \(\pi^g_\theta\). Now the expected reward obtained when applying control \(u\) in state \(x\) is denoted by \(r(x, u)\), so that the expected reward achieved under control law \(g\) is: \[\mu^g_\theta = \sum_x r(x, g(x)) \pi^g_\theta(x)\]. There is an optimal control law given \(\theta\) whose expected reward is denoted \(\mu^*_\theta \in \arg \max_{g \in G} \mu^g_\theta\). Now the objective of the decision maker is to sequentially select control laws so as to maximize the expected reward up to a given time horizon \(T\). As for MAB problems, the performance of a decision scheme can be quantified through the notion of regret which compares the expected reward to that obtained by always applying the optimal control law.

We now apply the above framework to our correlated bandit problem, and we consider \(\theta \in \Theta_L\). The Markov chain has values in \(\{\{0, 1\}^{|S|} : S \in S\}\). The set of control laws is \(G = S\). These laws are constant, in the sense that the control applied by control law \(S\) does not depend on the state of the Markov chain, and corresponds to selecting superarm \(S\). The transition probabilities are:

\[p(x, y; S, \theta) = \prod_{i \in S} (\mathbb{1}_{[y_i = 1]} \theta_i + \mathbb{1}_{[y_i = 0]} (1 - \theta_i))\]

and the stationary distribution is given by \(\pi^S_\theta(y) = p(x, y; S, \theta)\). Consequently the Kullback-
Liebler information number has the form:

$$I^S(\theta, \lambda) = \sum_{k \in S} I(\theta_k, \lambda_k)$$

Finally, the reward $r(x, k)$ is just given by $\eta(x)$ where $\eta$ is as defined in section 2.2.1.

We now fix $\theta \in \Theta$. Define the set $B(\theta)$ consisting of all bad parameters $\lambda \in \Theta$ such that $S^*$ is not optimal under parameter $\lambda$, but which are statistically indistinguishable from $\theta$:

$$B(\theta) = \bigcup_{S \in \mathcal{S}} \Lambda^S(\theta),$$

By applying Theorem 1 in [4], we know that $C(\theta)$ is the minimal value of the following LP:

$$\min \sum_{S \in \mathcal{S}} c_S (\mu^* - \mu_\theta(S))$$

subject to

$$\inf_{\lambda \in B(\theta)} \sum_{S \in \mathcal{S}} c_S \sum_{k \in S} I(\theta_k, \lambda_k) \geq 1,$$

$$c_S \geq 0, \quad \forall S \in \mathcal{S}.$$  \hspace{1cm} (6.2)

To conclude the proof, it is sufficient to remark that the constraint in (6.2) is equivalent to:

$$\forall S \in \mathcal{S}, \quad \inf_{\lambda \in \Lambda^S(\theta)} \sum_{k \in \mathcal{K}} I(\theta_k, \lambda_k) \sum_{V \in \mathcal{S}} c_V \mathbb{1}_{k \in V} \geq 1,$$

which is easy in view of the definition of $B(\theta)$.

6.2 Proof of Theorem 3.7.1

Similar as in the case of the discrete Lipschitz bandit, we can describe our system as a Markov chain with the state space being comprised of pairs $(x, i)$ with the reward $x \in \{0, 1\}$ and the context $i \in \mathcal{C}$. The probability of transition from a state $(x, i)$ to a state $(y, j)$ is as follows:

$$p((x, i), (y, j); k, \theta, \mathbb{P}) = \begin{cases} P(j) \times \theta_k(i), & \text{if } y = 1 \\ P(j) \times (1 - \theta_k(i)), & \text{if } y = 0 \end{cases} \quad (6.4)$$

with $P(j)$ is the probability with which we are presented with a user of context $j$. The sought policy would thus be a mapping $g^* \in G$ with:

$$G = \{ g : \mathcal{C} \to \mathcal{K} \}.$$  \hspace{1cm} (6.5)

Let $\mu_g(\theta)$ be the average reward under the control law $g$:

$$\mu_g(\theta, \mathbb{P}) = \sum_{i \in \mathcal{C}} P(i) \times \theta_{g(i)}(i).$$  \hspace{1cm} (6.6)
therefore, the unique optimal control law will be:

\[ g^* = \arg\max_{g \in G} \mu_g(\theta, \mathbb{P}). \] (6.7)

The stationary distribution, \( \pi^g_{\theta, \mathbb{P}} \), of the markov chain under the stationary strategy \( g \) and parameter \( (\theta, \mathbb{P}) \), is given by:

\[
\begin{align*}
\pi^g_{\theta, \mathbb{P}}(0, i) &= \mathbb{P}(i) \times \sum_{j \in \mathbb{C}} \mathbb{P}(j) \times (1 - \theta_{g(j)}(j)) \\
\pi^g_{\theta, \mathbb{P}}(1, i) &= \mathbb{P}(i) \times \sum_{j \in \mathbb{C}} \mathbb{P}(j) \times \theta_{g(j)}(j)
\end{align*}
\] (6.8)

Therefore, the Kullback-Leibler information number in [4] is:

\[
I^g((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2)) = \sum_{i \in \mathbb{C}} \sum_{j \in \mathbb{C}} \sum_{x \in \{0,1\}} \sum_{y \in \{0,1\}} \log \left( \frac{p((x, i), (y, j) : g(x), \theta, \mathbb{P}_1)}{p((x, i), (y, j) : g(x), \lambda, \mathbb{P}_2)} \right) \\
\times p((x, i), (y, j) : g(x), \theta, \mathbb{P}_1) \times \pi^g_{\theta, \mathbb{P}_1}(x, i)
\]

Expanding, we obtain

\[
I^g((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2)) = A1((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2), i, j) + A2((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2), i, j) + A3((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2), i, j) + A4((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2), i, j)
\] (6.9)

where:

\[
A1((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2), i, j) = \sum_{i \in \mathbb{C}} \sum_{j \in \mathbb{C}} \log \left( \frac{\mathbb{P}_1(j) \times (1 - \theta_{g(i)}(i))}{\mathbb{P}_2(j) \times (1 - \lambda_{g(i)}(i))} \right) \mathbb{P}_1(j) \\
\times (1 - \theta_{g(i)}(i)) \times \mathbb{P}_1(i) \times \sum_{y \in \mathbb{C}} \mathbb{P}_1(y) \times (1 - \theta_{g(y)}(y))
\]

\[
A2((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2), i, j) = \log \left( \frac{\mathbb{P}_1(j) \times (1 - \theta_{g(i)}(i))}{\mathbb{P}_1(j) \times (1 - \lambda_{g(i)}(i))} \right) \mathbb{P}_1(j) \\
\times (1 - \theta_{g(i)}(i)) \times \mathbb{P}_1(i) \sum_{y \in \mathbb{C}} \mathbb{P}_1(y) \times \theta_{g(y)}(y)
\]

\[
A3((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2), i, j) = \log \left( \frac{\mathbb{P}_1(j) \times \theta_{g(i)}(i)}{\mathbb{P}_2(j) \times \lambda_{g(i)}(i)} \right) \mathbb{P}_1(j) \\
\times \theta_{g(i)}(i) \times \mathbb{P}_1(i) \sum_{y \in \mathbb{C}} \mathbb{P}_1(y) \times (1 - \theta_{g(y)}(y))
\]

\[
A4((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2), i, j) = \log \left( \frac{\mathbb{P}_1(j) \times \theta_{g(i)}(i)}{\mathbb{P}_2(j) \times \lambda_{g(i)}(i)} \right) \mathbb{P}_1(j) \\
\times \theta_{g(i)}(i) \times \mathbb{P}_1(i) \sum_{y \in \mathbb{C}} \mathbb{P}_1(y) \times (1 - \theta_{g(y)}(y))
\]
\[ A4((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2), i, j) = \log \left( \frac{\mathbb{P}_1(j) \times \theta_{g(i)}(i)}{\mathbb{P}_2(j) \times \lambda_{g(i)}(i)} \right) \mathbb{P}_1(j) \times \theta_{g(i)}(i) \times \mathbb{P}_1(i) \sum_{y \in \mathcal{C}} \mathbb{P}_1(y) \times \theta_{g(y)}(y) \]  

Grouping the terms in the first two rows and the ones in the last two we notice the sums over \( y \in \mathcal{C} \) add up to 1. Thus the expression becomes:

\[
I^g(\theta, \lambda) = \sum_{j \in \mathcal{C}} \sum_{i \in \mathcal{C}} \log \left( \frac{(1-\theta_{g(i)}(i))\mathbb{P}_1(j)}{(1-\lambda_{g(i)}(i))\mathbb{P}_2(j)} \right) \mathbb{P}_1(j) \times (1 - \theta_{g(i)}(i)) \times \mathbb{P}_1(i) \\
+ \log \left( \frac{\theta_{g(i)}(i)\mathbb{P}_1(j)}{\lambda_{g(i)}(i)\mathbb{P}_2(j)} \right) \mathbb{P}_1(j) \times \theta_{g(i)}(i) \times \mathbb{P}_1(i)  
\]

(6.10)

Since \( \sum_{j \in \mathcal{C}} \mathbb{P}(j) = 1 \), we have the final form of the information number:

\[
I^g(\theta, \lambda) = \sum_{i \in \mathcal{C}} (\mathbb{P}(i) \times I(\theta_{g(i)}(i), \lambda_{g(i)}(i)) + \log \left( \frac{\mathbb{P}_1(j)}{\mathbb{P}_2(j)} \right) \times \mathbb{P}_1(j) \times \mathbb{P}_1(i).  
\]

(6.11)

Using \( I^g((\theta, \mathbb{P}_1), (\lambda, \mathbb{P}_2)) \) as above and define \( B(\theta, \mathbb{P}) = \{ (\lambda, \mathbb{P}') \in \Theta \times (0, 1)^{|\mathcal{C}|} : \exists g \neq g^* \text{ such that } \mu_g(\lambda, \mathbb{P}') > \mu_{g^*}(\lambda, \mathbb{P}) \text{ and } I^{g^*}((\lambda, \mathbb{P}'), (\theta, \mathbb{P})) = 0 \} \). Note that all \( \lambda \in B(\theta, \mathbb{P}) \) must respect the Lipschitz continuity constraint that applies to \( \theta \). Furthermore, \( I^{g^*}((\lambda, \mathbb{P}'), (\theta, \mathbb{P})) = 0 \) means that:

\[
\sum_{i \in \mathcal{C}} \mathbb{P}(i) \times I(\theta_{g^*(i)}(i), \lambda_{g^*(i)}(i)) = 0
\]

(6.12)

and:

\[
\sum_{i \in \mathcal{C}} \log \left( \frac{\mathbb{P}_1(j)}{\mathbb{P}_2(j)} \right) \times \mathbb{P}_1(j) \times \mathbb{P}_1(i) = 0.
\]

(6.13)

Therefore, \( \forall (\lambda, \mathbb{P}') \in B(\theta, \mathbb{P}) \), we have \( \mathbb{P}' = \mathbb{P} \). Thus, we can use the following shorthand notations:

\[
I^g(\theta, \lambda) = \sum_{i \in \mathcal{C}} \mathbb{P}(i) \times I(\theta_{g^*(i)}(i), \lambda_{g^*(i)}(i))
\]

(6.14)

and

\[
B(\theta) = \{ \lambda \in \Theta : \exists g \neq g^* \text{ such that } \mu_g(\lambda) > \mu_{g^*}(\theta) \text{ and } I^{g^*}(\lambda, \theta) = 0 \}
\]

(6.15)

Applying the results of Graves and Lai in [4] on the Markov chain model of the contextual bandit, we obtain that, for \( n \to \infty \) rounds and parameters \( \theta \), the regret of the contextual bandit satisfies:

\[
R_n(\theta) \geq c(\theta) \times \log(n),
\]

(6.16)
6.2. Proof of Theorem 3.7.1

where

\[
c(\theta) = \inf_{c_g \geq 0} \sum_{g \neq g^*} c_g \times \sum_{k \in K} \sum_{i \in C} \mathbb{P}(i) \times (\theta^*(i) - \theta_k(i)) \times 1(g(i) = k) \tag{6.17}
\]

(where \(g^*\) is the optimal strategy), subject to:

\[
\inf_{\lambda \in B(\theta)} H(\lambda, c) = \sum_{i \in C} \mathbb{P}(i) \times I(\theta_k(i), \lambda_k(i)) \sum_{g \neq g^*} c_g \times 1(g(i) = k) \geq 1, \forall k \in K. \tag{6.18}
\]

where \(B(\theta) = \{ \lambda : \exists g \neq g^* \text{ such that } \mu_g(\lambda) > \mu_{g^*}(\lambda), \text{ and } \forall i \in C, \theta_{g^*}(i) = \lambda_{g^*}(i) \} \).

