Politecnico di Milano
Dipartimento di Elettronica, Informazione e Bioingegneria
Doctoral Program In Information Technology

A Learning Approach for Pricing in e-Commerce Scenario

Doctoral Dissertation of:
Stefano Paladino

Supervisor:
Prof. Nicola Gatti
Co-supervisors:
Prof. Marcello Restelli
Ph.D. Francesco Trovò

Tutor:
Prof. Francesco Amigoni

Chair of the Doctoral Program:
Prof. Andrea Bonarini

2017 – Cycle XXX

Abstract

Over the past few years, there has been a significant increase in the use of e-commerce websites. Nowadays, almost everything can be bought online, and market research shows that the online market is steadily growing. With the spread of e-commerce, metasearch engines began to arise, which conduct searches across multiple independent e-commerce, as a response to the need of users to compare offerings without having to consult each website individually. Metasearch engines effectively act as middlemen, sending a large portion of the traffic to e-commerce, but giving them barely no information about customers' behavior and purchasing history.

In this thesis, we investigate the problem of optimal pricing in the setting of online sales of goods from the point of view of an e-commerce selling its products in the profitable and challenging environment of metasearch engines. In this setting, we have a vast catalog of items to price, no information about our customers, low conversion rates and the environment is non-stationary. We study the problem of finding the pricing strategy that maximizes the profit of the e-commerce selling its products in this scenario. We propose an automatic pricing system which uses clustering techniques to partition the catalog of items into subsets sharing similar features, and machine learning techniques to learn the optimal price of each subset.

First, we deal with the problem of partitioning the catalog of items. We tackle it by learning from historical data collected by the interactions with customers. We propose a novel algorithm which, differently from existing solutions, provides a clear interpretability to business analysts of the resulting model and a risk-averse pricing policy to maximize the profit. With

a wide experimental campaign, we present empirical evidence for the improved performance of our algorithm over the state-of-the-art ones.

Then, we study algorithms to learn the optimal policy to follow in each of the subset. We focus on online learning techniques, in particular on the Multi-Armed Bandit ones, widely studied in the machine learning literature. Even if existing general-purpose algorithms can be applied to the pricing task, we propose novel algorithms exploiting the properties of the pricing problem (some of them already studied in literature, other unexplored so far). We derive upper bounds over the regret for the proposed algorithms and we present a thorough experimental evaluation in a wide range of configurations, some of them based on real-world data, showing that we significantly improve the performances of the state-of-the-art algorithms.

The clustering algorithm and the online learning policies we propose are interconnected and continuously communicating. Indeed, the data generated from the interaction of the users with the bandits algorithms are collected and passed through the clustering algorithm to update the partitioning and to improve the performance of the system.

Contents

CHAPTER 1

Introduction

1.1 Scenario

Over the past few years, there has been a significant increase in the use of e-commerce websites. Thanks to technological progress and the massive adoption of e-commerce, almost everything can be bought online, from groceries and clothing to holiday packages and cars. Market research shows that the one of e-commerce is a market with a global value of more than two trillion USD, and it will even grow in the future [1].

Online markets have many features that can be exploited by vendors, thus, with the advent of e-commerce, a number of new strategies became possible [2]. For instance, prices can be easily adjusted without incurring in any cost, while, in traditional markets, price changes would often induce costs, since a new catalog had to be printed or price tags had to be replaced. Furthermore, in online markets it is possible to access historical data without substantial costs, making it easier for vendors to study customers' behavior in order to make more accurate and informed decisions.

With the spread of e-commerce, metasearch engines began to arise as well. These tools are so named as they conduct searches across multiple independent e-commerce and they aggregate the results, to allow customers to evaluate and compare the offers for a product more clearly. The scenario of online selling of travel products is a noteworthy example in which the role of metasearch engines is acquiring a great importance. In this scenario, we have Online Travel Agencies (OTAs) which provide online booking facilities for flight tickets, hotels and other travel-related services to customers. Some of the most famous OTAs are lastminute.com, Expedia and eDreams. Metasearch engines have emerged in this field as a response to the need of users to compare offerings without having to consult each OTA individually. The most famous metasearch engines are websites such as Skyscanner, Google Flights or Kayak. Their relevance has been increasing in recent years, and market analysis for the US shows that travelers, when asked about their last flight trips, were almost equally likely to have consulted metasearch engines versus OTAs websites, with roughly three-quarters of the total doing so [3].

Our work investigates the pricing problem in the setting of online sales of digital goods from the point of view of an e-commerce, such as an OTA, selling its products in a metasearch environment. Metasearch engines send a lot of traffic, then resulting in sales, to OTAs' websites. From the data of one of the major European OTA, we saw that more than a half of the profits are made from the sales on metasearch engines.[1] Thus, this scenario presents a great profitability, but also a number of characteristics which makes the problem very challenging. We have a vast catalog of items to price. We have almost no information about our customers since users do not directly use our websites but they go on metasearch engines, which actually act as middlemen. They send a lot of traffic to OTAs' websites, but they give OTAs very few information about customers' behavior and purchasing history. Furthermore, most users perform searches without the actual intent of buying but only for informational purposes, generating a huge amount of searches performed every day on metasearch engines, but with only few of them converting into bookings. Another difficulty arises from the non-stationarity of the environment, since we have seasonal effects on the market and we have a lot of competitors which impact on the non-stationarity of the environment by changing their marketing strategies.

All these characteristics make the problem of finding the optimal pricing strategy really complex and with a lot of variables to take into account. It is tough for a human operator to tackle this computational burden, considering all the facets of the problem.

[1] We do not specify the name of the online travel agency due to confidentiality issues.

1.2 Proposed Solution

In this dissertation, we study the problem of finding the pricing strategy that maximizes the expected profit of an e-commerce. We design an automatic pricing system which uses clustering techniques to partition the catalog in contexts of items sharing similar features, and online learning techniques to learn the optimal price of each context.

First, we tackle the clustering problem by learning from historical data collected by recording the interactions with customers. We focus on the Learning from Logged Bandit Feedback (LLBF) setting. Commonly, the logs generated by the interaction between the system and a user present the structure of a sequential decision process: basing on a context, the system takes an action from a set of possible choices and, afterwards, the user provides the system with a feedback, in terms of reward. The peculiarity of this setting is that the feedback, as it happens in bandit settings, is only on the chosen action to show to the user, while no information is available about other possible actions. Some approaches had been proposed in the last years to address this problem, but they lack in some of the fundamental characteristics that make an algorithm suitable for practical purposes. Indeed, they did not provide a clear interpretability of the final model since there is no direct method to infer those features that most influence the resulting model. Moreover, in economics scenarios, it is important that the proposed algorithm should be as risk-averse as possible, but most of the theoretical guarantees available in literature are provided in terms of average value. Finally, existing approaches usually require knowledge of the behavior of the user and assume it to be stationary, which is rarely met in practice in microeconomics scenarios. In this work, we propose a novel algorithm, whose goal is to solve all the mentioned drawbacks of the literature approaches. The algorithm we propose can learn a risk-averse policy to maximize the expected profit and makes use of statistical lower confidence bounds to build a decision tree, which provides both a decisional tool over future samples and an instrument to highlight the features that influence the profit the most.

Then, we study algorithms to learn the optimal policy to follow in each context to find the price that maximizes the expected profit. We study online learning techniques, in particular the Multi-Armed Bandit (MAB) ones, which have been widely studied in literature and provided evidence to be effective also in real-world scenario. MAB problems have been tackled with two distinct approaches, the frequentist and the Bayesian ones. The goal of a frequentist algorithm is to achieve the best parameter-dependent performance, and the expected mean rewards corresponding to the arms are considered as unknown deterministic quantities. Conversely, in the Bayesian approach, each arm is characterized by a distribution corresponding to the arm parameter. Even if it is possible to use existing general-purpose algorithms to solve our problem, by exploiting the pricing structure we can improve the performance of the classical algorithms. More specifically, we exploit the monotonicity of the conversion rate in the price and the fact that e-commerce sellers have *a priori* information about the customers' behavior and the maximum conversion rate. To the best of our knowledge, these two properties have never been studied before. Furthermore, we tackle both stationary and non-stationary settings, as already done in literature. Finally, we study the property of unimodality over the expected profit. We present algorithms exploiting one or more of these features at the same time, also providing theoretical guarantees for all of the proposed methods.

These are the techniques we used to design an automatic pricing system, deployed in collaboration with one of the major Online Travel Agencies in Europe. The two problems of clustering and online learning algorithms are interconnected and continuously communicating: the data generated from the interaction of the users with our MAB algorithms are collected and passed through our LLBF algorithm to update the contexts model and to improve the performance of the system.

1.3 Structure of the Thesis

The remaining part of this thesis is structured as follows:

- In Chapter 2, we present some preliminaries necessary to understand the remaining part of the thesis.
- In Chapter 3, we give a general overview on the architecture of our proposed solution and we describe the state-of-the-art techniques related to it.
- In Chapter 4, we analyze the clustering problem to tackle the partitioning of the catalog in contexts.
- In Chapter 5, we propose frequentist techniques to exploit the monotonicity property of conversion rates as well as the *a priori* information on the maximum conversion rate, both in stationary and non-stationary settings.
- In Chapter 6, we focus on the Bayesian approach and we design novel algorithms to tackle non-stationary environment and the property of unimodality over the expected profit.