Let \(\lambda^{y,y,i}, y \in K, i \in C, q \in (\theta_y(i), 1]\) be such that \(\forall j \in C \text{ and } y \in K:\)

\[
\lambda_{k,j}^{y,y,i}(j) = \max(\theta_k(j), q - L \times D((k, i), (y, j))) \tag{6.19}
\]

From theorem 6.2.1, we thus have that the solution to the optimization problem in (6.17) is identical to the solution to the following problem:

\[
c(\theta) = \inf_{c_g \geq 0} \sum_{k \in K} \sum_{i \in C} (\theta^*(i) - \theta_k(i)) \times \mathbb{P}(i) \times \sum_{g \in G} 1(g(i) = k) \tag{6.20}
\]

subject to:

\[
\sum_{k \in K} \sum_{j \in C} I(\theta_k(j), \lambda_{k,j}^{\theta^*(i),y,i}(j)) \times \mathbb{P}(i) \times \sum_{g \in G} 1(g(j) = k) \geq 1. \tag{6.21}
\]

Defining:

\[
c_{k,j} = \mathbb{P}(i) \times \sum_{g \in G} 1(g(j) = k) \tag{6.22}
\]

and replacing above, we obtain the announced form of the optimization problem representing the Lipschitz contextual bandits regret lower bound:

\[
c(\theta) = \inf_{c_g \geq 0} \sum_{k \in K} \sum_{i \in C} (\theta^*(i) - \theta_k(i)) \times c_{k,j} \tag{6.23}
\]

subject to:

\[
\sum_{k \in K} \sum_{j \in C} I(\theta_k(j), \lambda_{k,j}^{\theta^*(i),y,i}(j)) \times c_{k,j} \geq 1. \tag{6.24}
\]

\[\square\]

**Theorem 6.2.1** The minimum of the expression (6.18) is given by \(\lambda^{\theta^*(y),y,k}\) for some context \(y \in C\) and some arm \(k \in K\).
Proof. Assume the minimum for the desired expression is achieved for some $\lambda^* \in B(\theta)$, and therefore $H(\lambda^*, c) < H(\lambda^{\theta^*(i), y, i}, \forall c \in \mathbb{R}^{|G|})$, and for $i \in C$ and $y \in K$ such that $\lambda^*_y(i) \geq \theta^*(i) \neq \theta_y(i)$. Define the set:

$$A(y, i) = \{(k, j) \in K \times C : \theta^*(i) - L \times D((y, i), (k, j)) > \theta_k(j)\}$$  \hspace{1cm} (6.25)

Then, because $\lambda^*$ is Lipschitz and

$$\lambda_k^{\theta^*(i), y, i}(j) = \theta^*(i) - L \times D((k, i), (y, j)), \hspace{1cm} (6.26)$$

$\forall (k, j) \in A(y, i)$ we have:

$$I(\theta_k(j), \lambda^*_k(j)) \geq I(\theta_k(j), \lambda_k^{\theta^*(i), y, i}(j)) \iff \lambda^*_y(i) \geq \lambda_y^{\theta^*(i), y, i}(i) = \theta^*(i).$$  \hspace{1cm} (6.27)

Thus, since $\forall (k, j) \in A(y, i)$:

$$I(\theta_k(j), \lambda^*_k(j)) \geq I(\theta_k(j), \lambda_k^{\theta^*(i), y, i}(j)) \hspace{1cm} (6.28)$$

and $\forall (k, j) \notin A(y, i)$, by how we define $\lambda_k^{\theta^*(i), y, i}$:

$$I(\theta_k(j), \lambda_k^{\theta^*(i), y, i}(j)) = 0,$$  \hspace{1cm} (6.29)

we have obtain $H(\lambda^*, c) < H(\lambda^{\theta^*(i), y, i})$, which concludes the proof. \hfill \Box

6.3 Proof of Theorem 3.5.2 - OSLB Finite Time Analysis

We first present two important corollaries of our concentration inequality (Theorem 3.5.1).

Corollary 6.3.1 Let $f(n) = \log(n) + (3K + 1) \log \log(n)$. There exists $n_0$ such that for all $n \geq n_0$:

$$\mathbb{P} \left[ \sum_{k=1}^{K} t_k(n) I^+ (\hat{\theta}_k(n), \theta_k) \geq f(n) \right] \leq \frac{1}{n \log(n)}.$$  

Corollary 6.3.2 Let $f(n) = \log(n) + (3K + 1) \log \log(n)$, and define $\lambda_k^{q, k} = q - L|x_k - x_{k'}|$. Then there exists $n_0$ such that for all $n \geq n_0$:

$$\mathbb{P}[b_k(n) < \theta_k] \leq \frac{1}{n \log(n)}.$$  

Proof of Corollary 6.3.2. Since $I^+$ is increasing in its second argument, the event $b_k(n) < \theta_k$ implies that:

$$\sum_{k' = 1}^{K} t_{k'}(n) I^+ (\hat{\theta}_{k'}(n), \lambda_{k'}^{q, k}) \geq f(n).$$
Furthermore, by definition \( \lambda_{k, k'}^{\theta_k, \theta_{k'}} = \theta_k - L |x_k - x_{k'}| \leq \theta_{k'} \). Hence:

\[
\sum_{k' = 1}^{K} t_{k'}(n) I^+(\hat{\theta}_{k'}(n), \theta_{k'}) \geq f(n).
\]

We can now apply Corollary 6.3.1 and obtain:

\[
\mathbb{P}[b_k(n) < \theta_k] \leq \mathbb{P}\left[ \sum_{k' = 1}^{K} t_{k'}(n) I^+(\hat{\theta}_{k'}(n), \theta_{k'}) \geq f(n) \right] \leq \frac{1}{n \log(n)}. \quad (6.30)
\]

We then give an important lemma that allows us to upper bound the average cardinalities of particular sets of rounds. This lemma is stated and proved in [36].

**Lemma 6.3.1** Let \( k \in \mathcal{K} \), and \( \epsilon > 0 \). Define \( \mathcal{F}_n \) the \( \sigma \)-algebra generated by \((X_k(t))_{1 \leq t \leq n, 1 \leq k \leq K}\). Let \( \Lambda \subset \mathbb{N} \) be a (random) set of instants. Assume that there exists a sequence of (random) sets \((\Lambda(s))_{s \geq 1}\) such that (i) \( \Lambda \subset \bigcup_{s \geq 1} \Lambda(s) \), (ii) for all \( s \geq 1 \) and all \( n \in \Lambda(s) \), \( t_k(n) \geq \epsilon s \), (iii) \( |\Lambda(s)| \leq 1 \), and (iv) the event \( n \in \Lambda(s) \) is \( \mathcal{F}_n \)-measurable. Then for all \( \delta > 0 \):

\[
\mathbb{E}\left[ \sum_{n \geq 1} \mathbb{1}\{ n \in \Lambda, |\hat{\theta}_k(n) - \theta_k| > \delta \} \right] \leq \frac{1}{\epsilon \delta^2}. \quad (6.31)
\]

We are now ready to analyze the regret achieved under OSLB(\( \epsilon \)).

**Proof of Theorem 3.5.2.** Let \( S(\theta) \) denote the set of solutions of (4.3) for a given \( \theta \). For any \( \chi > 0 \), we define the set

\[
\Gamma_{\chi, \theta} = \bigcup_{\theta' : |\theta' - \theta_k| < \chi} S(\theta'),
\]

and for all \( k \), \( c_k^\chi = \sup\{ c_k : c \in \Gamma_{\chi, \theta} \} \). In view of Lemma 6.3.2, \( \theta' \mapsto S(\theta') \) is upper hemicontinuous at \( \theta \) and by Assumption 1 \( S(\theta) \) reduces to a point. Therefore, for any open neighbourhood \( \mathcal{B} \) of \( S(\theta) \), there exists \( \chi > 0 \) such that \( S(\theta') \subset \mathcal{B} \) if \( \sup_k |\theta_k - \theta_k'| < \chi \). Hence for all \( k \): \( c_k^\chi \to c_k(\theta) \), as \( \chi \to 0 \).

Fix \( 0 < \delta < (\theta^* - \max_{k \neq k^*} \theta_k)/2 \) and \( \epsilon > 0 \). To simplify the notation, we replace \( \epsilon \) by \( K\epsilon \) in the Theorem 3.5.2, and prove the result for this choice of \( \epsilon \).

Let \( k \) be a suboptimal arm. We derive an upper on the number of times it is played. Let \( n \) be a round where \( k \) is played, i.e., \( k(n) = k \). In view of the design of OSLB(\( \epsilon \)), there are three possible scenarios: (a) \( k \) can be the leader and its empirical reward exceeds the indexes of other arms, \( L(n) = k \) and \( \theta_k(n) \geq \max_i b_i(n) \); (b) \( k \) and \( k^* \) are not the leader, and \( k \) can be either \( k(n) \) or \( \bar{k}(n) \); (c) \( k^* \) is the leader, and again \( k \) can be either \( k(n) \) or \( \bar{k}(n) \). We investigate all cases, but we start by defining sets of rounds whose average
cardinalities can be easily controlled:

\[
A_k = \{1 \leq n \leq T : k(n) = k, b_k(n) \leq \theta_k\}
\]

\[
B_k = \{n \geq 1 : k(n) = k, \min_{k'} t_{k'}(n) \geq \epsilon t_k(n), \max_{k'} |\hat{\theta}_{k'}(n) - \theta_{k'}| \geq \delta\}
\]

\[
E_k = \{n \geq 1 : k(n) = k, |\hat{\theta}_k(n) - \theta_k| \geq \delta\}
\]

\[
F_k = \{n \geq 1 : k(n) = k, t_k(n) \leq \min(t_{k'}(n), t_{k^*}(n)), \max_{l \in \{k', k^*\}} |\hat{\theta}_l(n) - \theta_l| \geq \delta\}
\]

and \(A = \cup_k A_k, B = \cup_k B_k, E = \cup_k E_k, F = \cup_k F_k\). From the concentration inequality, and its corollaries, we have \(\mathbb{E}[|A|] \leq C_1 \log \log(T)\). We use Lemma 6.3.1 to bound the cardinalities of the other sets.

- **Bound for \(B_k\).** Let us fix \(k' \neq k\). We apply Lemma 6.3.1 to \(k'\) with \(\Lambda(s) = \{n : k(n) = k, \min_l t_l(n) \geq \epsilon s, t_k(n) = s\}\), and \(\Lambda = \cup_s \Lambda(s)\). We get that:

\[
\mathbb{E}\left[|\{n : k(n) = k, \min_l t_l(n) \geq \epsilon t_k(n), |\hat{\theta}_{k'}(n) - \theta_{k'}| \geq \delta\}|\right] \leq \frac{1}{\epsilon \delta^2}.
\]

We conclude that: \(\mathbb{E}[|B_k|] \leq K/(\epsilon \delta^2)\).

- **Bound for \(E_k\).** The application of lemma is direct here, and we get: \(\mathbb{E}[|E_k|] \leq 1/\delta^2\).

- **Bound for \(F_k\).** Using the same argument as that used to bound the cardinality of \(B_k\), we get: \(\mathbb{E}[|F_k|] \leq 2/\delta^2\).

Next we consider \(n \notin A \cup B \cup E \cup F\) such that \(k\) is played. We treat all cases (a), (b), and (c) that can arise in such a round.

**Case (a)** We assume here that \(k = L(n)\) and that \(k(n) = k\), so that \(\hat{\theta}_k(n) \geq \max_l b_l(n)\). Hence, since \(n \notin A_{k^*}\), \(\hat{\theta}_k(n) \geq b_{k^*}(n) \geq \theta^*\). In summary, \(\hat{\theta}_k(n) \geq \theta^*\), which is impossible because of our choice of \(\delta < \theta^* - \theta_k\), and \(n \notin E_k\).

**Case (b)** Let \(k' \notin \{k, k^*\}\) be the leader in round \(n\), and assume that \(k(n) = k\). We consider two subcases: (i) \(k = k(n)\), and (ii) \(k = \overline{k}(n)\).

(i) In this case, \(k\) has been played less than any other arm, and so \(t_k(n) \leq \min(t_{k'}(n), t_{k^*}(n))\). On the other hand, since \(k'\) is the leader, we have \(\hat{\theta}_{k'}(n) \geq \hat{\theta}_{k^*}(n)\), which implies that either \(\theta_{k'}\) or \(\theta_{k^*}\) is badly estimated. More precisely, we proved that \(n \in F_k\), which is impossible.

(ii) In this case, we know that \(t_k(n) \leq t_{k}(n)/\epsilon\). In addition, again, we have \(\hat{\theta}_{k'}(n) \geq \hat{\theta}_{k^*}(n)\), and so either \(\theta_{k'}\) or \(\theta_{k^*}\) is badly estimated. We proved that \(n \in B_k\), which is impossible.

**Case (c)** Assume that \(k^* = L(n)\). \(k\) is played, and we need to consider two subcases: (i) \(k = \overline{k}(n)\), and (ii) \(k = \overline{k}(n)\).

(i) In this case, since \(k = \overline{k}(n)\), we have \(t_k(n) \leq \min_l t_l(n)\), and hence \(\epsilon t_k(n) \leq \min_l t_l(n)\), and hence \(\epsilon t_k(n) \leq \min_l t_l(n)\).
6.3. Proof of Theorem 3.5.2 - OSLB Finite Time Analysis

\[ \min_l t_l(n). \] Since \( n \in B_k \), in view of the previous inequality, all arms must be well-estimated, i.e., \( \max_l |\hat{\theta}_l(n) - \theta_l| < \delta \). This implies that for all \( l \in \mathcal{K} \), \( \hat{\theta}_l(n) \leq c_l^\delta \). Now by definition in our algorithm, if \( k(n) = k = k(n) \), then \( t_k(n) < \epsilon t_{k(n)}(n) \), and so \( t_k(n) < \epsilon \max_l c_l^\delta \log(n) \). In other words, \( n \in D_k \) where

\[ D_k = \{ 1 \leq n \leq T, n \notin A \cup B \cup E \cup F, L(n) = k^*, k(n) = k, t_k(n) \leq \epsilon \max_k c_k^\delta \log(T) \}. \]

We shall bound the size of \( D_k \) later in the proof.

(ii) In this case, we must have \( t_{l(n)}(n) \geq \epsilon t_k(n) \). Hence since \( n \in B_k \), all arms are well estimated, and hence again, for all \( l \in \mathcal{K} \), \( \hat{\theta}_l(n) \leq c_l^\delta \). In particular, since \( k \) is played, \( t_k(n) \leq c_k^\delta \log(n) \), and thus \( n \in C_k \) where

\[ C_k = \{ 1 \leq n \leq T, n \notin A \cup B, k(n) = k, t_k(n) \leq c_k^\delta \log(T) \}. \]

Next we bound the expected cardinalities of \( C_k \) and \( D_k \). Since \( t_k(n) \) is incremented if \( n \in C_k \) or \( n \in D_k \), we simply have:

\[ |C_k| \leq c_k^\delta \log(T), \quad |D_k| \leq \epsilon \max_{k'} c_k^\delta \log(T). \]

Putting it all together we have proven the announced regret bound:

\[ R^\pi(T) \leq \sum_{k \neq k^*} (\theta^* - \theta_k)(\mathbb{E}[|C_k|] + \mathbb{E}[|D_k|]) \]

\[ + \mathbb{E}[|A|] + \mathbb{E}[|B|] + \mathbb{E}[|E|] + \mathbb{E}[|F|], \]

\[ \leq \log(T) \sum_{k \neq k^*} (\theta^* - \theta_k)(c_k^\delta + \epsilon \max_k c_k^\delta), \]

\[ + C_1 \log \log(T) + K^2 \epsilon^{-1} \delta^{-2} + 3K \delta^{-2}. \]

This completes the proof (because of our particular choice of \( \epsilon \), and \( \max_l c_l^\delta \leq \sum_l c_l^\delta \)). \( \square \)

### 6.3.1 Continuity of solutions to parametric linear programs

We state and prove Lemma 6.3.2, a technical result about the continuity of the solutions of a parametric linear program with respect to its parameters. It follows from the general conditions of [37].

**Lemma 6.3.2** Consider \( A \in (\mathbb{R}^+)^{K \times K} \), \( c \in (\mathbb{R}^+)^K \), and \( \mathcal{T} \subset (\mathbb{R}^+)^{K \times K} \times (\mathbb{R}^+)^K \). Define \( t = (A, c) \). Consider the function \( Q \) and the set-valued map \( Q^* \)

\[ Q(t) = \inf_{x \in \mathbb{R}^K} \{ cx | Ax \geq 1, x \geq 0 \} \]

\[ Q^*(t) = \{ x : cx \leq Q(t) | Ax \geq 1, x \geq 0 \}. \]

Assume that:
(i) For all \( t \in \mathcal{T} \), all rows and columns of \( A \) are non-identically 0

(ii) \( \min_{t \in \mathcal{T}} \min_k c_k > 0 \)

Then:

(a) \( Q \) is continuous on \( \mathcal{T} \).

(b) \( Q^* \) is upper hemicontinuous on \( \mathcal{T} \).

**Proof.** Define

\[
c_0 = \min(1, \min_{t \in \mathcal{T}} \min_k c_k) > 0, \]

and \( a = \max_{(k,k')} A_{k,k'} \). Define the sets \( \mathcal{K} = \{ x | Ax \leq 1 \} \), \( \mathcal{D} = \{ x | Ax \leq c \} \) and \( \mathcal{B} = [0, c_0 / (aK)]^\mathcal{K} \). Then \( \mathcal{B} \subset \mathcal{K} \cap \mathcal{D} \), so that both \( \mathcal{K} \) and \( \mathcal{D} \) have non-empty interior. By [37][Corollary 7], \( t \to \mathcal{K} \) and \( t \to \mathcal{D} \) are continuous on \( \mathcal{T} \) since they have non-empty interior and all rows of \( (A, 1) \) and columns of \( (a) \) are non identically 0. By [37][Theorem 2], \( Q \) is continuous on \( \mathcal{T} \) since both \( \mathcal{K} \) and \( \mathcal{D} \) are continuous on \( \mathcal{T} \), proving the first statement.

Consider a sequence \( \{(t^i, x^i)\}_{i \geq 1} \), such that \( x^i \in Q^*(t^i) \) and \( (t^i, x^i) \to (t, \bar{x}), i \to \infty \). Since for all \( i \geq 1 \) \( cx^i \leq Q(t^i) \) and \( Ax^i \geq 1 \) we have, by continuity, \( A\bar{x} \geq 1 \) and \( c\bar{x} = Q(t) \) and so \( \bar{x} \in Q^*(t) \). Hence \( Q^* \) is upper hemicontinuous. \( \square \)

### 6.4 Proof of Theorem 3.5.3 - Finite Time Analysis of POSLB

We define \( B = \{ n \in \mathbb{N} : b_{k^*}(n) < \mu_\theta(k^*) \} \), the set of rounds when the index of the optimal superarm underestimates its true value \( \mu_\theta \). Consider \( 0 < \delta < \min_k \theta^* - \theta_k \). Further, define \( D_k = \{ n : k(n) = k, b_k(n) \geq \theta^* - \delta \} \). Let \( k \neq k^* \) be a suboptimal arm and let \( n \in B \) such that \( k(n) = k \) and note that by definition of our index, \( b_k(n) > b_{k^*}^l(n) \) is equivalent to \( f(n) - f_k(n, b_{k^*}^l(n)) > f(n) - f_{L(n)}(n, b_{k^*}^l(n)) = 0 \). Furthermore, for all \( n \geq 1 \) and \( j \in \mathcal{K} \), we have \( b_j(n) \leq b_{k^*}^j(n) \). Then, if \( n \notin B \) and \( k(n) = k \), the possible events are:

(a) If \( L(n) \in \{k, k^*\} \) then \( b_k(n) \geq b_{k^*}(n) \geq \theta^* - \delta \) since \( n \notin B \), so \( n \in D_k \).

(b) If \( L(n) = k' \notin \{k, k^*\} \), then \( b_k(n) \geq b_{k^*}^l(n) \) and:

(b-i) If we further have \( b_k(n) \geq b_{k^*}^l(n) \geq \theta^* - \delta \) then \( n \in D_k \) as well.

(b-ii) Otherwise \( b_{k^*}^l(n) \leq \theta^* - \delta < b_{k^*}(n) \).

Define the random set of instants \( E_k = \{ n \notin B : k(n) = k, L(n) \notin \{k, k^*\}, \theta^* - \delta > b_{k^*}^l(n), |\hat{\theta}_{k^*} - \theta^*| > \delta \} \). Thus, in the case (b-ii) we have \( n \in E_k \). Consequently, when \( k(n) = k \) we must have that \( n \in B \cup E_k \cup D_k \) and therefore:

\[
\mathbb{E}[t_k(T)] \leq \mathbb{E}[|B|] + \mathbb{E}[|D_k|] + \mathbb{E}[|E_k|].
\]

We proceed to bound the expected cardinality of the sets \( B, D_k \) and \( E_k \).
From Theorem 2 in [12], there exists a constant $C_1 \geq 0$ such that $\mathbb{E}[\|B\|]$ is upper bounded by the Bertrand series:

$$\mathbb{E}[\|B\|] \leq \sum_{n=1}^{T} C_1 (n \log(n))^{-1} \leq C_1 \log(\log(T)).$$

Expected size of $D_k$: Define the random sets of rounds $F^\delta_k = \{ n : k(n) = k, |\hat{\theta}_k(n) - \theta_k| < \delta \}$ and $F^\delta_S = \{ n : S(n) = S, |\hat{\theta}_k(n) - \theta_k| \geq \delta \}$. Again, a direct application of Lemma 6.3.1, yields $\mathbb{E}[\|F^\delta_k\|] \leq \delta^{-2}$.

Consider $n \in D_k \cap F^\delta_k$ then, since $n \in D_k$ then $b_k(n) \geq \theta^* - \delta$ and consequently:

$$t_k(n)I(\hat{\theta}_k(n), \theta^* - \delta) \leq \sum_{j \in K} t_j(n)I(\hat{\theta}_j(n), \lambda_j^{k,\theta^* - \delta}) \leq f(n)$$

Since $|\hat{\theta}_k(n) - \theta_k| < \delta$ we further have:

$$t_k(n) \leq \frac{f(n)}{I(\theta_k + \delta, \theta^* - \delta)}$$

Thus we obtain:

$$\mathbb{E}[\|D_k\|] \leq \frac{f(n)}{I(\theta_k + \delta, \theta^* - \delta)} + \delta^{-2}.$$  

Expected size of $E_k$:

Consider $n \in E_k \cap F^\delta_k$. Since $n \in E_k$ and therefore $k(n) = k$ and $k' = L(n)$, then:

$$t_k(n)I(\hat{\theta}_k(n), \lambda_k^{k,b_{k'}^l(n)}) \leq \sum_{j \in K} t_j(n)I(\hat{\theta}_j(n), \lambda_j^{k,b_{k'}^l(n)}) \leq \sum_{j \in K} t_j(n)I(\hat{\theta}_j(n), \lambda_j^{k,\theta^* - \delta})$$

Furthermore, by our definition of $\lambda^{k^*,q}$, and since $b_{k'}^l(n) \leq \theta^* - \delta$ we have:

$$\sum_{j \in K} t_j(n)I(\hat{\theta}_j(n), \lambda_j^{k^*,b_{k'}^l(n)}) \leq \sum_{j \in K} t_j(n)I(\hat{\theta}_j(n), \theta_j)$$

We further distinguish two cases:

1) $\theta_k > \theta_{k'}$

2) $\theta_k < \theta_{k'}$

1) In this case, by our choice of $\delta$ and since $L(n) = k'$ we must have that $|\theta_{k'} - \hat{\theta}_{k'}(n)| \geq \delta$. Since $S(n) = k$ we have:

$$t_k(n)I(\hat{\theta}_k(n), b_{k}^l(n)) \leq t_{k'}I(\hat{\theta}_{k'}(n), b_{k'}^l(n))$$

then, since $L(n) = k'$, we have $\hat{\theta}_k(n) < \hat{\theta}_{k'}(n)$, and therefore $t_k(n) \leq \beta t_{k'}(n)$. As previously, applying Lemma 6.3.1 results in $\mathbb{E}[E_k] \leq \delta^{-2}$. 

6.4. Proof of Theorem 3.5.3 - Finite Time Analysis of POSLB

Expected size of $B$: From Theorem 2 in [12], there exists a constant $C_1 \geq 0$ such that $\mathbb{E}[\|B\|]$ is upper bounded by the Bertrand series:
2) In this case we have \( b_{k'}(n) \geq \theta_{k'} \) and therefore \( |\hat{\theta}_k(n) - b_{k'}(n)| \geq \Delta \). Therefore:

\[
I(\hat{\theta}_k(n), b_{k'}(n)) \geq I(\theta_k + \delta, \theta_{k'}) > 0.
\]

and consequently:

\[
E[|E_k|] \leq \mathbb{E}[\sum_{j \in K} t_j(n)I(\hat{\theta}_j(n), \theta_j)]/I(\theta_k + \delta, \theta_{k'}).
\]

From Equations (6.32) and Lemma 6.4.1, denoting by \( \Gamma(x, y) \) the upper incomplete gamma function, we obtain that \( \exists n_0 < \infty \) such that for all \( n \geq n_0 \):

\[
\mathbb{E}[|E_k|] \leq \frac{2K \log \log(n) + \Gamma(2K + 2, 2K \log \log(n)) \left( \frac{\log(n)}{K} \right)^{K+1} e^{K+1}}{I(\theta_k + \delta, \theta_{k'})} + \delta^{-2}.
\]

For convenience, we will use the following notation:

\[
Y(K, n) = \Gamma(2K + 2, 2K \log \log(n)) \left( \frac{\log(n)}{K} \right)^{K+1} e^{K+1}.
\]

Putting everything together we get \( \exists C_1, C_2 < \infty \) and \( \exists n_0 < \infty \) such that for all \( n \geq n_0 \):

\[
\mathbb{E}[t_k(T)] \leq \frac{f(n)}{I(\theta_k + \delta, \theta^* - \delta)} + C_1 \log \log(T) + \delta^{-2}
+ C_2 Y(K, n)
\]

and we point out that \( Y(K, n) \to 0 \) when \( n \to \infty \) since we have:

\[
\lim_{y \to \infty} \frac{\Gamma(x, y)}{y^{-1} e^{-x}} = 1
\]

and therefore, replacing \( x = 2K + 2 \) and \( y = 2K \log \log(n) \):

\[
\lim_{n \to \infty} \frac{\Gamma(2K + 2, 2K \log \log(n))}{\log \log(n)^{2K+1} e^{-2K \log \log(n)}} = \\
\lim_{n \to \infty} \Gamma(2K + 2, 2K \log \log(n)) \times \frac{\log(n)^{2K}}{\log \log(n)^{2K+1}} = 1.
\]

which concludes the proof. \(\square\)