- In Chapter 7, we summarize the results obtained and we provide some suggestions for future works.

CHAPTER 2

Preliminaries

In this chapter, we present the essential preliminaries needed to understand the remaining part of the thesis. First, we introduce the Learning from Logged Bandit Feedback (LLBF) problem. Then, we present the stochastic Multi-Arm Bandit (MAB) formulation.

2.1 Learning from Logged Bandit Feedback

Logged data is one of the most widespread forms of recorded information since almost any system can acquire it and stored at a little cost. Commonly, the logs generated by the interaction between the system and users present the structure of a sequential decision process: by basing on a context, the system takes an action from a set of possible choices and, afterwards, the user provides the system with a feedback, in terms of either reward or loss. The peculiarity of this setting is that the feedback, as it happens in *bandit* settings, is only on the action observed by the user, while no information is available about other possible choices. The problem of learning a policy mapping each context to an action from interactions which took place in the past is known in literature as the Learning from Logged Bandit Feedback (LLBF) problem. This setting is fundamentally different from classical supervised learning, where correct predictions together with a loss function provide a full-information feedback.

Consider an LLFB setting defined as the tuple $(\mathcal{X}, A, R)$, where $\mathcal{X} = (X, \mathcal{D})$ is a finite-dimensional multivariate probability space of contexts with support in $X \subseteq \{0,1\}^c$ with $c \in \mathbb{N}$ and unknown multivariate distribution $\mathcal{D}$, $A := \{a_1, \dots, a_K\}$ with $K \in \mathbb{N}$ is the finite action space, and R is the reward distribution. A generic sample $z_i = (x_i, a_i, r_i)$ has a context vector $x_i = (x_{i1}, \dots, x_{ic}) \in X$, which is drawn from the distribution $\mathcal{D}$, i.e., $x_i \sim \mathcal{D}$. The corresponding action $a_i \in \mathcal{A}$ is chosen by a generic sampling policy $\mathfrak{U}_0$, i.e., $a_i \sim \mathfrak{U}U_0$, which is assumed to be unknown. Finally, the reward r_i gained by selecting action a_i in the context x_i is the realization of a random variable $R(x_i, a_i)$ with unknown distribution $\mathcal{R}(x_i, a_i)$ and finite support $\Omega \subset \mathbb{R}$ (w.l.o.g. from now on we consider $\Omega \subseteq [0,1]$) provided for the chosen action a_i in the chosen context x_i, i.e., $R(x_i, a_i) \sim \mathcal{R}(x_i, a_i)$. We denote with $\mu(x_i, a_i)$ the expected value of the reward $R(x_i, a_i)$, i.e., $\mu(x_i, a_i) := \mathbb{E}[R(x_i, a_i)]$, where the expected value is computed over the distribution $\mathcal{R}(x_i, a_i)$. A policy (or mapping) $\mathfrak{U}$ dealing with the LLBF problem is a function (either deterministic or stochastic) providing for each context $x \in X$ the choice of the action $a \in A$, i.e., $\mathfrak{U}(x) = a$. The performance of a policy $\mathfrak{U}(\cdot)$ over a generic LLBF problem $(\mathcal{X}, A, R)$ can be evaluated by means of its the expected *profit*, defined as:

$$P(\mathfrak{U}) = \mathbb{E}\left[R(x, a)\right],$$

where the expectation is taken with respect to the considered policy $\mathfrak{U}$ and the reward distributions $\{\mathcal{R}(x, a)\}_{x \in X, a \in A}$.

In [4], the authors propose an algorithm based on counterfactual risk minimization called POEM (Policy Optimizer for Exponential Models) for learning stochastic linear rules for structured output prediction. They develop a new objective function considering both estimated rewards and their uncertainty, and propose an optimization procedure to fit a linear classification model. The number of parameters in their model is usually large since it linearly depends both on the number of context variables and the number of arms. For this reason they develop an efficient method to train the model, by decomposing the objective function in different terms and performing stochastic gradient descend. Then, they test POEM both on simulated bandit feedback, derived from existing full information classification dataset as already done in literature in [5], and on real-world application.

POEM is one of the possible solutions to LLBF problem. In the next chapter, we give a more detailed review of the other state-of-the-art algorithms related to our work used to tackle this problem.

2.2 Multi-Arm Bandit

A Multi-Armed Bandit (MAB) problem is a sequential allocation problem defined by a set of actions. At each round, an item is allocated to an action and a payoff is obtained. The goal is to maximize the total payoff obtained in a sequence of allocations.

“One armed bandit” is a colloquial name given to the original slot machines. These machines have one long arm, or lever, that set the mechanism in motion. They got the nickname due to their propensity to steal all of your money. In a casino, the player faces a sequential allocation problem, since he is dealing with many slot machines at once (a "multi-armed bandit") and must repeatedly choose the next arm to pull. Once the player pulls an arm, he can only see the reward of the chosen arm and cannot know what reward he would have got pulling other arms. Bandit problems address the fundamental trade-off between exploration and exploitation. The player must balance the exploitation of arms that did well in the past and the exploration of arms that might give higher payoffs in the future. To analyze the behavior of a player using a bandit strategy, we compare its performance with the one of a clairvoyant strategy that systematically plays the best arm in terms of payoffs. In other terms, we study the loss, usually addressed as *regret*, of the player for not playing always optimally.

More formally, for any horizon of N rounds, given $K \geq 2$ arms and sequences $X_{i,1}, X_{i,2}, \dots, X_{i,N}$ of unknown rewards associated with each arm a_i of the set $A = \{a_1, a_2, \dots, K\}$, we study a policy $\mathfrak{U}(h_t)$ that at each round $t = 1, 2, \dots, N$ selects an arm a_{i_t} and receives the associated reward $X_{i_t,t}$, given history h_t. The regret after N plays is defined by:

$$R_N = \max_{i=1,\dots,K} \sum_{t=1}^{N} X_{i,t} - \sum_{t=1}^{N} X_{i_t,t}.$$

Since both the rewards and the player’s choices might be stochastic, in what follows we will refer to the notion of *pseudo-regret*, defined as:

$$\bar{R}_N = \max_{i=1,\dots,K} \mathbb{E}\left[\sum_{t=1}^{N} X_{i,t} - \sum_{t=1}^{N} X_{i_t,t}\right].$$

The pseudo-regret is a weaker notion of regret since one competes against the action which is optimal only in expectation, but it is a more natural figure of merit in a stochastic framework. In the stochastic formulation of MAB problem, the rewards $X_{i,t}$ of each arm a_i are independent draws from an unknown probability distribution ν_i, such as Bernoulli distribution. For $i =$

$1, \ldots, K$, we denote by μ_i the mean, or expected value, of distribution ν_i and we define the optimal arm a_{i^*} and the mean μ_{i^*} of the optimal arm as, respectively:

$$a_{i^*} = \arg \max_{i=1,\ldots,K} \mu_i,$$

$$\mu_{i^*} = \max_{i=1,\ldots,K} \mu_i.$$

Let $T_i(t) = \sum_{m=1}^{t} \mathbb{1}\{\mathfrak{U}(h_m) = a_i\}$ be the number of times the arm a_i was pulled in the first t rounds, where $\mathbb{1}\{B\}$ denotes the indicator function of the event B. Now, the pseudo-regret can be written as:

$$\bar{R}_N = \mu_{i^*} N - \sum_{t=1}^{N} \mathbb{E}\left[\mu_{i_t}\right] = \mu_{i^*} N - \sum_{i=1}^{K} \mu_i \mathbb{E}[T_i(N)].$$

A simple principle to follow to deal with the exploration-exploration dilemma is the so-called *optimism in the face of uncertainty*. Despite our lack of knowledge in which is the best arm, we will consider an optimistic guess to decide how good the expected reward of each arm is, and we will pull the arm with the highest guess. If the guess is wrong, then the optimistic guess will decrease, leading the choice to a different arm. If the guess is good, we will be able to exploit that arm and to incur in little regret. Thanks to this principle, we can balance exploration and exploitation.

The policy UCB1, proposed in [6], follows the above mentioned principle, making use of *upper confidence bound* as a form of optimism. Formally, we want that, in high probability, the true expected value μ_i of an arm is lower than a prescribed upper bound. Let $\bar{X}_{i,t}$ be the empirical mean, at round t, of the outcomes obtained by pulling arm a_i for $T_i(t-1)$ rounds:

$$\bar{X}_{i,t} := \frac{1}{T_i(t-1)} \sum_{n=1}^{T_i(t-1)} X_{i,n}.$$

UCB1 uses the Chernoff-Hoeffding inequality [7], that gives an upper bound on the probability that $\bar{X}_{i,t}$ deviates from its expected value μ_i:

$$\mathbb{P}(\bar{X}_{i,t} + \varepsilon < \mu_i) \leq e^{-2T_i(t)\varepsilon^2},$$

where ε is the upper bound. By setting $\varepsilon = \sqrt{2\log(N)/T_i(t)}$, we get:

$$\mathbb{P}(\bar{X}_{i,t} + \varepsilon < \mu_i) \leq N^{-4},$$

so that the probability quickly converges to zero as the number of rounds N grows.[1] The pseudo-code of UCB1 is presented in Algorithm 1, where $\bar{x}_{i,t}$ is

[1] With log we refer to the natural logarithm.