**Lemma 6.4.1** There exists a time \( n_0 < \infty \) such that \( \forall n \in \mathbb{N}, n \geq n_0 \), we have:

\[
\mathbb{E}[\sum_{k \in K} t_k(n)I(\hat{\theta}_k(n), \theta_k)] \leq 2K \log \log(n) +
+ \Gamma(2K + 2, 2K \log \log(n)) \times \left( \frac{\log(n)}{K} \right)^{K+1} e^{K+1}
\]
Proof. From Theorem 2 in [12] we have $\forall n \in \mathbb{N}$ and $\forall \delta \geq K + 1$:

$$\mathbb{P} \left[ \sum_{k \in K} \tau_k(n) I(\hat{\theta}_k(n), \theta_k) \geq \delta \right] \leq e^{-\delta} \left( \frac{\log(n) \delta}{K} \right)^K e^{K+1}$$

and since $\tau_k(n) I(\hat{\theta}_k(n), \theta_k) \geq 0$ for all $n \in \mathbb{N}$ and $k \in K$ we have:

$$\mathbb{E} \left[ \sum_{k \in K} \tau_k(n) I(\hat{\theta}_k(n), \theta_k) \right] = \int_0^\infty \mathbb{P} \left[ \sum_{k \in K} \tau_k(n) I(\hat{\theta}_k(n), \theta_k) \geq \delta \right] d\delta \leq 2K \log \log(n) + \int_{2K \log \log(n)}^\infty e^{-\delta} \left( \frac{\log(n) \delta}{K} \right)^K e^{K+1} d\delta$$

Note that there exists $n_0 < \infty$ such that $\forall n > n_0$ and $\delta \geq 2K \log \log(n)$ we can bound the previous expression as follows:

$$\left( \frac{\log(n) \delta}{K} \right)^K \leq \left( \frac{(\log(n) + 1) \delta}{K} \right)^K < \left( \frac{\delta^2 \log(n)}{K} \right)^{K+1}$$

Consequently, $\exists n_0 < \infty$ such that $\forall n \geq n_0$ we have:

$$\mathbb{E} \left[ \sum_{k \in K} \tau_k(n) I(\hat{\theta}_k(n), \theta_k) \right] \leq 2K \log \log(n) + \Gamma(2K + 2, 2K \log \log(n)) \times \left( \frac{\log(n)}{K} \right)^{K+1} e^{K+1}$$

with $\Gamma(x, y) = \int_y^\infty t^{x-1} e^{-t} dt$, the upper incomplete gamma function.

Throughout the asymptotic analysis of POSB, we will make use of the following lemma:

Lemma 6.4.2 Under POSLB, we have:

$$b_{kl}^{k*}(n) \rightarrow \theta^* \text{ a.s.}$$

Proof. Since $\lim \sup_{n \rightarrow \infty} t_k(n)/f(n) < \infty$ for all $k \neq k^*$, we have that

$$\lim_{n \rightarrow \infty} \inf_{n \rightarrow \infty} t_{k^*}(n)/f(n) = \infty \text{ a.s.}$$

and by definition:

$$t_{k^*}(n) I(\hat{\theta}_{k^*}(n), b_{k^*}^{k^*}(n)) = f(n).$$

Thus, we have that $I(\hat{\theta}_{k^*}(n), b_{k^*}^{k^*}(n)) \rightarrow 0$ a.s. and therefore $b_{k^*}^{k^*}(n) \rightarrow \theta^*$ a.s. since $\hat{\theta}_{k^*}(n) \rightarrow \theta_{k^*} \text{ a.s.}$
Lemma 6.4.3  Under POSLB, \( \exists n_0 < \infty \) such that \( \forall n \geq n_0 \) we have \( L(n) = k^* \).

Since from the LLN we have that \( \hat{\theta}_k(n) \rightarrow \theta_k \) when \( n \rightarrow \infty \), we can immediately conclude that \( \forall k \in \mathcal{K}^- \) such that we have \( t_k(n) \rightarrow \infty \) when \( n \rightarrow \infty \), \( \exists n_0(k) < \infty \) such that \( \forall n \geq n_0(k) \) we have \( L(n) \neq k \) as \( \hat{\theta}_k(n) \leq \hat{\theta}_{k^*}(n) \). Note that \( t_{k^*}(n) \rightarrow \infty \) when \( n \rightarrow \infty \), since from Theorem 3.5.3 we have that \( k^* \) is selected in all but \( O(\log(n)) \) rounds.

To conclude the proof, it is sufficient to show for all \( k \in \mathcal{K}^+ \cup \{k^*\} \exists n_0(k) \) such that \( \forall n \geq n_0(k) \) we have \( L(n) \neq k \). Consider one such arm \( k \) and therefore \( \exists n'(k) < \infty \) such that \( \forall n \geq n'(k) \) we have \( t_k(n) = t_k(n'(k)) \). We know such an \( n'(k) \) exists since \( k \notin \mathcal{K}^+ \) is played finitely many times.

Assume \( \exists n \geq \max\{n'(j) | j \notin \mathcal{K}^+, j \neq k^*\} \) such that \( \forall j \in \mathcal{K}^+, t_j(n) > t_k(n) \) and \( L(n) = k \). From our choice of \( n \) we have that \( k(n) \in \mathcal{K}^+ \) and therefore:

\[
t_k(n)I(\hat{\theta}_k(n), b_k^{kl}(n)) \geq t_k(n)(n)I(\hat{\theta}_{k(n)}(n), b_k^{kl}(n)).
\]

A contradiction, since \( t_k(n) < t_{k(n)}(n) \), by our choice of \( n \) and, \( I(\hat{\theta}_k(n), b_k^{kl}(n)) \leq I(\hat{\theta}_{k(n)}(n), b_k^{kl}(n)) \) since \( L(n) = k \), which concludes the proof.

\[\Box\]

6.5 Proof of Theorem 3.5.4 - POSLB Asymptotic Regret for Lipschitz Bandits

Without loss of generality, assume \( \mathcal{K} \subset [0, 1] \). We restate our definition of the most confusing parameter \( \lambda^{q(n),k}(n) \), for an arm \( k \) and a level \( q(n) \):

\[
\lambda^{q(n),k}(n) = \arg\min_{\lambda \in \Theta_L; \lambda \geq q(n)} \sum_{j \in \mathcal{K}} t_k(n)I(\hat{\theta}_j(n), \lambda_{j}^{q(n),k})
\]

Throughout this proof we will use the short-hand notation \( \lambda^{q(n),k} = \lambda^{q(n),k}(n) \).

Now define:

\[
f_k(n) = \sum_{j \in \mathcal{K}} t_j(n)I(\hat{\theta}_j(n), \lambda_{j}^{b_{k^*}(n),k}).
\]

The proof of the second statement uses a dominated convergence argument and a sample path analysis. Since we proved Theorem 3.5.3 above and will later show there exists a round \( n_0 < \infty \) a.s. such that \( \forall n \geq n_0 \) we have \( L(n) = k^* \) (see Lemma 6.4.3), to conclude the proof it suffices to prove that for all \( k \neq k^* \):

\[
\lim_{n \rightarrow \infty} \sup_{\forall n} \frac{f_k(n)}{f(n)} = 1 \text{ a.s.}
\]

Note: At this point, for a better and faster understanding of the main arguments of this proof, we advise the reader to first follow the asymptotic analysis of the behavior of POSB in the deterministic setting (i.e. \( \hat{\theta}_k(n) = \theta_k \) and \( b_{k^*}(n) = 0^* \) for all \( n \in \mathbb{N} \) and \( k \in \mathcal{K} \), presented in Subsection 6.5.1 of the Appendix. This proof follows the same ideas.
while taking into account for the noise in estimates \( \hat{\theta}(n) \) and \( \lambda^{q(n),k} \), and is therefore, significantly more opaque.

We consider a fixed sample path throughout the proof and we define \( \mathcal{K}^+ = \{ k \in \mathcal{K} \setminus \{ k^* \} : \lim_{n \to \infty} t_k(n) \to \infty \} \). We proceed by proving Equation 6.33 holds for all arms in \( \mathcal{K}^+ \), then prove that \( \mathcal{K}^+ = \mathcal{K} \) a.s.

We denote by \( j_1(n) \) the last arm to be played before time \( n \), \( j_2(n) \) the 2nd to last and so on. Define \( n^1_0(n) \) as the round when arm \( j_i(n) \) was last played before \( n \). For simplicity, we refer to \( n^i_0(n) \) as \( n^i_0 \) and \( j_i(n) \) as \( j_i \).

We prove by induction on \( i \) that

\[
\lim_{n \to \infty} \sup_{n} f_{j_i}(n)/f(n) = 1 \text{ a.s.} \tag{6.34}
\]

**Proof of (6.34), Step 1:**

We prove:

\[
\lim_{n \to \infty} \sup_{n} \frac{f_{j_2}(n)}{f(n)} = \lim_{n \to \infty} \sup_{n} \frac{f_{j_1}(n)}{f(n)} = 1.
\]

Since by definition \( k(n^2_0) = j_2 \) we have \( f_{j_2}(n^2_0) \leq f_{j_1}(n^2_0) \). Define:

\[
d = t_{j_1}(n^1_0) - t_{j_1}(n^2_0) = t_{j_1}(n) - t_{j_1}(n^2_0).
\]

Now let us consider the round \( n^1_0 \). Remembering that \( \lambda^{q,i}_i = q \) and using the notation \( q(n) = b_{L(n)}(n) \), then:

\[
f_{j_1}(n^1_0) = f_{j_1}(n^2_0) + d I(\hat{\theta}_{j_1}(n^2_0), q(n^2_0)) + I(\hat{\theta}_{j_2}(n^2_0), \lambda^{q(n^2_0),j_1}_{j_2}) - \sum_{i \in \mathcal{K}^-} t_i(n^1_0)[I(\hat{\theta}_{i}(n^2_0), \lambda^{q(n^2_0),j_1}_i) - I(\hat{\theta}_{i}(n^1_0), \lambda^{q(n^1_0),j_1}_i)] \tag{6.35}
\]

and

\[
f_{j_2}(n^1_0) = f_{j_2}(n^2_0) + d I(\hat{\theta}_{j_1}(n^2_0), \lambda^{q(n^2_0),j_1}_{j_2}) + I(\hat{\theta}_{j_2}(n^2_0), q(n^2_0)) - \sum_{i \in \mathcal{K}^-} t_i(n^1_0)[I(\hat{\theta}_{i}(n^2_0), \lambda^{q(n^2_0),j_1}_i) - I(\hat{\theta}_{i}(n^1_0), \lambda^{q(n^1_0),j_1}_i)]. \tag{6.36}
\]

Consider \( 0 < \varepsilon < \min_j |\theta^* - \theta_j|/2 \) and a round \( n \) such that \( n^2_0(n) > n(\varepsilon) \), with \( n(\varepsilon) < \infty \) such that for all \( k \in \mathcal{K}^+ \) and \( n' > n(\varepsilon) \) we have \( |\hat{\theta}_k(n') - \theta_k| \leq \varepsilon \) and \( |b_{k'}(n') - \theta^*| < \varepsilon \). From the L.L.N. and Lemma 6.4.2 we know that such a time exists. Note that at round \( n^2_0 \) we have \( k(n) = j_2 \) and therefore \( f_{j_1}(n^2_0) \geq f_{j_2}(n^2_0) \).

Define:

\[
A_1(n^1_0) = \sum_{i \in \mathcal{K}^+} t_i(n^1_0)[I(\theta_i + \varepsilon, \lambda^{\theta^*+\varepsilon,k}_i) - I(\theta_i - \varepsilon, \lambda^{\theta^*+\varepsilon,k}_i)] + \sum_{i \notin \mathcal{K}^+, i \neq k^*} t_i(n)[I(\hat{\theta}_i(n), \lambda^{b_{k^*}(n),k}_i) \tag{6.37}
\]

Note that at round \( n^1_0 \) we have \( k(n) = j_1 \) and therefore we have:

\[
\lim_{n \to \infty} \sup_{n} f_{j_1}(n)/f(n) = 1 \text{ a.s.} \tag{6.38}
\]
and
\[ A_2(n_0^1, j_2) = \sum_{i \in K^+} t_i(n_0^1)[I(\theta_i - \varepsilon, \lambda_i^{\theta_i^* + \varepsilon, k}) - I(\theta_i + \varepsilon, \lambda_i^{\theta_i^*-\varepsilon, k})] + \sum_{i \notin K^+, i \neq k^*} t_i(n)I(\hat{\theta}_i(n), \lambda_i^{b_{k^*}(n), k}) \] (6.39)

Thus, from Equations (6.35) and (6.36):
\[ f_{j_1}(n_0^1) - f_{j_2}(n_0^1) \geq 0 + d[I(\hat{\theta}_{j_1}(n_0^2), q(n_0^2)) - I(\hat{\theta}_{j_1}(n_0^2), \lambda_{j_1}^{q(n_0^2), j_2})] + \\
+ [I(\hat{\theta}_{j_2}(n_0^2), \lambda_{j_2}^{q(n_0^2), j_1}) - I(\hat{\theta}_{j_2}(n_0^2), q(n_0^2))] - A_1(n_0^1, j_1) + A_2(n_0^1, j_2). \]

We then obtain that for all \( n \geq n(\varepsilon) \) we have:
\[ \limsup_{n \to \infty} \frac{0 \geq f_{j_1}(n_0^1) - f_{j_2}(n_0^1)}{f(n_0^1)} \geq \left[ I(\theta_{j_1} + \varepsilon, \theta^* - \varepsilon) - I(\theta_{j_1} - \varepsilon, \lambda_{j_2}^{\theta_{j_2}^* + \varepsilon, j_2}) \right] \frac{d}{f(n_0^1)} \]
(6.41)
\[ + \left[ I(\theta_{j_2} + \varepsilon, \lambda_{j_2}^{\theta_{j_2}^*-\varepsilon, j_1}) - I(\theta_{j_2} - \varepsilon, \theta^* + \varepsilon) \right] \frac{1}{f(n_0^1)} \]
\[ - \frac{A_1(n_0^1, j_1) - A_2(n_0^1, j_2)}{f(n_0^1)} = \frac{B}{f(n_0^1)} \]
with the first inequality is due to \( k(n_0^1) = j_1 \).