Algorithm 1: UCB1

```
Input: $N$ time horizon, $A$ arm set
for $t \in \{1, \dots, K\}$ do
    Play arm $a_t$ and observe $x_{t,1}$
for $t \in \{K+1, \dots, N\}$ do
    for $i \in \{1, \dots, K\}$ do
        Compute: $u_{i,t} = \bar{x}_{i,t} + \sqrt{\frac{2\log(t)}{T_i(t-1)}}$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} u_{i,t}$ and observe $x_{i_t, T_{i_t}(t)}$
```

the empirical mean reward of each arm a_i after t rounds. Note that if an arm is not pulled, its upper bound grows logarithmically in the number of rounds. This means that an arm will never be permanently discarded, no matter how poorly it performs.

In [6], the authors prove the following theorem:

Theorem (Auer et al., 2002 [6]). *If policy UCB1 is run over a stochastic MAB setting, the pseudo-regret after N rounds is at most:*

$$\bar{R}_N \leq 8 \sum_{i:\mu_i<\mu^*} \frac{\log N}{\Delta_i} + \left(1 + \frac{\pi^2}{3}\right)\left(\sum_{j=1}^{K} \Delta_j\right)$$

where $\Delta_i = \mu_{i^*} - \mu_i$.

The first term of the sum states that any suboptimal arm is pulled only a logarithmic number of rounds, also depending on how hard it is to distinguish from the optimal arm. The smaller the Δ_i, the higher the number of pulls required to know that arm i is suboptimal, and hence the higher the regret. The second term represents a constant number that caps the number of rounds we will pull suboptimal arms in excess of the first term. This is a worst-case upper bound on the pseudo-regret. Simply, UCB1 cannot do worse than this, but it does not mean it always achieves this much regret. In their seminal work [8], Lai and Robbins show that, for Bernoulli reward distributions, the lower bound for the pseudo-regret of any policy is logarithm in the number of rounds. For $p, q \in [0, 1]$, denote by $\mathrm{kl}(p, q)$ the Kullback-Leibler divergence between a Bernoulli of parameter p and a Bernoulli of parameter q, defined as:

$$\mathrm{kl}(p, q) = p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q}.$$

Lai and Robbins prove the following theorem:

Theorem (Lai and Robbins, 1985 [8])**.** *Consider a strategy that satisfies $\mathbb{E}[T_i(N)] = o(N^a)$ for any set of Bernoulli reward distributions, any arm a_i with $\Delta_i > 0$, and any $a > 0$. Then, for any set of Bernoulli reward distributions the following holds:*

$$\liminf_{n \to +\infty} \frac{\bar{R}_N}{\log N} \geq \sum_{i:\Delta_i > 0} \frac{\Delta_i}{\mathrm{kl}(\mu_i, \mu_{i^*})}.$$

For all the UCB-like policies, using Pinsker's inequality:

$$2(p-q)^2 \leq \mathrm{kl}(p,q) \leq \frac{(p-q)^2}{q(1-q)},$$

we have the result that there exist two constants $K_1 > 2$ and $K_2 > 0$ such that for every suboptimal arm a_i:

$$\bar{R}_N \leq \frac{K_1}{\Delta_i^2} \log N + K_2.$$

Thus, UCB1 asymptotically matches the lower bound on the regret for the MAB problem. This means that, theoretically, the achieved regret cannot be improved, except for the constants.

In the very first paper on the MAB problem [9], a simple strategy was proposed, the so-called Thompson Sampling (TS). In Algorithm 2, we present the pseudo-code of TS. Assume to have a prior $\pi_{i,0}$ on each reward expected value $\mu_{i,t}$ and let $\pi_{i,t}$ be the posterior distribution for the parameter $\mu_{i,t}$ after t rounds. For instance, in the case of Bernoulli reward, we consider a uniform uninformative prior and we choose $\pi_{i,0} :=$ Beta$(1,1)$, where we denote with Beta(a,b) the Beta distribution with parameters a and b. The posterior becomes $\pi_{i,t} :=$ Beta$(S_i(t)+1, T_i(t) - S_i(t) + 1)$, where $T_i(t)$ is the number of times the arm a_i has been pulled in the first t rounds, and $S_i(t) := \sum_{m=1}^{t} x_{i,m} \mathbb{1}\{\mathfrak{U}(h_m) = a_i\}$ is the cumulative reward of the arm a_i in the first t rounds. Let $\theta_{i,t}$, also known as *Thompson sample*, denote a sample from $\pi_{i,t}$. TS is the algorithm which at time t selects the arm with the highest Thompson sample $\theta_{i,t}$.

Recently there has been a surge of interest for this policy, mainly because of its flexibility to incorporate prior knowledge on the arms. The first asymptotically optimal finite-time analysis of Thompson Sampling has been proven only recently in [10] for Bernoulli distributed rewards in stationary settings. The authors show that the asymptotic pseudo-regret of the algorithm matches the asymptotic rate for general MAB given by [8]. This analysis is extended in [11] to a more general class of distributions.

Algorithm 2: Thompson Sampling

```
Input: $\{\pi_{i,0}\}_i$ prior distributions, $N$ time horizon, $A$ arm set
for $t \in \{1, \dots, N\}$ do
    for $i \in \{1, \dots, K\}$ do
        Compute $\pi_{i,t} = \text{Beta}(S_i(t) + 1, T_i(t) - S_i(t) + 1)$
        Sample $\theta_{i,t}$ from $\pi_{i,t}$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} \theta_{i,t}$ and observe $x_{i_t, T_{i_t}(t)}$
```

We introduced the classical MAB setting, along with two of the most famous policies: UCB1, a *frequentist* algorithm, and Thompson Sampling, a *Bayesian* one. The goal of a frequentist algorithm is to achieve the best parameter-dependent performance, and the expected mean rewards corresponding to the arms are considered as unknown deterministic quantities. In contrast, in the Bayesian approach, each arm is characterized by a parameter which is related to a prior distribution.

In MAB literature, there are several policies which tackle different setting and analyze some important variants and extensions of the presented algorithms. In the next chapter, we give a more detailed review of the state-of-the-art policies related to our work. We refer the interested reader to [12] for more details on the MAB formulation.

CHAPTER 3

General Overview

3.1 Proposed Solution Architecture

Our work addresses the pricing problem in the setting of online sales of digital goods from the point of view of an e-commerce, such as an Online Travel Agency (OTA), selling its products in an environment where it is not possible to use information about the customers, like the metasearch environment. We study the problem of optimal pricing, that is the search for the pricing strategy that maximizes the expected profit of the OTA. Metasearch engines send a lot of traffic, then resulting in sales, to OTAs' websites. This scenario presents a great profitability, but also a number of characteristics which make the problem very challenging, both from a scientific and a practical point of view. In the specific, we have a vast catalog of items to price, which are all the possible flights tickets. We have almost no information about our customers since users do not directly use our websites but they go on metasearch engines, which effectively act as middlemen (Figure 3.1). Even if they do send a lot of traffic to OTAs' websites, they give OTAs very few information about customers' behavior and purchasing history, making users identification possible only after the purchase. Therefore, the pricing scheme cannot rely on personal customer information that we may have collected in other

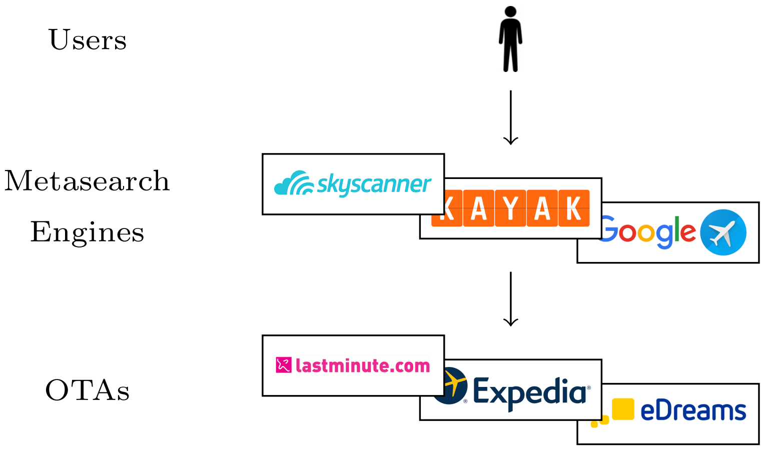


Figure 3.1: *Schema of the relation between metasearch engines and OTAs. The user performs the search over a metasearch engine. The search is then passed to all the OTAs, which return their prices. The metasearch engine aggregates all the results and finally show them to the user, which can make a clear comparison of the different OTAs prices. Metasearch engines effectively act as middlemen, sending traffic to OTAs but without giving them information about the customers.*

setting, such as with the direct selling of the product on our website to a registered user. Furthermore, we have a huge amount of searches performed every day on metasearch engines, but only a few of them actually convert into bookings, since most of the users perform searches without the actual intent of buying, but only for informational purposes. This makes the problem of building a user model and learning the optimal price very hard and long in time. Another difficulty arises from the fact that the environment is non-stationary since we have seasonal effects on the market and we have a lot of competitors which impact on the non-stationarity of the environment by changing their marketing strategies.

Formally, the pricing problem is characterized by a price which is associated with a revenue known by the seller and a conversion rate, measuring the probability that the item will be sold at a given price, which, instead, is unknown to the seller. As mentioned above, the behavior of the users may be non-stationary, thus the average conversion rate may change over time. Extremely low conversion rates, as customary in this setting, make the estimation process excessively long. As a result, the estimation process rarely converges to stable solutions, and it is in a transient for most of the time. Therefore, the effectiveness of a pricing strategy mainly depends on its performance during the transient, and this makes the problem of finding the best

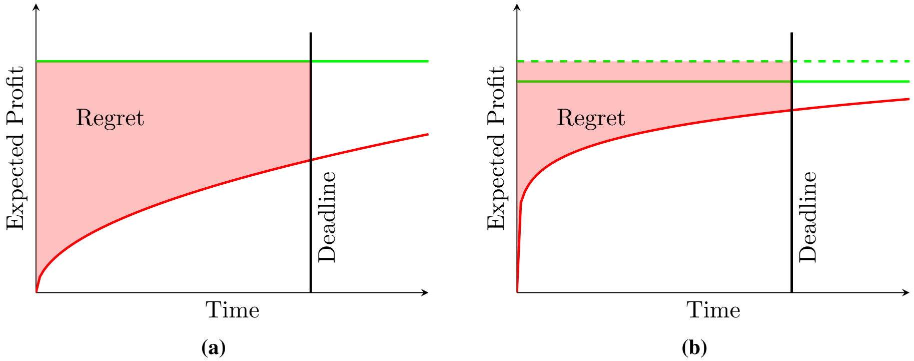


Figure 3.2: *The graphs roughly describe the profit made by a seller which uses bandit algorithms to price his items, in the case he applies (b) or not (a) clustering techniques. In (b), the number of feedbacks collected before the deadline is higher, thus the bandit algorithm is able to reach better profit performances and to reduce its regret. However, the bandit algorithm will not converge to the profit of (a), due to a loss in precision. Nevertheless, the regret made in (b) is still lower than the one made in (a).*

price an online learning problem.

In an online learning problem, a learner chooses at each round an action and observes the reward associated with the actions. The goal of the learner is to accumulate as much reward as possible over a sequence of rounds while minimizing the loss incurred from choosing sub-optimal actions. The Multi-Armed Bandit (MAB) problem is a form of online learning that perfectly matches the characteristics of our setting. In a MAB, the learner has a finite set of available actions. At each round, he can only choose one action to play and, at the end of the round, he can only see the reward generated from the chosen action. In our case, the actions are the prices the seller can set on an item, and each round is a request, or a search, made by a user. When a user performs a search, the seller shows the chosen price, and he collects the feedback from the user in the form of "item bought" or "item not bought".

In general, the best strategy to maximize the expected profit for the seller would be to set a different price for each single item of the catalog. In our specific problem, the solution is not so straightforward. We have a vast catalog of product to price and the number of feedback we are able to collect for most of them is very low. So, if we set a different price for each item, it would take very long time to collect samples for the learning and to converge to the optimal price. Moreover, we are in a non-stationary environment, so, if the convergence is too slow, it could happen that we do not converge to the optimal price before it actually changes. For these reasons, we use clustering techniques to aggregate items into contexts, which are set of items sharing similar features, and then we apply bandit algorithms to each of the context. Clustering items and setting one price for each context will lead to a loss of precision, but, on the other hand, we can collect more samples and to converge in shorter time. Figure 3.2 gives an idea of this behavior. The profit made when the seller sets a different price on each item is described in Figure 3.2a. The green line is the profit made by an oracle which at each turn knows the optimal price to set on an item, while the red line is the profit made by a bandit algorithm, which finally converges to the optimal profit. The black line is the deadline, that is the turn in which the optimal price changes due to the non-stationarity of the environment. The colored area between the green and red lines is called *regret*, which is the loss in which the bandit policy incurs in the learning process. Since the seller can collect only the feedbacks of one single item, the bandit algorithm slowly converges to the optimal price, and it is not able to reach good profit performances before the deadline. Conversely, if the seller sets a price on a context, the number of feedbacks collected in the same amount of time is higher since the samples of all the items in the context are aggregated. Thus, the bandit algorithm can reach better profit performance and to reduce the regret, as in Figure 3.2b where the colored area is smaller than the one in Figure 3.2a. However, the bandit algorithm will not converge to the profit we would have had in the case of Figure 3.2a, now specified with a green dashed line: the algorithm will loose precision due to the use of the context and it will converge to the optimal price of the context, that generates the profit specified by the green line. Indeed, the optimal price for a context may not be the optimal price for each item inside the context, leading to an ineluctable loss: the price could be too high for some of the items or could even be increased for some others. Nevertheless, the regret made using clustering techniques is still lower than the one done without using them.

For these reasons, our solution consists in dividing the problem into two sub-problem: a clustering one and an online optimization one.

The first sub-problem is the one of clustering, that is the partitioning of the catalog in contexts of items sharing similar features. This goal has been tackled by learning from historical data collected by the interactions with customers. Some approaches had been proposed in the last years to address this problem but they lack in some of the fundamental characteristics that make an algorithm suitable for practical purposes. First, they did not provide a clear interpretability of the final model since there is no direct method to infer those features that most influence the resulting model. Second, in economics scenarios, it is important that the proposed algorithm should be as risk-averse as possible, but most of the theoretical guarantees available in literature are provided in terms of average value. Finally, existing approaches usually require knowledge of the behavior of the user and assume it to be stationary, assumption which is rarely met in practice in microeconomics scenarios. In this work, we propose a novel algorithm, whose goal is to solve all the mentioned drawbacks of the literature approaches.

The second sub-problem is the study of algorithms to learn the optimal policy that maximizes the expected profit of each context. As mentioned before, we study online learning techniques, in particular the Multi-Armed Bandit ones, which have been widely studied in literature. It is possible to use existing general-purpose algorithms to solve our problem, but, by exploiting the pricing structure, we investigate if it is possible to improve the performance of the classical algorithms. There are several works in literature which take into account one of the characteristics of our problem, but they usually deal with only one feature to exploit at time. In this work, we propose novel algorithms which exploit specific properties of the problem, some of them already studied in literature, other unexplored so far, that we introduce in the next section.

The two sub-problems are thus interconnected and continuously communicating since the data generated from the interaction of the users with the optimization algorithms are collected and passed through the clustering algorithm to update the contexts model and improve the performance of the system. This is the architecture of the automatic pricing system we designed in collaboration with one of the major Online Travel Agencies in Europe.

3.2 Related Works and Original Contributions

The main focus of the thesis is on bandits techniques. The Multi-Armed Bandit (MAB) setting [6] models the sequential decision-making problem, addressing the well-known exploration-exploitation trade-off, as introduced in Chapter 2.

As described before, we study different properties at the same time. In literature, to the best of our knowledge, there are no other works which propose a solution taking into account all the aspects of our problem. For this reason, in this section, we separately describe each of the features we exploit, and we mention the state-of-the-art studies related to that feature.

3.2.1 Monotonicity and low conversion rate

The main work dealing with bandits for pricing is provided in [13], where the authors study the value of knowing the demand curve in stationary settings and provide a tight regret-bound analysis. Furthermore, the authors present an algorithm to select a finite set of arms to which each state-of-the-art MAB algorithm can be applied. Here, we focus on a different problem, specifically by exploiting two properties unexplored in the literature so far.

First, we exploit the monotonicity of the conversion rate in the price. More precisely, the conversion rate is monotonically decreasing in the price, and this follows from the fact that the customer demand is monotonically decreasing in the price when no network externalities are present [14]. This means that, each time a buyer makes a purchase at a given price, we may infer that the sale would have been made at any lower price, and, *vice versa*, each time the buyer refuses to buy at a certain price, we may infer that all the higher prices would not have been accepted too. Notice that this property is common also in many other application domains, thus making our algorithms applicable in settings different from the pricing one. For instance, in multi-slot online advertising, where it is necessary to estimate the Click-Through Rates (CTRs) of ads [15] and the expected value of the CTR of an ad monotonically decreases from the slot in the top to the one in the bottom; and in bandwidth allocation, where it is necessary to estimate the best packet size for the link between some servers [16] and, if a packet has been successfully transmitted, also a smaller one would have been received too.

Second, we exploit the fact that e-commerce sellers have *a priori* information about the customer behavior, coming from past transactions. This information is not usually sufficient for producing pre-estimates sufficiently accurate to avoid a cold start of the learning algorithms since sellers pulled in the past a very limited number of arms. However, such information is sufficient to estimate a lower bound to the percentage of the buyers that are only interested in checking the price without buying the item, which leads to a low probability of purchasing a good [17]. Indeed, it is common that human users check the price for some days before buying an item as well as it is common that companies use bots to check the prices of the competitors frequently. As a result, for every specific pricing setting, associated with a product, we can set an upper bound over the curve of the conversion rate as a function of price. This may allow exploiting tighter concentration inequalities, thus reducing the experience needed to get accurate estimates of the expected conversion rate, and, consequently, reducing the loss due to the algorithm exploration.