Since for all \( k \in K^- \), \( \limsup_{n \to \infty} t_k(n)/f(n) < \infty \) then \( \exists C < \infty \) such that
\[ \limsup_{n \to \infty} \frac{t_{j_1}(n_0^1)/f(n_0^1)}{f(n_0^1)} \leq C \] (one can easily show \( C = \max_i 1/I(\theta_i, \theta^*) \), (6.42)
where the last equality is due to Theorem 3.5.3. From Equations (6.42), (6.37) and (6.39) we have that for \( i \in \{1, 2\} \):
\[ \limsup_{\varepsilon \to 0} \limsup_{n \to \infty} A_i(n_0^1, j_i)/f(n_0^1) = 0. \] (6.43)

From (6.41) and (6.43):
\[ \limsup_{n \to \infty} B = \lim_{\varepsilon \to 0} \left[ I(\theta_{j_1} + \varepsilon, \theta^* - \varepsilon) - I(\theta_{j_1} - \varepsilon, \lambda_{j_2}^{\theta_{j_2}^* + \varepsilon, j_2}) \right] \limsup_{n \to \infty} \frac{d}{f(n_0^1)} \]
and consequently:
\[ 0 \geq \limsup_{n \to \infty} \left[ I(\theta_{j_1}, \theta^*) - I(\theta_{j_1}, \lambda_{j_1}^{\theta^*, j_2}) \right] \frac{d}{f(n_0^1)} \]

Since \( q(n) > \hat{\theta}_{L(n)}(n) \geq \hat{\theta}_{k'}(n) \), for all \( k' \in K \) and from the monotonicity of the KL divergence, we have that for all \( k \):
\[ \lambda_k^{q(n), k} = q(n) \text{ and } \lambda_{j}^{q(n), k} < q(n), \forall j \neq k. \]
Consequently \( I(\theta_{j_1}, \theta^*) - I(\theta_{j_1}, \lambda_{j_1}^{\theta_{j_2}}) > 0 \), and we immediately obtain:

\[
\lim_{n \to \infty} \sup \frac{d}{f(n_0^1)} = \lim_{n \to \infty} \sup \frac{t_{j_1}(n_0^1) - t_{j_2}(n_0^2)}{f(n_0^1)} = 0 \quad (6.45)
\]

From (6.41), we obtain:

\[
\frac{f_{j_2}(n_0^1)}{f(n_0^1)} \geq \frac{f_{j_1}(n_0^1)}{f(n_0^1)} \geq \frac{f_{j_2}(n_0^1)}{f(n_0^1)} + B
\]

Letting \( \varepsilon \to 0 \) and \( n \to \infty \) (\( B \to 0 \) when \( \varepsilon \to 0 \) and \( n \to \infty \)) we obtain:

\[
\lim_{n \to \infty} \sup \frac{f_{j_2}(n)}{f(n)} = \lim_{n \to \infty} \sup \frac{f_{j_1}(n)}{f(n)} = 1
\]

where the last inequality is due to arm \( j_1 \) being chosen at time \( n_0^1 \). Since no suboptimal arm is played between \( n_0^1 \) and \( n \), then for all \( i \in K \), \( \lim_{n \to \infty} f_i(n_0^1) = \lim_{n \to \infty} f_i(n) \) and therefore:

\[
\lim_{n \to \infty} \sup \frac{f_{j_2}(n)}{f(n)} = \lim_{n \to \infty} \frac{f_{j_1}(n)}{f(n)} = 1
\]

**Proof of (6.34), Step 2:**

Consider the arm \( k = j_y, y \in \mathbb{N} \) and \( 0 < y < |K|^+ \) and assume that for all \( i, h < y \) we have:

\[
\lim_{n \to \infty} \sup \frac{f_i(n)}{f(n)} = \lim_{n \to \infty} \frac{f_h(n)}{f(n)}
\]

To complete the proof is suffices to prove that under this assumption we obtain:

\[
\lim_{n \to \infty} \sup \frac{f_k(n)}{f(n)} = \lim_{n \to \infty} \frac{f_{j_1}(n)}{f(n)} = 1
\]

By our assumption for all \( \alpha \in (0, 1) \) there exists \( n(\alpha) \) such that for all \( n' > n(\alpha) \) and for all \( i, h < y \):

\[
\frac{f_{j_2}(n')(1 + \alpha)}{f(n')} \geq \frac{f_{j_2}(n')}{f(n')}
\]

Consider \( \alpha \in (0, 1) \) and a time \( n \) such that \( n_0^{|K|^+}(n) > \max(n(\alpha), n(\varepsilon)) \) and \( k(n) = k^* \). Since

\[
\lim_{n \to \infty} t_i(n) \to \infty, \text{ for all } i \in K^+ \cup \{k^*\} \text{ such an } n \text{ always exists. Define } k_1 = \max\{j_i : j_i \leq j_y, i < y\} \text{ and } k_2 = \min\{j_i : j_i \geq j_y, i < y\} \text{ (from Step 1, we know } k_1 \neq \emptyset \neq k_2 \text{) as the closest arms to the left and right of } k \text{ that have been played after arm } k \text{ (or } k \text{ if such an arm doesn’t exist).}
\]

We define:

\[
a_1(n, x) = \sum_{i \in K^+, i \leq k_1} (t_i(n) - t_i(n_0^1)) I(\hat{\theta}_x(n), \lambda_{b_{k^*}(n), i}^{b_i(n)} I[\theta_x < b_{k^*}(n) - L|x - i|]
\]

\[
a_1(n, x) = \sum_{i \leq k_1} t_i(n) - t_i(n_0^1)
\]
\[ a_2(n, x) = \frac{\sum_{i \in K^+, i \geq k_2} (t_i(n) - t_i(n_0^k)) I(\hat{\theta}_x(n), \lambda_x^{b_k^*(n), i}) \mathbb{I}[\theta_x < b_k^*(n) - L|x - i|]}{\sum_{i \geq k_2} t_i(n) - t_i(n_0^k)} \]

Define \( I^+(x, y) = I(x, y) \) if \( x < y \) and 0 otherwise. Then note that \( \forall i \in K, i < k_1 \) and for all \( n \) and \( q \geq \hat{\theta}_i(n) + L|k_1 - i| \) we have that:

\[ \frac{(k_2 - k)I^+(\hat{\theta}_i(n), q - L|k_1 - i|) + (k - k_1)I^+(\hat{\theta}_i, q - L|k_2 - i|)}{k_1 - k_2} \]

The above comes from the observation that the KL divergence \( I(x, y) \) is convex in \( y \) over the interval \((x, 1)\), for all \( x \in (0, 1) \) and therefore \( I(x, \frac{a y_1 + b y_2}{a + b}) < \frac{b I(x, y_1) + a I(x, y_2)}{a + b} \) for any \( a, b \in (0, 1) \), \( a \neq b \), and \( y_1, y_2, x \in (0, 1) \), \( y_1, y_2 > x \), \( y_1 \neq y_2 \).

Consequently, in light of Lemma 6.5.1, \( \exists n_0 \) such that:

\[ \frac{a_1(n, k_1)(k_2 - k) + a_1(n, k_2)(k - k_1)}{k_2 - k_1} > a_1(n, k), \; \forall n > n_0. \]  

(6.47)

Similarly:

\[ \frac{a_2(n, k_1)(k_2 - k) + a_2(n, k_2)(k - k_1)}{k_2 - k_1} > a_2(n, k), \; \forall n > n_0. \]  

(6.48)

Define \( d_1 = \sum_{i \leq k_1, i \neq k^*} t_i(n) - t_i(n_0^k) \) and \( d_2 = \sum_{i \geq k_2, i \neq k^*} t_i(n) - t_i(n_0^k) \), the number of times arms to the left and right, respectively, of \( k \) were played since the last play of arm \( k \).

Consider \( 0 < \varepsilon < \min_j |\theta^* - \theta_j|/2 \) and a round \( n \) such that \( n_0^{\lceil K^+ \rceil}(n) > \max\{n_0(n(\varepsilon)), n(\alpha)\} \), with \( n(\varepsilon) \) such that for all \( i \in K^+ \) and \( n' > n(\varepsilon) \) we have \( |\hat{\theta}_i(n') - \theta_i| \leq \varepsilon \) and \( |b_k^*(n') - \theta^*| < \varepsilon \). From the L.L.N. and Lemma 6.4.2 we know that such a time exists.

Define:

\[ E_1(n, x) = \frac{\sum_{i \in K^+, i \leq k_1} (t_i(n) - t_i(n_0^k)) I(\hat{\theta}_x(n), \lambda_x^{b_k^*(n), i}) \mathbb{I}[\theta_x \geq b_k^*(n) - L|x - i|]}{\sum_{i \leq k_1} t_i(n) - t_i(n_0^k)} \]

\[ E_2(n, x) = \frac{\sum_{i \in K^+, i \geq k_2} (t_i(n) - t_i(n_0^k)) I(\hat{\theta}_x(n), \lambda_x^{b_k^*(n), i}) \mathbb{I}[\theta_x \geq b_k^*(n) - L|x - i|]}{\sum_{i \geq k_2} t_i(n) - t_i(n_0^k)} \]

and \( A'_2(n, k) = A_2(n, k) + d_1 E_1(n, k) + d_2 E_2(n, k) \). Note that, since

\[ \lim_{n \to \infty} \sup_{t_k(n)/f(n)} < \infty \text{ (from Theorem 3.5.3)} \]
6.5. Proof of Theorem 3.5.4 - POSLB Asymptotic Regret for Lipschitz Bandits

and Lemma 6.5.1, we have that \( E_i(n, x) / f(n) \to 0 \) when \( \varepsilon \to 0 \), and from Equation (6.43), we immediately obtain:

\[
\lim_{\varepsilon \to 0} \limsup_{n \to \infty} A_i(n^1_0, j_i) / f(n^1_0) = 0. \tag{6.49}
\]

Then \( \forall n \geq n(\varepsilon) \) we have that:

\[
f_k(n) \leq f_k(n^0_0) + d_1 a_1(n, k) + d_2 a_2(n, k) + A'_2(n, k) + I(\hat{\theta}_k(n), b_k^*(n))
\leq \frac{f_k(n^0_0)(k_2 - k) + f_k(n^0_0)(k - k_1)}{k_2 - k_1} + d_1 a_1(n, k) + d_2 a_2(n, k) + A'_2(n, k) + I(\hat{\theta}_k(n), b_k^*(n)) +
\]

\[
d_1 \frac{a_1(n, k_1)(k_2 - k) + a_1(n, k_2)(k - k_1)}{k_2 - k_1} + d_2 \frac{a_2(n, k_1)(k - k_1) + a_2(n, k_2)(k_2 - k)}{k_2 - k_1} + A'_2(n, k) + I(\theta_k - \varepsilon, \theta^* + \varepsilon).
\]

where the second inequality is true since by definition \( f_k(n^0_0) \leq f_i(n^0_i), \forall i \in K \) (because \( k(n^0_i) = k \)), and the third inequality is due to equations (6.61) and (6.62) (which in turn stem from the convexity of the KL divergence).

Then, dividing by \( f(n) \), we obtain:

\[
\frac{f_k(n)}{f(n)} \leq \frac{f_k(n)(k_2 - k_1) + f_k(n)(k - k_1)}{f(n)(k_2 - k_1)} + \frac{A'_2(n, k)}{f(n)}
\leq \frac{f_k(n)(k_2 - k_1) + \alpha(k_2 - k)f_k(n)}{f(n)(k_2 - k_1)} + \frac{A'_2(n, k)}{f(n)}
\]

Where the last inequality comes from our assumption in Equation (6.60). Since \( k(n) = k^* \), and from Equation (6.63), taking letting \( n \to \infty \) then \( \varepsilon \to 0 \), we obtain:

\[
1 \leq \limsup_{n \to \infty} \frac{f_k(n)}{f(n)} \leq \limsup_{n \to \infty} \frac{f_k(n)}{f(n)} \left( 1 + \frac{\alpha(k_2 - k)}{k_2 - k_1} \right) = 1 + \frac{\alpha(k_2 - k)}{k_2 - k_1}.
\]

Letting \( \alpha \to 0 \) we obtain the advertised result.