Several previous works exploit the structure of specific classes of sequential games to improve the performance provided by the general-purpose algorithms. In [18, 19], the authors study a graph model for the arm feedback in an adversarial setting under the assumption that the realizations are correlated and that this correlation is known. The treatment of this last assumption is different from the treatment of the monotonicity assumption, where, conversely, the correlation is over the expected value of the arms and not over the realizations. In [20, 21], the authors propose a more general setting named partial monitoring games, for which several studies on asymptotic regret bounds have been produced in the last decade both in stochastic [22, 23] and adversarial [16, 21, 24] settings. Other similar works study the problem of dark pools [25, 26], that are recent type of stocks exchange designed to facilitate large transactions, in which a key aspect is the censored feedback that the trader receives. To the best of our knowledge, no work takes advantage of the monotonicity property as defined above or exploits *a priori* information about the magnitude order of low conversion probabilities.

In the economics literature and, more precisely, in the sub-area of learning and earning, several works study the pricing problem [27, 28, 29, 30]. Most of these works assume that *a priori* information on the structure of the problem is available (e.g., on the product supply availability or the user behavior). More specifically, [27] deals with a limited initial inventory of a single product and designs a parametric and a non-parametric algorithm to estimate the demand function. Several works propose techniques to learn the optimal price under the assumption that the expected revenue curve has a unique global optimal solution [29, 30, 31]. Finally, [32] studies the case of an adversarial model for the user in an online posted-price auction and directly applies the Exp3 algorithm [33] to minimize the regret. Remarkably, most of the works in the *learning and earning* field do not provide any theoretical guarantee on the regret bounds. Even if heuristic algorithms might perform better than the algorithms with theoretical guarantees, the lack of worst-case guarantees discourages their employment in practice.

A problem related to pricing is the design of nearly-optimal auctions in the case the bidders' valuations are drawn from an unknown distribution [34, 35]. The proposed solution relies on statistical learning theory techniques to compute the number of samples required to bound the distance of the approximated solution from the real expected revenue.

In this thesis, we propose techniques to exploit the monotonicity property of conversion rates as well as the *a priori* information on the maximum conversion rate. Our techniques can be paired, in principle, with any MAB algorithm. We tailor our techniques for two main Upper Confidence Bound (UCB) like algorithms working in stationary settings: UCB1 [6], being the most popular and basic MAB algorithm, and UCBV [36], being one of the UCB-like algorithms with the best empiric performance. We prove that the asymptotic regret bounds of our algorithms are of the same order as UCB1 and UCBV. We present a thorough experimental evaluation of our algorithms in several different configurations based on real-world data, comparing our algorithms with the main general-purpose frequentist stochastic MAB policies and showing that exploiting the two properties mentioned above allows one to improve the profit significantly. The empirical analysis shows that our algorithms provide significant advantages with respect to general-purpose MAB algorithms in the early stages of the learning process. This is crucial in real pricing scenarios, where very low conversion rates (that require a long exploration phase to have accurate estimations) and non-stationary buyers' demands make the algorithms to work in a never-ending transient.

3.2.2 Unimodality

In the so-called Unimodal MAB (UMAB), introduced in [37], each arm corresponds to a node of a graph, and each edge is associated with a relationship specifying which node of the edge gives the largest expected reward (providing thus a partial ordering over the arm space). Furthermore, from any node, there is a path leading to the unique node with the maximum expected reward along which the expected reward is monotonically increasing. While the graph structure may be (not necessarily) known *a priori* by the UMAB algorithm, the relationship defined by the edges is discovered during the learning.

Models presenting a graph structure have become more and more interesting in last years due to the spread of social networks. Indeed, the relationships between the entities of a social network have a natural graph structure. A practical problem in this scenario is the targeted advertisement problem, whose goal is to discover the part of the network that is interested in a given product. This task is heavily influenced by the graph structure since in social networks people tend to have similar characteristics to those of their friends (i.e., neighbor nodes in the graph). Therefore interests of people in a social network change smoothly and neighboring nodes in the graph look similar to each other [38, 39]. More specifically, an advertiser aims at finding those users that maximize the ad expected revenue (i.e., the product between click probability and value per click), while at the same time reducing the amount of times the advertisement is presented to people not interested in its content. The unimodal structure also occurs in the sequential pricing problem, as described in [40].

Under the assumption of unimodal expected reward, the learner can move from low expected rewards to high ones just by climbing them in the graph, preventing the need for a uniform exploration over all the graph nodes. This assumption reduces the complexity in the search for the optimal arm since the learning algorithm can avoid to pull the arms corresponding to some subset of non-optimal nodes, reducing thus the regret. Other applications might benefit from this structure, e.g., recommender systems which aim at coupling items with those users are likely to enjoy them. Similarly, the use of the unimodal graph structure might provide more meaningful recommendations without testing all the users in the social network. Finally, notice that problems like bidding in online sponsored search auctions [41] and single-peak preferences economics and voting settings [14], are graph-structured problems in which the graph is a line.

Frequentist approaches for UMAB with graph structure are proposed in [40] and [37]. In [40], the authors introduce the GLSE algorithm with a regret of order $O(\sqrt{T}\log(T))$. However, GLSE performs better than classical bandit algorithms only when the number of arms is $\Theta(T)$. Combes and Proutiere [37] present the OSUB algorithm, based on KLUCB, achieving asymptotic regret of $O(\log(T))$ and outperforming GLSE in settings with a few arms.

Interestingly, the assumptions of monotonicity, described in the previous section, and unimodality are orthogonal, none of them being a special case of the other one and, therefore, the results known for unimodal bandits cannot be directly adopted in monotonic settings.

Some works deal with unimodal reward functions in continuous armed bandit setting [40, 42, 43]. In [40] a successive elimination algorithm, called LSE, is proposed achieving regret of $O(\sqrt{T}\log T)$. In this case, assumptions over the minimum local decrease and increase of the expected reward is required. Combes and Proutiere [42] study stochastic bandit problems with a continuous set of arms and where the expected reward is a continuous and unimodal function of the arm. They propose the SP algorithm, based on the stochastic pentachotomy procedure to narrow the search space. Unimodal MAB on metric spaces is studied in [43].

An application-dependent solution to the recommendation systems which exploits the similarity of the graph in social networks in targeted advertisement has been proposed in [44]. Similar information has been analyzed in [45] where the problem of cold-start users (i.e., new users) is studied. In [18, 19], a graph structure of the arm feedback in an adversarial setting is investigated. In the specific, they assume to have correlation over rewards and not over the expected values of arms.

In this thesis, we propose a novel algorithm relying on the Bayesian learning approach for a generic UMAB setting. We derive a tight upper bound over the regret, which asymptotically matches the lower bound for the

UMAB setting. We provide a wide experimental campaign showing better performance of our algorithm in applicative scenarios than those of state-of-the-art ones, also evaluating how the performance of our algorithm and the state-of-the-art ones varies as the graph structure properties vary.

3.2.3 Non-stationarity

Non-stationarity behaviors are common in real-world applications and play a prominent role in Internet economics, for the reasons mentioned before. Other forms of non-stationarity in the pricing problems may be due to a new product invading the market: the price maximizing the expected profit of a product already present in the market may change abruptly when a newer product enters [46]. Non-stationarity is common also in many other application domains. We recall that the former motivation for MAB settings argued in [9] was the study of clinical trials, where different treatments are available, and a learner aims at selecting the one to use for the next patient. Although in its original formulation the clinical trial scenario was assumed stationary over time, in the real world it may be not. Indeed, the disease to defeat may mutate over time, thus a treatment that initially is optimal might subsequently slowly decrease its effectiveness and another treatment, which initially was ineffective, might become the best option [47]. Another example is untruthful auction mechanisms for search advertising (e.g., the GSP used by Google and Bing [48]), where advertisers try to learn the best bid to obtain their ad displayed in some profitable slot. Here, non-stationarity may be due to the arrival and departure of advertisers which change the profitability of the slots [49].

Differently from the classical stochastic MAB setting, in Non-Stationary stochastic MAB (NS-MAB) settings the expected reward of each arm may change over time, thus potentially changing the optimal arm. As stressed in [50], general-purpose classical MAB algorithms are not suitable when tackling NS-MAB settings, their regret bounds not holding anymore. In non-stationary settings, some frequentist algorithms with theoretical guarantees are known [51, 37, 52, 53, 54], whereas, to the best of our knowledge, all the Bayesian methods are only based on heuristics [55, 56].

NS-MAB settings have been receiving attention in the scientific community only in the last few years. When rewards may change arbitrarily over time, a non-stationary MAB setting is equivalent to an adversarial MAB one [33]. As a result, the literature mainly focuses on non-stationary MAB settings with some specific structure in the attempt to design algorithms with better regret bounds.

Some frequentist algorithms with theoretical guarantees are known [51, 37, 52, 53, 54]. In [52], the authors study abruptly changing MAB settings and present the SW-UCB algorithm achieving an $\tilde{O}(\sqrt{N})$ regret.[1] In economic domains, an abrupt change can be due to the invasion of the market by a new product. The same setting is tackled in [57], where the authors present the SER4 algorithm which empirically outperforms the SW-UCB one. In [37], the authors present SW-KL-UCB, a policy working in a smoothly changing MAB settings. In this non-stationary setting, the regret is upper bounded by $\tilde{O}(\frac{N}{\tau})$, being τ the length of a sliding window used by the algorithm and N the learning horizon.

In [51], the authors study a non-stationary MAB setting under the assumption that the total variation of the expected rewards over whole the time horizon is bounded by a budget that is *a priori* fixed. They provide a distribution-independent lower bound and they propose the REXP3 algorithm, a near-optimal frequentist algorithm with a regret of order $O(N^{2/3})$. In [58], the authors focus on the dynamic bandit setting, a special case of the restless bandits, in which the reward distribution of the arms changes at each round according to an arbitrary stochastic process. The authors propose algorithms that minimize the per-round regret over an infinite time horizon.

The MAB literature provides also some works that exploit MAB techniques as heuristics on applicative scenarios without providing theoretical guarantees. To cite a few, in [55], the authors propose a Bayesian algorithm for the specific case of non-stationary normal bandit settings; in [56], the authors analyze an NS-MAB where the probabilities according which the expected value of the arms change are *a priori* fixed, and propose the CTS algorithm that combines Thompson Sampling together with a change point detection mechanism; in [59], the authors propose a variant of Thompson Sampling which can be used in both rested and restless bandit non-stationary scenarios; in [60], the authors present an evolutionary algorithm to deal with generic non-stationary environments which empirically outperforms classical solutions.

Other settings, closely related to the MAB one, have also been studied in the presence of non-stationaries. For instance, in [54] the authors present a study of the regret in the case of non-stationary stochastic experts, providing a bound of order $O(N^{1/3})$ in the case we assume a constant number of switches and limited variance of the expected rewards over time.

In this thesis, we provide frequentist and Bayesian MAB algorithms for non-stationary settings with theoretical guarantees in terms of regret, and we empirically show their superior performance over the state-of-the-art algo-

[1] $\tilde{O}$ denotes a big O notation that ignores logarithmic factors.

rithms.

3.2.4 Contexts Generation

Due to the large availability of logged bandit feedbacks, an LLBF problem rises in a wide range of real-world micro-economic applications, besides ours of pricing. Another scenario where this kind of feedback is easily available is the ad-layout selection problem [61]. In this setting, the performance of a given set of web ads might change, for instance, depending on the specific website the ad is displayed and on the ad size. The available feedback one can have is the click/non-click event only for the displayed layout. Another example in which LLBF problem rises is the scenario of news recommendation [62], in which, given a search query or geolocalization information, a specific article is proposed to the user who can provide a feedback only on the displayed article.

Classical classification (to generate the map between contexts and actions) and regression (to approximate the reward of each pair context-action and then finding the best policy) methods provided by the machine learning literature can be extended to tackle the LLBF problem [63]. However, since they are based on the hypothesis that feedback (usually an error score) for each possible decision is available and this hypothesis is not satisfied in this setting, the extension of these algorithms to the LLBF problem may have a negative impact on their performances (e.g., see [5]). This pushed the need for exploring in the literature novel *ad hoc* approaches. In particular, the state-of-the-art approach for the LLBF problem is the POEM algorithm described in [64, 4]. It is based on the counterfactual risk minimization criterion, where the decision policy is produced taking into account the reward estimates and uncertainties. POEM assumes that the learner knows the stochastic behavior of the user (called logging policy) and that such behavior is stationary, assumptions that are hardly satisfied in real-world settings. Moreover, POEM suffers from two further drawbacks. First, as classical classification and regression methods, it does not allow a clear interpretation of how the decision policy depends on the most relevant context features. Such an interpretation is crucial when a human operator inspects and uses the policy. Second, in POEM, the generation of the policy is driven by the minimization of the so-called counterfactual risk. This approach does not explicitly take into account the risk aversion of the learner, nor provides a given confidence over the expected profit of the proposed algorithm.

The online version of the LLBF problem has been addressed in the literature with the name of contextual Multi-Armed Bandit (MAB) [65, 66].

For instance, in [65] the authors analyze the problem of contextual bandits assuming that the reward is a linear combination of the context vector and propose a modification of the LinUCB algorithm whose regret upper bound matches the lower bound up to logarithmic factors. In [66], the authors develop an algorithm based on random forest which presents promising results in comparison with the state-of-the-art ones. The techniques developed for the contextual MAB cannot be applied in LLBF scenario since they prescribe a non-stationary policy aimed at solving the so-called exploration/exploitation dilemma, while our proposed algorithm focuses on the maximization of the profit given the available logs.

In this thesis, we propose a novel method for the LLBF problem, whose main novelties are the construction of a decision tree providing a risk-averse stationary policy for future decision by means of the maximization of statistical lower bounds over the action rewards (as suggested for bandit feedbacks in [67]) and an easy way to compute an estimate (in high probability) of the expected profit provided by the tree. Our method also provides a clear interpretability of the resulting decision tree, useful for business analysis, which allows to easily identify the most relevant features for the definition of the contexts. We performed an experimental evaluation of our algorithm, showing empirical evidence of the effectiveness of the proposed approach when compared to state-of-the-art techniques (including classification and regression methods).

Since the main idea of our algorithm lies in the construction of a decision tree for the LLBF problem, it has been influenced by those works using decision trees to solve classification and regression problems either in offline [68] or online [69, 70] fashion. The aforementioned approaches are used to address the problems with complete feedback and cannot be directly applied to solve the LLBF problem. Nonetheless, their construction shares some similarities with the algorithm we propose. For instance, both our algorithm and the one in [71] make use of confidence bounds to determine the best split to build a decision tree. Furthermore, to the best of our knowledge, there are no works addressing the risk-aversion paradigm directly in the field of decision trees.

CHAPTER 4

Learning from Logged Bandit Feedback

In this chapter, we consider the Learning from Logged Bandit Feedback (LLBF) problem and we propose an algorithm specifically shaped for the this setting, based on a risk-averse learning method which exploits the joint use of regression trees and statistical confidence bounds. Differently from existing techniques developed for this setting, our algorithm generates policies aiming to maximize a lower bound on the expected reward and provides a clear characterization of those features in the context that influence the process the most. Section 4.1, provides the formulation for the considered LLBF setting. In Section 4.2, we describe the proposed algorithm. Finally, in Section 4.3, we provide a wide experimental campaign over both synthetic and real-world datasets showing empirical evidence that the proposed algorithm outperforms both state-of-the-art machine learning classification and regression techniques and existing methods addressing the LLBF setting.

4.1 Problem Formulation

Consider an LLFB setting defined as the tuple $(\mathcal{X}, A, R)$, where $\mathcal{X} = (X, \mathcal{D})$ is a finite-dimensional multivariate probability space of contexts with support in $X \subseteq \{0,1\}^c$ with $c \in \mathbb{N}$ and unknown multivariate distribution $\mathcal{D}$,

$A := \{a_1, \dots, a_K\}$ with $K \in \mathbb{N}$ is the finite action space, and R is the reward distribution.[1] A generic sample $z_i = (x_i, a_i, r_i)$ has a context vector $x_i = (x_{i1}, \dots, x_{ic}) \in X$, which is drawn from the distribution $\mathcal{D}$, i.e., $x_i \sim \mathcal{D}$. The corresponding action $a_i \in \mathcal{A}$ is chosen by a generic sampling policy $\mathfrak{U}_0$, i.e., $a_i \sim \mathfrak{U}_0$, which is assumed to be unknown.[2] Finally, the reward r_i gained by selecting action a_i in the context x_i is the realization of a random variable $R(x_i, a_i)$ with unknown distribution $\mathcal{R}(x_i, a_i)$ and finite support $\Omega \subset \mathbb{R}$ (w.l.o.g. from now on we consider $\Omega \subseteq [0, 1]$) provided for the chosen action a_i in the chosen context x_i, i.e., $R(x_i, a_i) \sim \mathcal{R}(x_i, a_i)$.[3] We denote with $\mu(x_i, a_i)$ the expected value of the reward $R(x_i, a_i)$, i.e., $\mu(x_i, a_i) := \mathbb{E}\left[R(x_i, a_i)\right]$, where the expected value is computed over the distribution $\mathcal{R}(x_i, a_i)$.

A policy (or mapping) $\mathfrak{U}$ dealing with the LLBF problem is a function (either deterministic or stochastic) providing for each context $x \in X$ the choice of the action $a \in A$, i.e., $\mathfrak{U}(x) = a$. The performance of a policy $\mathfrak{U}(\cdot)$ over a generic LLBF problem $(\mathcal{X}, A, R)$ can be evaluated by means of its the expected *profit*, defined as:

$$P(\mathfrak{U}) = \mathbb{E}\left[R(x, a)\right],$$

where the expectation is taken w.r.t. the considered policy $\mathfrak{U}$ and the reward distributions $\{\mathcal{R}(x, a)\}_{x \in X, a \in A}$.