Note that this result implies that \( K^+ = K \) and \( \limsup_{n \to \infty} t_k(n)/f(n) > 0 \) for all \( k \in K \), - see Corollary 6.5.1.

**Corollary 6.5.1** If for all \( k \in K^+ \), \( \lim_{n \to \infty} f_k(n)/f(n) = 1 \) a.s., then \( K^+ = K \).

**Proof.** Consider a fixed sample path. Let us assume there exists \( k \in K^- \) such that \( \limsup_{n \to \infty} t_k(n)/f(n) = 0 \) and define \( J = \{ k \in K^- : \limsup_{n \to \infty} t_k(n)/f(n) > 0 \} \).
Let \( j_1 = \max\{j \in J : j < k\} \) and \( j_2 = \min\{j \in J : j > k\} \). If \( j_1 = \emptyset \), then \( j_1 = k \) and if \( j_2 = \emptyset \), then \( j_2 = k \). Since \( f_j(n)/f(n) \to 1 \) when \( n \to \infty \) for all \( j \in \mathcal{K}^+ \), we know that we cannot have \( j_1 = j_2 = k \) as this would imply \( f_j(n)/f(n) \to 0 \) for all \( j \in \mathcal{K}^- \) and consequently \( b_j(n) \to 1 \), for all \( j \in \mathcal{K}^- \).

Now, by our definition of \( J \), for all \( \varepsilon > 0 \) there exists \( n_0(\varepsilon) \) such that for all \( n \geq n_0(\varepsilon) \):

\[
\frac{f_k(n)}{f(n)} < \frac{\sum_{j \in J} t_j(n)I(\hat{\theta}_j(n), \lambda_j^{q(n), k})}{f(n)} + K\varepsilon. \tag{6.50}
\]

and

\[
1 + \varepsilon > \frac{f_{j_1}(n)}{f(n)} \quad \text{and} \quad 1 + \varepsilon \geq \frac{f_{j_2}(n)}{f(n)}. \tag{6.51}
\]

From Lemma 6.5.1 we have that \( \exists \Delta \) (with \( \Delta \) a constant depending on \( \theta \) and \( \mathcal{K} \)) such that for all \( \delta \in (0, \Delta) \) there exists \( n_0'(\delta) \) such that \( \forall n > n_0'(\delta) \) we have:

\[
I(\hat{\theta}_j(n), \lambda_j^{q(n), j_1})(j_2 - k) + I(\hat{\theta}_j(n), \lambda_j^{q(n), j_1})(k - j_1) \quad > \quad I(\hat{\theta}_j(n), \lambda_j^{q(n), k}) + \delta, \quad \forall j \in J. \tag{6.52}
\]

Then, considering some \( n \geq \max\{n_0, n_0(\varepsilon)\} \) such that \( k(n) = k^* \) (this is legitimate, in light of Lemmas 6.4.3 and 6.5.3) from Equations (6.50) and (6.52) we obtain:

\[
\frac{f_k(n)}{f(n)} < \frac{f_{j_1}(n)(j_2 - k)}{f(n)(j_2 - j_1)} + \frac{f_{j_2}(n)(k - j_1)}{f(n)(j_2 - j_1)} + K\varepsilon - \sum_{j \in \mathcal{K}^-} \frac{t_j(n)\delta}{f(n)}
\]

which from (6.51) further becomes:

\[
\frac{f_k(n)}{f(n)} < 1 + (K + 1)\varepsilon - \frac{\sum_{j \in \mathcal{K}^-} t_j(n)\delta}{f(n)}
\]

Letting \( \varepsilon \to 0 \) we obtain:

\[
\lim_{\varepsilon \to 0} f_k(n)/f(n) < 1 - \frac{\sum_{j \in \mathcal{K}^-} t_j(n)\delta}{f(n)}
\]

and consequently \( b_k(n) > q(n) \) meaning \( k(n) \neq k^* \), a contradiction. Hence \( k \in J \), and therefore \( \forall k \in \mathcal{K} \) we have \( \lim_{n \to \infty} t_k(n)/f(n) > 0 \) and therefore \( k \in \mathcal{K}^+ \), which concludes the proof.

**Lemma 6.5.1** *In the Lipschitz bandit setting, \( \forall k \in \mathcal{K}^-, j \in \mathcal{K}^+ \). \( \lim_{n \to \infty} \lambda_j^{q(n), k} = \max\{\theta_j, \theta^* - L|j - k|\} \) a.s.*
6.5. Proof of Theorem 3.5.4 - POSLB Asymptotic Regret for Lipschitz Bandits

**Proof.** Throughout this proof we consider a fixed sample path. We use the following shorthand notation \( q(n) = b_{L(n)}(n) \). Fix the arm \( k \in K^- \). Define \( \Delta = \min \{ |\theta_k - L|i - k| - \theta_i| : i \in K^+ \text{ and } \theta_k - L|i - k| - \theta_i \neq 0 \} \). From Lemmas 6.4.2 and 6.4.3 and from the LLN we have that \( \forall \varepsilon \in (0, \Delta/2), \exists n(\varepsilon) \text{ such that } \forall n > n(\varepsilon) \text{ we have } L(n) = k^*, |\hat{\theta}_i(n) - \theta_i| \leq \varepsilon, \forall i \in K^+ \text{ and } |q(n) - \theta^*| < \varepsilon \text{ and } k(n) \in K^+ \).

Consider the rounds \( n \geq \max \{ n(\varepsilon), n_0 \} \) and the arms \( j \in K \) such that \( \hat{\theta}_j(n) \geq q(n) - L|k - j| \). By our choice of \( \varepsilon \) and \( n \), \( \forall j \in K^+, \theta_j < \theta^* - |j - k|L \) iff \( \hat{\theta}_j(n) < q(n) - |j - k|L \). We define the sets of arms:

\[
J_+(n) = \{ j \in K^+ : \theta_j \geq q(n) - L|k - j| \}
\]

and

\[
J_-(n) = \{ j \in K^+ : \theta_j < q(n) - L|k - j| \}
\]

And consider \( \lambda^*(n) \) such that \( \lambda^*_j(n) = \theta_j \) for all \( j \in J_+(n) \) and \( \lambda^*_j(n) = q(n) - L|j - k| \) for all \( j \in J_-(n) \). Observe that \( \lambda^* \in \Theta \) and considering \( \gamma^{q(n),k} \) be as defined in Lemma 6.5.2, we have:

\[
\sum_{j \in J_+(n)} t_j(n)I(\hat{\theta}_j(n), \gamma_j^{q(n),k}) + \sum_{j \in J_-(n)} t_j(n)I(\hat{\theta}_j(n), \gamma_j^{q(n),k}) \leq \sum_{j \in J_+(n)} t_j(n)I(\hat{\theta}_j(n), \lambda_j^*) + \sum_{j \in J_-(n)} t_j(n)I(\hat{\theta}_j(n), \lambda_j^*)
\]

By definition of \( \lambda^* \) we have \( \forall j \in J_+, I(\hat{\theta}_j(n), \lambda_j^*) \rightarrow 0 \) a.s. as \( n \rightarrow \infty \). Furthermore, since by definition \( \gamma_j^{q(n),k} = q(n) \) and \( \gamma^{q(n),k} \in \Theta \), then for all arms \( j \in J_-(n) \) we have \( \gamma_j^{q(n),k} \geq q(n) - |j - k|L \). From the monotonicity of the KL divergence in its second argument, we therefore have that \( \gamma_j^{q(n),k} = q(n) - |j - k|L, \forall j \in J_- \).

Consequently, since \( q(n) \rightarrow \theta^* \) when \( n \rightarrow \infty \) a.s., we have that as \( n \rightarrow \infty \):

\[
\gamma_j^{q(n),k} \rightarrow \lambda^* \rightarrow \max \{ \theta_j, \theta^* - |j - k|L \} \text{ a.s.}
\]

Since, as shown, the minimizer \( \lambda^* \) is unique, Lemma 6.5.2 implies \( \gamma_j^{q(n),k} \rightarrow \lambda_j^{q(n),k} \) a.s. when \( n \rightarrow \infty \). The proof is now complete provided we prove Lemma 6.5.2. \[ \square \]

**Lemma 6.5.2** Define \( \gamma^{q(n),k}(n) = \arg \min_{\lambda \in \Theta : \lambda_k \geq q(n)} \sum_{j \in K^+} t_j(n)I(\hat{\theta}_j(n), \lambda_j^{q(n),k}) \). Then, in the Lipschitz bandit setting:

\[
\lim_{n \rightarrow \infty} \sum_{i \in K^+} t_i(n) \left[ I(\hat{\theta}_i(n), \lambda_i^{q(n),k}) - I(\hat{\theta}_i(n), \gamma_i^{q(n),k}) \right] = 0.
\]

**Proof.** As with \( \lambda^{q(n),k} \), we will use the shorthand notation \( \gamma^{q(n),k} = \gamma^{q(n),k}(n) \). Consider \( \delta > 0 \) a define the round \( n_\delta \) such that \( \forall i \in K^+ \) and \( n \geq n_\delta \) we have \( |\theta_i - \hat{\theta}_i(n)| < \delta \),
$|q(n) - \theta^*| < \delta$ and $k(n) \in \mathcal{K}^+$. From the LLN and Lemma 6.4.2 we know there exists such an $n_\delta$.

Then we have:

$$\sum_{i \in \mathcal{K}^+} t_i(n) I(\hat{\theta}_i(n), \gamma_i^{q(n),k}) \leq \sum_{i \in \mathcal{K}^+} t_i(n) I(\hat{\theta}_i(n), \lambda_i^{q(n),k}) \quad (6.53)$$

and

$$\sum_{i \in \mathcal{K}} t_i(n) I(\hat{\theta}_i(n), \lambda_i^{q(n),k}) \leq \sum_{i \in \mathcal{K}} t_i(n) I(\hat{\theta}_i(n), \gamma_i^{q(n),k}) \quad (6.54)$$

Define:

$$\Delta_n = \sum_{i \in \mathcal{K}^+} t_i(n) \left[ I(\hat{\theta}_i(n), \lambda_i^{q(n),k}) - I(\hat{\theta}_i(n), \gamma_i^{q(n),k}) \right].$$

and

$$C_n = \sum_{i \notin \mathcal{K}^+} t_i(n) I(\hat{\theta}_i(n), \lambda_i^{q(n),k}).$$

Then, from (6.54) and (6.54) we have that, for all rounds $n \geq n_\delta$ we have $0 \leq \Delta_n \leq C_n$ and consequently:

$$0 \leq \frac{\Delta_n}{\min_{i \in \mathcal{K}^+} t_i(n)} \leq \frac{C_n}{\min_{i \in \mathcal{K}^+} t_i(n)}.$$

Since $t_j(n)/\min_{i \in \mathcal{K}^+} t_i(n) \geq 1$ for all $j \in \mathcal{K}^+$, we have that when $n \to \infty$ (and consequently $t_j \to \infty$ for all $j \in \mathcal{K}^+$):

$$\Delta_n \to 0 \text{ a.s.}$$

which concludes the proof. \hfill \square

### 6.5.1 Proof of Theorem 3.5.4 in Deterministic Setting

Since we are in a deterministic setting, $q(n) = \theta^*$ (in the original stochastic setting, we have $q(n) \to \theta^*$ when $n \to \infty$ - see Lemma 6.4.2) and $\hat{\theta}_k(n) = \theta_k$, $\forall n \in \mathbb{N}$.

We define $\mathcal{K}^+ = \{k \in \mathcal{K} \setminus \{k^*\} : \lim_{n \to \infty} t_k(n) \to \infty\}$ the set of arms played infinitely many times.

Without loss of generality, assume $\mathcal{K} \subset [0, 1]$. We restate out definition of the most confusing parameter $\lambda^{\theta^*,k}$, for an arm $k$ and the level $\theta^*$:

$$\lambda_j^{\theta^*,k} = \max\{\theta_j, \theta^* - L|k - j|\}$$

Now define:

$$f_k(n) = \sum_{j \in \mathcal{K}} t_j(n) I(\theta_j, \lambda_j^{\theta^*,k}).$$

We denote by $j_1(n)$ the last suboptimal arm to be played before time $n$, $j_2(n)$ the 2nd to last and so on. Define $n_i^0(n)$ as the round when arm $j_i(n)$ was last played before $n$. For simplicity, we refer to $n_i^0(n)$ as $n_i^0$ and $j_i(n)$ as $j_i$. 
6.5. Proof of Theorem 3.5.4 - POSLB Asymptotic Regret for Lipschitz Bandits

We prove by induction on \( i \)

\[
\lim_{n \to \infty} \sup f_{j_1}(n) / f(n) = 1.
\]  

(6.55)

The proof uses the following steps:

1) We prove the theorem holds for the last two played arms by showing that the last arm \( j_1 \) can only be played a finite number of times after the last play of \( j_2 \) (the second to last arm played). This is the first step of the induction.

2) Next, by induction, we show that for any arm in \( k \), \( f_j \) before

3) We later show, using the same convexity argument, that if the theorem holds for all arms in \( K^+ \), then \( K^+ = K \) and \( \limsup_{n \to \infty} t_k(n) / f(n) > 0 \), for all \( k \in K \) (Corollary 6.5.1).

Proof of (6.34), Step 1:

We prove:

\[
\lim_{n \to \infty} \sup \frac{f_{j_1}(n)}{f(n)} = \lim_{n \to \infty} \sup \frac{f_{j_1}(n)}{f(n)} = 1.
\]

Since by definition \( k(n_0^2) = j_2 \) we have \( f_{j_2}(n_0^2) \leq f_{j_1}(n_0^2) \). Define:

\[ d = t_{j_1}(n_0^1) - t_{j_1}(n_0^2) = t_{j_1}(n) - t_{j_1}(n_0^2). \]

We will proceed to prove arm \( j_1 \) can only be played a specific (finite) number of times before \( f_{j_1}(n) \) becomes greater than \( f_{j_2}(n) \) and would therefore not be picked again. To this end, we will use the fact that with every play of arm \( j_1 \), \( f_{j_1}(n) \) increases with a constant amount more than \( f_k(n) \) for all \( k \neq j_1 \).

Now let us consider the round \( n_0^1 \). Remembering that \( \lambda_{q,i} = q \):

\[
f_{j_1}(n_0^1) = f_{j_1}(n_0^2) + dI(\theta_{j_1}, \theta^*) + I(\theta_{j_2}, \lambda_{j_2}^{\theta^*, j_1}) \]

(6.56)

and

\[
f_{j_2}(n_0^1) = f_{j_2}(n_0^2) + dI(\theta_{j_1}, \lambda_{j_1}^{\theta^*, j_2}) + I(\theta_{j_2}, \theta^*) \]

(6.57)

Thus, from Equations (6.56) and (6.57):

\[
f_{j_1}(n_0^1) - f_{j_2}(n_0^1) \geq 0 + d[I(\theta_{j_1}, \theta^*) - I(\theta_{j_1}, \lambda_{j_1}^{\theta^*, j_2})] +

+ [I(\theta_{j_2}, \lambda_{j_2}^{\theta^*, j_1}) - I(\theta_{j_2}, \theta^*)].
\]

We then obtain that:

\[
0 \geq \frac{f_{j_1}(n_0^1) - f_{j_2}(n_0^1)}{f(n_0^1)} \geq \left[I(\theta_{j_1}, \theta^*) - I(\theta_{j_1}, \lambda_{j_2}^{\theta^*, j_2})\right] \frac{d}{f(n_0^1)} \]

(6.58)

\[ + \left[I(\theta_{j_2}, \lambda_{j_2}^{\theta^*, j_1}) - I(\theta_{j_2}, \theta^*)\right] \frac{1}{f(n_0^1)} = B \]
with the first inequality is due to \( k(n_0^1) = j_1 \).

Letting \( n \to \infty \):

\[
0 \geq \lim_{n \to \infty} \left[ I(\theta_{j_1}, \theta^*) - I(\theta_{j_1}, \lambda_{j_1}^{\theta_{j_2}^*}) \right] \frac{d}{f(n_0^1)} + \left[ I(\theta_{j_2}, \lambda_{j_2}^{\theta_{j_1}^*}) - I(\theta_{j_1}, \theta^*) \right] \frac{1}{f(n_0^1)}
\]

Since \( I(\theta_{j_1}, \theta^*) - I(\theta_{j_1}, \lambda_{j_1}^{\theta_{j_2}^*}) > 0 \) and \( I(\theta_{j_2}, \lambda_{j_2}^{\theta_{j_1}^*}) - I(\theta_{j_1}, \theta^*) \) \( \frac{1}{f(n_0^1)} \to 0 \) when \( n_0^1 \to \infty \), we immediately obtain:

\[
0 \geq \lim_{n \to \infty} \frac{d}{f(n_0^1)} = \lim_{n \to \infty} \frac{t_{j_1}(n_0^1) - t_{j_1}(n_0^2)}{f(n_0^1)} \geq 0 \quad (6.59)
\]

From (6.58), we obtain:

\[
\frac{f_{j_2}(n_0^1)}{f(n_0^1)} \geq \frac{f_{j_1}(n_0^1)}{f(n_0^1)} \geq \frac{f_{j_2}(n_0^1)}{f(n_0^1)} + B
\]

Letting \( n \to \infty \) (\( B \to 0 \) when and \( n \to \infty \)) we obtain:

\[
\lim_{n \to \infty} \frac{f_{j_2}(n_0^1)}{f(n_0^1)} = \lim_{n \to \infty} \frac{f_{j_1}(n_0^1)}{f(n_0^1)} \leq 1
\]

where the last inequality is due to arm \( j_1 \) being chosen at time \( n_0^1 \). Considering only \( n \) belonging to the infinite set of rounds \( \{n' : k(n') = k^*\} \) (we can restrict ourselves just to these rounds in light of Lemma 6.5.3) and since no suboptimal arm is played between \( n_0^1 \) and \( n \), then for all \( i \in \mathcal{K} \), \( f_i(n_0^1) = f_i(n) \) and therefore:

\[
1 \leq \lim_{n \to \infty} \frac{f_{j_2}(n)}{f(n)} = \lim_{n \to \infty} \frac{f_{j_1}(n)}{f(n)} \leq 1
\]

**Proof of (6.34), Step 2:**

Consider the arm \( k = j_y, y \in \mathbb{N} \) and \( 0 < y < |\mathcal{K}^+| \) and assume that for all \( i, h < y \) we have:

\[
\lim_{n \to \infty} \frac{f_{j_i}(n)}{f(n)} = \lim_{n \to \infty} \frac{f_{j_h}(n)}{f(n)}
\]

To complete the proof is suffices to prove that under this assumption we obtain:

\[
\lim_{n \to \infty} \frac{f_k(n)}{f(n)} = \lim_{n \to \infty} \frac{f_{j_y}(n)}{f(n)} = 1
\]

By our assumption for all \( \alpha \in (0, 1) \) there exists \( n(\alpha) \) such that for all \( n' > n(\alpha) \) and for all \( i, h < y \):

\[
\frac{f_{j_i}(n')(1 + \alpha)}{f(n')} \geq \frac{f_{j_h}(n')}{f(n')}
\]

(6.60)

Consider \( \alpha \in (0, 1) \) and a time \( n \) such that \( n_0^{|\mathcal{K}^+|}(n) > n(\alpha) \) and \( k(n) = k^* \). Define \( k_1 = \max \{j_i : j_i \leq j_y, i < y\} \) and \( k_2 = \min \{j_i : j_i \geq j_y, i < y\} \) (if \( k_1 = \emptyset \) then
$k_1 = k$, if $k_2 = \emptyset$, then $k_2 = k$) as the closest arms to the left and right of $k$ that have been played after arm $k$ (or $k$ if such an arm doesn’t exist).

We define:

$$a_1(n, x) = \frac{\sum_{i \in \mathcal{K}^+, i \leq k_1} (t_i(n) - t_i(n_0^k))I(\theta, \lambda_{x, i}^\theta)}{\sum_{i \in \mathcal{K}^+, i \leq k_1} t_i(n) - t_i(n_0^k)}$$

$$a_2(n, x) = \frac{\sum_{i \in \mathcal{K}^+, i \geq k_2} (t_i(n) - t_i(n_0^k))I(\theta, \lambda_{x, i}^\theta)}{\sum_{i \in \mathcal{K}^+, i \geq k_2} t_i(n) - t_i(n_0^k)}$$

Intuitively, $a_1(n, x)$ represents the average amount with which a play to the left of $k$ (between $n_0^k$ and $n$) increases the amount in $f_x(n)$. Similarly, $a_2(n, x)$ represents the same amount but for arms to the right of $k$. From the perspective of arm $k$, playing arms to its left and right would increase $f_k(n)$ as if we only play two fictitious neighboring arms $k'_1$ and $k'_2$ and $I(\theta_{k'_1}, \lambda_{k'_1}^{\theta, k}) = a_1(n, k)$ and $I(\theta_{k'_2}, \lambda_{k'_2}^{\theta, k}) = a_2(n, k)$. Thus, this scenario reduces to a simple setting with three suboptimal arms, with $k$ in the middle and $k^*$ to the far left or far right.

Define $I^+(x, y) = I(x, y)$ if $x < y$ and 0 otherwise. Then note that $\forall i \in \mathcal{K}$, $i < k_1$ and for all $n$ and $q \geq \hat{\theta}_i(n) + L|k_1 - i|$ we have that:

$$\frac{(k_2 - k)I^+(\theta, q - L|k_1 - i|) + (k - k_1)I^+(\theta, q - L|k_2 - i|)}{k_1 - k_2} > I(\theta, q - L|k - i|).$$

The above comes from the observation that the KL divergence $I(x, y)$ is convex in $y$ over the interval $(x, 1)$, for all $x \in (0, 1)$ and therefore $I(x, a\frac{y_1 + by_2}{a + b}) < \frac{bI(x, y_1) + aI(x, y_2)}{a + b}$ for any $a, b \in (0, 1)$, $a \neq b$, and $y_1, y_2, x \in (0, 1), y_1, y_2 > x, y_1 \neq y_2$.

Consequently:

$$\frac{a_1(n, k_1)(k_2 - k) + a_1(n, k_2)(k - k_1)}{k_2 - k_1} > a_1(n, k), \forall n. \quad (6.61)$$

Similarly:

$$\frac{a_2(n, k_1)(k_2 - k) + a_2(n, k_2)(k - k_1)}{k_2 - k_1} > a_2(n, k), \forall n. \quad (6.62)$$

Define $d_1 = \sum_{i \leq k_1, i \neq k^*} t_i(n) - t_i(n_0^k)$ and $d_2 = \sum_{i \geq k_2, i \neq k^*} t_i(n) - t_i(n_0^k)$, the number of times arms to the left and right, respectively, of $k$ were played since the last play of arm $k$.

Define $A = \sum_{i \notin \mathcal{K}^+} t_i(n)I(\theta, \lambda_{i}^{\theta, k})$ (the terms corresponding to the arms that do not matter - i.e. not played infinitely many times) and note that:

$$\lim_{n \to \infty} A/f(n_0^k) = 0. \quad (6.63)$$
Therefore, we have that:

\[ f_k(n) \leq f_k(n_0^y) + d_1 a_1(n, k) + d_2 a_2(n, k) + A + I(\theta_k, \theta^*) \]

\[ \leq \frac{f_k_1(n_0^y)(k_2 - k) + f_k_2(n_0^y)(k - k_1)}{k_2 - k_1} + d_1 a_1(n, k) + d_2 a_2(n, k) + A + I(\theta_k, \theta^*) \]

\[ \leq \frac{f_k_1(n_0^y)(k_2 - k) + f_k_2(n_0^y)(k - k_1) + A + I(\theta_k, \theta^*) + d_1 a_1(n, k_1) + a_2(n, k_2)(k_2 - k)}{k_2 - k_1} \]

\[ = \frac{f_k_1(n)(k_2 - k) + f_k_2(n)(k - k_1)}{k_2 - k_1} + A + I(\theta_k, \theta^*). \]

where the second inequality is true since by definition \( f_k(n_0^y) \leq f_i(n_0^y), \forall i \in K \) (because \( k(n_0^y) = k \)), and the third inequality is due to equations (6.61) and (6.62) (which in turn stem from the convexity of the KL divergence).

Then, dividing by \( f(n) \), we obtain:

\[ \frac{f_k(n)}{f(n)} < \frac{f_k_1(n)(k - k_1) + f_k_2(n)(k_2 - k)}{f(n)(k_2 - k_1)} + \frac{A + I(\theta_k, \theta^*)}{f(n)} \]

\[ \leq \frac{f_k_1(n)(k_2 - k_1) + \alpha(k_2 - k)f_k_1(n)}{f(n)(k_2 - k_1)} + \frac{A + I(\theta_k, \theta^*)}{f(n)} \]

where the last inequality comes from our assumption in Equation (6.60). Since \( k(n) = k^* \), and from Equation (6.63), letting \( n \to \infty \) we obtain:

\[ 1 \leq \lim_{n \to \infty} \frac{f_k(n)}{f(n)} \leq \lim_{n \to \infty} \frac{f_k_1(n)}{f(n)} \left( 1 + \frac{\alpha(k_2 - k)}{k_2 - k_1} \right) = 1 + \frac{\alpha(k_2 - k)}{k_2 - k_1}. \]

Letting \( \alpha \to 0 \) we obtain the advertised result. \( \square \)

**Lemma 6.5.3** There exists \( n_0 \) such that \( \forall n \geq n_0 \) we have \( \exists 0 \leq i \leq 2K \) such that \( k(n + i) = k^* \).

**Proof.** Consider \( n_0 \) such that \( k(n_0 - 1) = k^* \) and \( f(n_0 + 2K) - f(n_0 - 1) < \min_k I(\theta_k, \theta^*) \). Such an \( n_0 \) exists since \( k^* \) is played infinitely many times and \( f(n' + K) - f(n') \to 0 \) when \( n' \to \infty \). Assume there exists \( n > n_0 \) such that only suboptimal arms are played between (and including) rounds \( n \) and \( n + 2K \).