In real-world applications a finite dataset $Z_N = \{z_1, \dots, z_N\}$ of $N \in \mathbb{N}$ logged bandit feedbacks is provided to learn a policy maximizing $P(\mathfrak{U})$. The maximization of the empirical expected profit is not a sufficient criterion to design the decision policy [14], since it might be arbitrarily distant from the real expected profit value. An alternative approach, commonly used in economic scenarios, is to consider some lower bound over the regret [67]. This takes into account the uncertainty that affects the actual profit, but sacrifices part of the expected profit by choosing policies that minimize the probability of realizing a very low profit or even a loss. Nonetheless, many companies prefer to choose such an approach, their strategy being risk-averse. Formally, a risk-averse variant of the estimator $\underline{P}(\mathfrak{U}, Z_N, \delta)$ of the expected profit $P(\mathfrak{U})$, called *risk-averse profit*, computed over a dataset Z_N with an overall confidence of $\delta \in (0, 1)$ satisfies:

$$\mathbb{P}\left(P(\mathfrak{U}) \leq \underline{P}(\mathfrak{U}, Z_N, \delta)\right) \leq \delta,$$

[1] The choice of a binary context space is carried here for sake of notation. The case of $X \subseteq \mathbb{R}^c$ is discussed in Section 4.2.2.

[2] Here, we focus on sampling policies not depending on the context. Suggestions about the extension to the contextual policies case, i.e., $U_0 = U_0(x)$, will be discussed in Section 4.2.2.

[3] The extension of this framework to the case we receive a loss function instead of rewards is straightforward. For sake of concision, here we will only consider the reward as feedback.

The aforementioned risk-averse profit should be considered as figure of merit to evaluate the performance of algorithms for LLBF in risk-averse scenarios.

4.2 Proposed Method

In principle, we aim at finding the best decision policy, i.e., the mapping $\mathfrak{U}(x) := a^*$ between each context $x \in X$ and actions in $a^* \in A$ maximizing one of the aforementioned figures of merit, while keeping the risk as low as possible. In practice, one faces two main challenges to find such a mapping. First, if we had an infinite number of samples and therefore we are able to compute the expected reward $\mu(x, a)$ for each couple context-action exactly, the problem could be formulated as a combinatorial optimization problem which is not tractable with an exhaustive approach. More specifically, since the number of possible contexts is usually much larger than the number of available actions ($2^c > K$), the same action may be taken in multiple contexts. Therefore, by denoting with X_{a_i} the set of contexts where we choose the action a_i, i.e., $X_{a_i} = \{x \in X \mid a_i = \arg\max_{a \in A} \mu(x, a)\}$, and with $\mathcal{P} := \{X_{a_1}, \dots, X_{a_K}\}$ a partition of the space X ($\cup_{i=1}^{K} X_{a_i} = X$ and $X_{a_i} \cap X_{a_j} = \emptyset \; \forall i \neq j$), finding the policy which maximizes the expected profit becomes equivalent to finding the partition $\mathcal{P}$. Unfortunately, the number of possible partitions is exponential in the number of contexts (that in its turn is exponential in the number of features c), thus finding the optimal partition is computationally hard. Second, in a real scenario the values of the expected reward $\mu(x, a)$ for each couple context-action are unknown and should be estimated from a finite dataset Z_N. In this case, the use of the empirical mean to estimate $\mu(x, a)$ is not sufficient. In fact, due to the uncertainty present in the reward distributions, the empirical mean value might deviate from the real values, thus its use does not provide any guarantee that the optimization procedures also minimizes the probability of having very low profit. We remark that this holds since the empirical mean does not capture the risk-aversion of the learner.

We propose the RADT algorithm which greedily learns a binary tree to approximate the optimal mapping $\mathfrak{U}(x)$ on the basis of the available dataset Z_N and maximizes statistical lower bounds over the expected profit. This approach exploits the fact that similar contexts, i.e., contexts that differ only for few features, are likely to share the same optimal action and thus it keeps the complexity of the optimization procedure at bay by considering only a linear number of possible contexts (w.r.t. the feature space dimension c) during the process of creating the partitions. Furthermore, to cope with the uncertainty present on the reward, as discussed in the previous section, the algorithm makes use of a risk-averse approach by considering statistical lower bounds over the expected reward, which provide guarantees on the expected rewards in high probability.

More specifically, the decision tree is built by choosing in each node the split that guarantees the maximum improvement w.r.t. the lower bound on the expected profit of the data corresponding to that node. When no improvement can be obtained the node becomes a leaf and it gets associated to the action corresponding to the maximum value of the lower bound on the expected reward. The lower bound over the expected profit of an action a over a generic dataset Z (subset of Z_N) holding with probability δ is:

$$\underline{G}(Z, a, \delta) = \underline{p}(Z, \delta/2) \cdot \underline{\mu}(Z(a), \delta/2), \tag{4.1}$$

which is computed as the product of two terms:

- a lower bound over the probability of the context corresponding to dataset Z:
  $$\underline{p}(Z, \delta/2) := \hat{p}(Z, Z_N) - \varepsilon_1(Z, \delta/2),$$
  where $\hat{p}(Z, Z_N) := \frac{|Z|}{|Z_N|}$ is the empirical estimate of the probability of a context vector to come from the considered dataset Z w.r.t. the complete one Z_N, $\varepsilon_1(Z, \delta/2)$ is a confidence bound over the previously mentioned quantity holding with probability at least $1 - \delta/2$ and $|\cdot|$ is the cardinality operator;

- a lower bound over the action a expected reward:
  $$\underline{\mu}(Z(a), \delta/2) := \hat{\mu}(Z(a)) - \varepsilon_2(Z(a), \delta/2),$$
  where $Z(a) := \{z_i = (x_i, a_i, r_i), z_i \in Z \mid a_i = a\}$ is the subset of samples $z_i \in Z$ whose action $a_i = a$, $\hat{\mu}(Z(a)) := \frac{S(Z(a))}{|Z(a)|}$ is the empirical estimate of the reward corresponding to action a, $S(Z(a)) := \sum_{i|z_i=(x_i,a_i,r_i),z_i\in Z(a)} r_i$ is the sum of the rewards of action a in the dataset $Z(a)$ and $\varepsilon_2(Z(a), \delta/2)$ is a confidence bound over the expected reward of an action a in the context corresponding to Z holding with probability at least $1 - \delta/2$.[4]

In what follows, we provide a detailed description of the RADT algorithm and, after that, we present some remarks about its characteristics. At last, we describe the possible choices of lower bounds to adopt in different settings.

4.2.1 The RADT algorithm

Algorithm 3: RADT

```
Input: Dataset $Z_N$, Confidence $\delta_0$
Output: Tree root $n_0$
$n_0 \leftarrow$ SPLIT$(Z_N, \delta_0)$
return $n_0$
```

Algorithm 4: SPLIT(Z, δ)

```
for $a \in A$ do
    Extract $Z(a) = \{z_i \in Z | a_i = a\}$
    Compute $\underline{G}(Z, a, \delta)$ as in Eq. (4.1)
Compute $a^* = \arg\max_{a \in A} \underline{G}(Z, a, \delta)$
for $j \in \{1, \dots, c\}$ do
    Extract $Z_j^l = \{z_i \in Z | x_{ij} = 0\}$
    Extract $Z_j^r = \{z_i \in Z | x_{ij} = 1\}$
    for $a \in A$ do
        Extract $Z_j^l(a) = \{z_i \in Z_j^l | a_i = a\}$
        Extract $Z_j^r(a) = \{z_i \in Z_j^r | a_i = a\}$
        Compute $\underline{G}(Z_j^l, a, \delta/4)$ as in Eq. (4.1)
        Compute $\underline{G}(Z_j^r, a, \delta/4)$ as in Eq. (4.1)
    Compute $\underline{H}(Z, j, \delta)$ as in Eq. (4.2)
Compute $j^* = \arg\max_j \underline{H}(Z, j, \delta)$
Compute $\Delta\underline{G} = \underline{H}(Z, j^*, \delta) - \underline{G}(Z, a^*, \delta)$
if $\Delta\underline{G} > 0$ then
    $n^l \leftarrow$ SPLIT$(Z_{j^*}^l, \delta/2)$
    $n^r \leftarrow$ SPLIT$(Z_{j^*}^r, \delta/2)$
    Set $n = (j^*, a^*, n^l, n^r, \delta)$
else
    Set $n = (\emptyset, a^*, \emptyset, \emptyset, \delta)$
return $n$
```

The high level pseudo-code of RADT is presented in Algorithm 3, while a subroutine used by the main algorithm is described in Algorithm 4. Algorithm 3 receives as input the entire dataset $Z_N = \{(x_i, a_i, r_i)\}_{i=1}^N$ of $N \in \mathbb{N}$ logged bandit feedback and a confidence value $\delta_0 \in (0, 1)$ providing the required probability that the policy $\mathfrak{U}_{RADT}$, induced by the RADT algorithm, has risk-averse profit $\underline{P}(\mathfrak{U}_{RADT}, Z_N, \delta_0)$ smaller than the expected profit $P(\mathfrak{U}_{RADT})$.