Then, at time \( n - 1 \) for all \( k \in K^- \) we have \( f(n - 1) \leq f_k(n - 1) \) and consequently for all suboptimal arms played between \( n \) and \( n + 2K \) we have:

\[ f(n + 2K) < f(n - 1) + I(\theta_k, \theta^*) \]

hence, each suboptimal arm can only be played at most once between rounds \( n \) and \( n + 2K \). Therefore, \( \exists 0 < i < 2K \) such that \( k(n + i) = k^* \), which concludes the proof. \( \square \)
Lemma 6.6.1 allows to control the fluctuations of the estimate $\hat{\theta}_i(n)$ evaluated at a random time $\phi$. We assume that $\phi$ is a stopping time, and that the number of rounds before $\phi$ where a decision containing $i$ has been taken is greater than a number $s$. This result is instrumental in analyzing the finite time regret of algorithms (such as ours) that take decisions based on the estimates $\hat{\theta}_i(n)$. Lemma 6.6.1 is a consequence of Lemma 6.6.2, which is reproduced here for completeness.

**Lemma 6.6.1** Let $\{Z_t\}_{t \in \mathbb{Z}}$ be a sequence of independent random variables with values in $[0, 1]$. Define $\mathcal{F}_n$ the $\sigma$-algebra generated by $\{Z_t\}_{t \leq n}$ and the filtration $\mathcal{F} = (\mathcal{F}_n)_{n \in \mathbb{Z}}$. Consider $s \in \mathbb{N}$, $n_0 \in \mathbb{Z}$ and $T \geq n_0$. We define $S_n = \sum_{t=n_0}^n B_t(Z_t - \mathbb{E}[Z_t])$, where $B_t \in \{0, 1\}$ is a $\mathcal{F}_{t-1}$-measurable random variable. Further consider that for all $t$, almost surely we have $B_t \geq \bar{B}_t C_t$, where both $\bar{B}_t$ and $C_t$ are $\{0, 1\}$-valued, $\mathcal{F}_{t-1}$-measurable random variables, such that for all $t$: $\mathbb{P}[C_t = 1] \geq c > 0$.

Further define $t_n = \sum_{t=n_0}^n B_t$ and $c_n = \sum_{t=n_0}^n \bar{B}_t$. Define $\phi \in \{n_0, \ldots, T + 1\}$ a $\mathcal{F}$-stopping time such that either $c_\phi \geq s$ or $\phi = T + 1$.

Then for all $\epsilon > 0$ we have that:

$$\mathbb{P}[S_\phi \geq t_\phi \delta, \phi \leq T] \leq e^{2s\epsilon^2 c^2} + e^{2c(1-\epsilon)s\delta^2}.$$ 

As a consequence:

$$\mathbb{P}[|S_\phi| \geq t_\phi \delta, \phi \leq T] \leq 2(e^{2s\epsilon^2 c^2} + e^{2c(1-\epsilon)s\delta^2}).$$

**Proof.** We prove the first statement, as the second statement follows by symmetry.

When event $S_\phi \geq t_\phi \delta$ occurs, we have either that (a) $t_\phi \leq c(1 - \epsilon)c_\phi$ or (b) $S_\phi \geq t_\phi \delta$ and $t_\phi \geq c(1 - \epsilon)c_\phi \geq c(1 - \epsilon)s$. In case (a), if $\phi \leq T$, we have:

$$\sum_{t=n_0}^n \bar{B}_t C_t \leq \sum_{t=n_0}^n B_t = t_\phi \leq c(1 - \epsilon)c_\phi = c(1 - \epsilon) \sum_{t=n_0}^\phi \bar{B}_t,$$

and therefore:

$$\sum_{t=n_0}^\phi \bar{B}_t C_t \leq c(1 - \epsilon) \sum_{t=n_0}^\phi \bar{B}_t$$

$$\sum_{t=n_0}^\phi \bar{B}_t (C_t - c) \leq -c\epsilon \sum_{t=n_0}^\phi \bar{B}_t$$

$$\sum_{t=n_0}^\phi \bar{B}_t (C_t - \mathbb{E}[C_t]) \leq -c\epsilon \sum_{t=n_0}^\phi \bar{B}_t.$$
where the last inequality holds because \( \mathbb{E}[C_t] \geq c \) for all \( t \). We may now apply Lemma 6.6.2 (with \( Z_t = C_t \), \( B_t = B_t \), and \( \delta \equiv \epsilon \)) to obtain:

\[
\mathbb{P}[t_\phi \leq c(1-\epsilon)c_\phi, \phi \leq T] \\
\leq \mathbb{P} \left[ \sum_{t=n_0}^{\phi} \bar{B}_t(C_t - \mathbb{E}[C_t]) \leq -c\epsilon \sum_{t=n_0}^{\phi} \bar{B}_t, \phi \leq T \right] \\
\leq e^{-2s\epsilon^2 c^2}.
\]

In case (b), define another stopping time \( \phi' \), such that \( \phi' = \phi \) if \( t_\phi \geq c(1-\epsilon)c_\phi \) and \( \phi' = T + 1 \) otherwise. Note that \( \phi' \) is indeed a stopping time. We apply Lemma 6.6.2 a second time (with \( \phi \equiv \phi' \), \( s \equiv c(1-\epsilon)s \)) to obtain:

\[
\mathbb{P}[S_\phi \geq t_\phi \delta, t_\phi \geq c(1-\epsilon)c_\phi, \phi \leq T] \\
= \mathbb{P}[S_{\phi'} \geq t_\phi \delta, \phi' \leq T] \leq e^{-2c(1-\epsilon)s\delta^2}.
\]

Summing the inequalities obtained in cases (a) and (b), we prove the announced result:

\[
\mathbb{P}[S_\phi \geq t_\phi \delta, \phi \leq T] \leq e^{-2s\epsilon^2 c^2} + e^{-2c(1-\epsilon)s\delta^2}.
\]

which concludes the proof.

\[ \square \]

**Lemma 6.6.2** ([9]) Let \( \{Z_t\}_{t \in \mathbb{Z}} \) be a sequence of independent random variables with values in \([0,1]\). Define \( \mathcal{F}_n \) the \( \sigma \)-algebra generated by \( \{Z_t\}_{t \leq n} \) and the filtration \( \mathcal{F} = (\mathcal{F}_n)_{n \in \mathbb{Z}} \). Consider \( s \in \mathbb{N} \), \( n_0 \in \mathbb{Z} \) and \( T \geq n_0 \). We define \( S_n = \sum_{t=n_0}^{n} B_t(Z_t - \mathbb{E}[Z_t]) \), where \( B_t \in \{0,1\} \) is a \( \mathcal{F}_{t-1} \)-measurable random variable. Further define \( t_n = \sum_{t=n_0}^{n} B_t \). Define \( \phi \in \{n_0, \ldots, T+1\} \) a \( \mathcal{F} \)-stopping time such that either \( t_\phi \geq s \) or \( \phi = T + 1 \).

Then we have that:

\[
\mathbb{P}[S_\phi \geq t_\phi \delta, \phi \leq T] \leq \exp(-2s\delta^2).
\]

As a consequence:

\[
\mathbb{P}[|S_\phi| \geq t_\phi \delta, \phi \leq T] \leq 2 \exp(-2s\delta^2).
\]

Lemma 6.6.3 is a consequence of Lemma 6.6.1, and allows to upper bound the size of random sets of rounds where decisions containing \( i \) have been sampled and the empirical mean \( \hat{\theta}_i(n) \) deviates from its expectation by more than a fixed amount \( \delta > 0 \).

**Lemma 6.6.3** Let us fix \( c > 0 \) and \( 1 \leq i \leq N \). Consider a random set of rounds \( H \subset \mathbb{N} \), such that, for all \( n \), \( \mathbb{I}\{n \in H\} \) is \( \mathcal{F}_{n-1} \) measurable. Further assume for all \( n \) we have: \( \mathbb{E}[\alpha_i(n) \mathbb{I}\{n \in H\}] \geq c > 0 \). Consider a random set \( \Lambda = \bigcup_{s \geq 1} \{\tau_s\} \subset \mathbb{N} \), where for all \( s \), \( \tau_s \) is a stopping time such that \( \sum_{n=1}^{\tau_s} \mathbb{I}\{n \in H\} \geq s \).

Then for all \( i \) and \( \epsilon > 0 \) and \( \delta > 0 \) we have that:

\[
\sum_{n \geq 0} \mathbb{P}[n \in \Lambda, |\hat{\theta}_i(n) - \theta_i| \geq \delta] \leq c^{-1} \left[ \frac{1}{\epsilon^2 c} + \frac{1}{\delta^2 (1-\epsilon)} \right].
\]
As a consequence:

\[
\sum_{n \geq 0} \mathbb{P}[n \in \Lambda, |\hat{\theta}_i(n) - \theta_i| \geq \delta] \leq 2c^{-1} \left[ 2c^{-2} + \delta^{-2} \right].
\]

**Proof.** Fix \( T < \infty \) and \( s \). Apply Lemma 6.6.1 (with \( Z_t \equiv X_i(t), B_t \equiv \alpha_i(n), \tilde{B}_t \equiv \mathbb{1}\{n \in H\}, C_t \) a Bernoulli variable with parameter \( \mathbb{E}[\alpha_i(n)|n \in H] \) which is conditionally independent of \( \mathbb{1}\{n \in H\} \)) to obtain:

\[
\mathbb{P}[|\hat{\theta}_i(\tau_s) - \theta_i| \geq \delta, \tau_s \leq T] \leq 2(e^{-2se^2c^2} + e^{-2c(1-\epsilon)s\delta^2}).
\]

Using a union bound over \( s \), for all \( \epsilon > 0 \) we get:

\[
\sum_{n \leq T} \mathbb{P}[n \in \Lambda, |\hat{\theta}_i(n) - \theta_i| \geq \delta] \leq \sum_{s \geq 1} \mathbb{P}[|\hat{\theta}_i(\tau_s) - \theta_i| \geq \delta, \tau_s \leq T]
\]

\[
\leq \sum_{s \geq 1} 2(e^{-2se^2c^2} + e^{-2c(1-\epsilon)s\delta^2})
\]

\[
\leq \frac{1}{e^2c^2} + \frac{1}{c(1-\epsilon)\delta^2}
\]

\[
= c^{-1} \left[ \frac{1}{e^2c^2} + \frac{1}{\delta^2(1-\epsilon)} \right],
\]

where we have used the following inequality twice:

\[
\sum_{s \geq 1} e^{-sw} \leq \int_0^{+\infty} e^{-sw} ds = 1/w,
\]

valid for all \( w > 0 \). Since the above inequality holds for all \( T \), and its r.h.s. does not depend on \( T \) we conclude that:

\[
\sum_{n \geq 1} \mathbb{P}[n \in \Lambda, |\hat{\theta}_i(n) - \theta_i| \geq \delta] \leq c^{-1} \left[ \frac{1}{e^2c^2} + \frac{1}{\delta^2(1-\epsilon)} \right],
\]

which concludes the proof of the first statement. The second statement is obtained by setting \( \epsilon = 1/2 \). \( \square \)

**Corollary 6.6.1** Consider \( c > 0 \) and \( 1 \leq i \leq N \) fixed. Consider a random set of instants \( H \subset \mathbb{N} \), such that, for all \( n, \mathbb{1}\{n \in H\} \) is \( \mathcal{F}_{n-1} \) measurable. Further assume for all \( n \) we have:

\[ \mathbb{E}[\alpha_i(n)|n \in H] \geq c > 0. \]

Define \( h_i(n) = \sum_{n' \leq n} \mathbb{1}\{n' \in H\} \). Consider \( \epsilon > 0 \) and \( \delta > 0 \) and define the set:

\[
\mathcal{H} = \left\{ n \in H : (t_i(n) \leq (1-\epsilon)h_i(n)) \lor (|\hat{\theta}_i(n) - \theta_i| \geq \delta) \right\}
\]

Then we have:

\[
\mathbb{E}[|\mathcal{H}|] \leq c^{-1} \left[ \frac{1}{\epsilon^2} + \frac{1}{\delta^{-2}}(1-\epsilon)^{-1} \right].
\]

**Proof.** Straightforward from the proof of Lemma 6.6.3 with \( \Lambda = H \). \( \square \)

Lemma 6.6.4 is a straightforward consequence of Theorem 10 in [2], and states that the expected number of times the index of a given item \( i \) underestimates its true value is finite, and upper bounded by a constant that does not depend on the parameters \( (\theta_i)_i \).
Lemma 6.6.4 \((\{2\})\) Define:

\[ b_i(n) = \max\{q \in [0, 1] : t_i(n)I(\hat{\theta}_i(n), q) \leq f(n)\}, \]

with \(f(n) = \log(n) + 4 \log(\log(n))\).

There exists a constant \(C_0\) independent of \((\theta_i)_i\) such that for all \(i\) we have:

\[ \sum_{n \geq 0} P[b_i(n) < \theta_i] \leq C_0. \]

In particular one has \(C_0 \leq 2e \sum_{n \geq 1} [f(n) \log(n)]e^{-f(n)} \leq 15.\)