Algorithm 3 calls Algorithm 4 over the dataset Z_N and with confidence δ_0. The subroutine described in Algorithm 4, called over a generic dataset Z and confidence δ, computes the lower bound $\underline{G}(Z, a^*, \delta)$ of the expected

[4] The choice and the computation of the confidence bounds $\varepsilon_1(\cdot)$ and $\varepsilon_2(\cdot)$ is discussed in Section 4.2.2.

profit of the dataset Z as described in Equation (4.1) and compares it with the one obtained by summing the lower bounds of the expected profit of the best binary split w.r.t. the j^*-th variable of the dataset Z, called $\underline{H}(Z, j^*, \delta)$.

More specifically, $\underline{G}(Z, a^*, \delta)$ is computed by finding the action $a^* \in A$ maximizing the lower bound of the profit computed over Z (Line 4). After that, to compute $\underline{H}(Z, j^*, \delta)$, the algorithm evaluates the lower bound over the expected profit for all the possible splits along each context dimension. More specifically, a split consists in the partition of the dataset Z into two subsets Z_j^l and Z_j^r (which correspond to two disjoint subsets of the input space X) s.t. all the samples $z_i \in Z$ with $x_{ij} = 0$ belong to Z_j^l and those with $x_{ij} = 1$ belong to Z_j^r (Lines 6-7). The algorithms computes the lower bounds over the expected profit of an action a on the datasets Z_j^l and Z_j^r, called $\underline{G}(Z_j^l, a, \delta/2)$ and $\underline{G}(Z_j^r, a, \delta/2)$, respectively, and, since the overall bound should hold with probability δ, we split evenly the confidence δ over the two subsets (Lines 11-12). For each context dimension index j the algorithm considers the maximum possible gain provided by selecting a single action in each subset, defined as:

$$\underline{H}(Z, j, \delta) := \max_{a \in A} \underline{G}(Z_j^l, a, \delta/2) + \max_{a' \in A} \underline{G}(Z_j^r, a', \delta/2) \tag{4.2}$$

and, finally, it selects the index $j^* := \arg\max_j \underline{H}(Z, j, \delta)$ maximizing the lower bound of the gain (Line 14).

If the lower bound over the gain of the best split $\underline{H}(Z, j^*, \delta)$ provides an improvement over the one of the single node $\underline{G}(Z, a^*, \delta)$, i.e., $\Delta\underline{G} := \underline{H}(Z, j^*, \delta) - \underline{G}(Z, a^*, \delta) > 0$, the algorithm recursively calls the subroutine in Algorithm 4 over the two datasets $Z_{j^*}^l$ and $Z_{j^*}^r$, each one with confidence $\delta/2$, to maintain an overall confidence of δ (Lines 17-18). If no improvement is provided by the best split ($\Delta\underline{G} \leq 0$), the algorithm stops the recursive step.

At the end of the execution the algorithm returns the decision tree having in each node n the information about the index of the context dimension j^* where the split has occurred, about the optimal action a^* in the node, about the two children n^l and n^r of the contexts corresponding to datasets $Z_{j^*}^l$ and $Z_{j^*}^r$, respectively, and the confidence δ we considered for the node corresponding to Z_N (Line 19). If the node is a leaf, it only contains the best action a^* and the confidence level of the node δ (Line 21).

4.2.2 RADT properties and limitations

Differently from other techniques for the LLBF setting, the RADT algorithm provides a natural way of computing the risk-averse profit $\underline{P}$. The output of RADT algorithm run over Z_N with confidence δ_0 is a tree structure inducing a partition over the context space $\{X_1, \ldots, X_e\}$ and each leaf n_k of the tree corresponds to a context X_k. Let us consider a dataset Z_M of logged bandit feedbacks, independent from Z_N and sampled by the sampling policy U_0 over $(\mathcal{X}, A, R)$. Each node n_k can be coupled with the portion of the dataset $Z(n_i) \subseteq Z_M$ s.t. $z_i \in Z(n_k)$ iff $x_i \in X_k$. The risk-averse profit of the policy $\mathfrak{U}$ (resulting from the RADT training) can be computed as follows:

$$\underline{P}(\mathfrak{U}, Z_M, \delta_0) = \sum_{i=1}^{e} \underline{p}_i \underline{\mu}_i,$$

where $\underline{p}_i := \hat{p}(Z(n_i), Z_M) - \varepsilon_1(Z(n_i), \delta(n_i))$ is the lower bound over the probability of the context X_i, $\underline{\mu}_i = \max_{a \in A} \hat{\mu}(Z(n_i, a)) - \varepsilon_2(Z(n_i, a), \delta(n_i))$ is the lower bound over the expected reward of context X_i, $Z(n_i)$ and $\delta(n_i)$ are the portion of the dataset Z_M and the confidence corresponding to node n_i, respectively, and $Z(n_i, a) = \{z_i \in Z(n_i) | a_i = a\}$. It is straightforward to proof, by considering a union bound argument, that the r.h.s. of the above equation satisfies the properties of the risk averse profit, i.e., that $\mathbb{P}\left(P(\mathfrak{U}) \leq \underline{P}(\mathfrak{U}, Z_M, \delta_0)\right) \leq \delta_0$.

Besides giving guarantees on the minimum expected reward, the RADT algorithm also provides a characterization of the context variables. In fact, the ones that have been chosen for the splits are the context variables that are more likely to be relevant for the considered problem. Moreover, we can infer the importance of a single variable by looking at improvement in term of gain $\Delta\underline{G}$ provided by performing a split on it: the more the gain, the more the variable is relevant.

Finally, the extension of the algorithm also to non-binary context variables x_{ij} is quite straightforward [72]. In the case of a finite set of ordered values, a conversion to binary numbers suffices to transform the problem to the LLBF setting considered here. If an attribute is categorical, we instantiate a binary variable for each category. If a context variable has values in a continuous domain, an approximate solution is to discretize the domain and again convert the obtained values into binary variables. This process clearly multiplies the computational complexity of the considered algorithm by a factor proportional to the number of values one considers per variable.

4.2.3 Using different bounds

The algorithm pseudo-code described in Algorithm 3 requires the computation of lower bounds $\varepsilon_1(Z, \delta)$ and $\varepsilon_2(Z, \delta)$ over the contexts probabilities and rewards, respectively. While the expected probabilities present a Binomial distribution, one might use different bounds on the reward due to the different *a priori* information the learner has in the considered setting. In what follows, some of the most interesting choices for the bounds are presented.

If the only information is related to the finite support of the reward distributions, one may resort to classical statistical bounds derived from the Hoeffding [73] or the Bernstein [36] inequalities:

$$\varepsilon_H(Z,\delta) = \sqrt{-\frac{\log\delta}{2|Z|}},$$

$$\varepsilon_{BE}(Z,\delta) = \sqrt{-\frac{2V(Z)\log\delta}{3|Z|}} - \frac{\log\delta}{|Z|},$$

where $V(Z)$ is the estimate of the variance of the rewards r_i s.t. $z_i \in Z$.

If one has also the prior/posterior conjugate distribution, it is possible to use a Bayesian bound, similarly to what considered in [74], e.g., in the case of Bernoulli variables we have Beta/Bernoulli conjugate distributions and:

$$\varepsilon_{BA}(Z,\delta) = q(\delta, \pi(Z)) - \hat{\mu}(Z),$$

where $\pi(Z) = Beta(1 + S(Z), 1 + |Z| - S(Z))$ is the posterior distribution, $S(Z)$ is the cumulative reward over the dataset Z and $q(\delta, \pi(Z))$ is the quantile of order δ of the distribution $\pi(Z)$.

The previously described bounds can be used in the case the collected logs Z_N are i.i.d. sampled. This is the case when we consider a non contextual sampling policy $\mathfrak{U}_0$. In the case this assumption is not verified, one might resort to bounds which does not rely on the i.i.d. hypothesis, like the one provided by McDiarmid inequalities [75]. In this case all the aforementioned properties and the confidence level would hold as is.

4.3 Experimental Results

In this section, we compare the empirical performance of RADT with the ones of a number of algorithms. We evaluate three different versions of RADT, each one using a different statistical approach: H-RADT using the Hoeffding's upper confidence bound $\varepsilon_H(Z,\delta)$; BE-RADT using the Bernstein's upper confidence bound $\varepsilon_{BE}(Z,\delta)$; BA-RADT using the Bayesian upper confidence bound $\varepsilon_{BA}(Z,\delta)$, as defined in Section 4.2. For all these versions of RADT we use a confidence of $\delta_0 = 10^{-2}$. Furthermore, we evaluate a set of off-line regression techniques to learn the reward function using samples $(x_i, a_i) \Rightarrow r_i$ and providing as policy for a context the action providing the highest estimated reward. In the specific, we considered FeedForward regression Neural Network (FFNN) [63] with a hidden layer with 10 neurons, Gaussian kernel Support Vector Machine (SVM) for regression, and Random Forest Regressor (RFR) [76] with 100 trees. We use these algorithms with the default parameters except for those mentioned above. Finally, we evaluate also the POEM algorithm [64] with values $c \in \{1.0, 0.1, 0.01, 0.001\}$ for parameter $\lambda := c\lambda_0$ similarly to what has been presented in the POEM empirical analysis, where λ_0 is the theoretically derived value for the λ parameter (see [64] for more details).[5] We evaluate, for each algorithm, the boxplot of the expected profit in terms of minimum, 1-st quartile, median, 3-rd quartile, maximum, and potential outliers.

4.3.1 Synthetic dataset

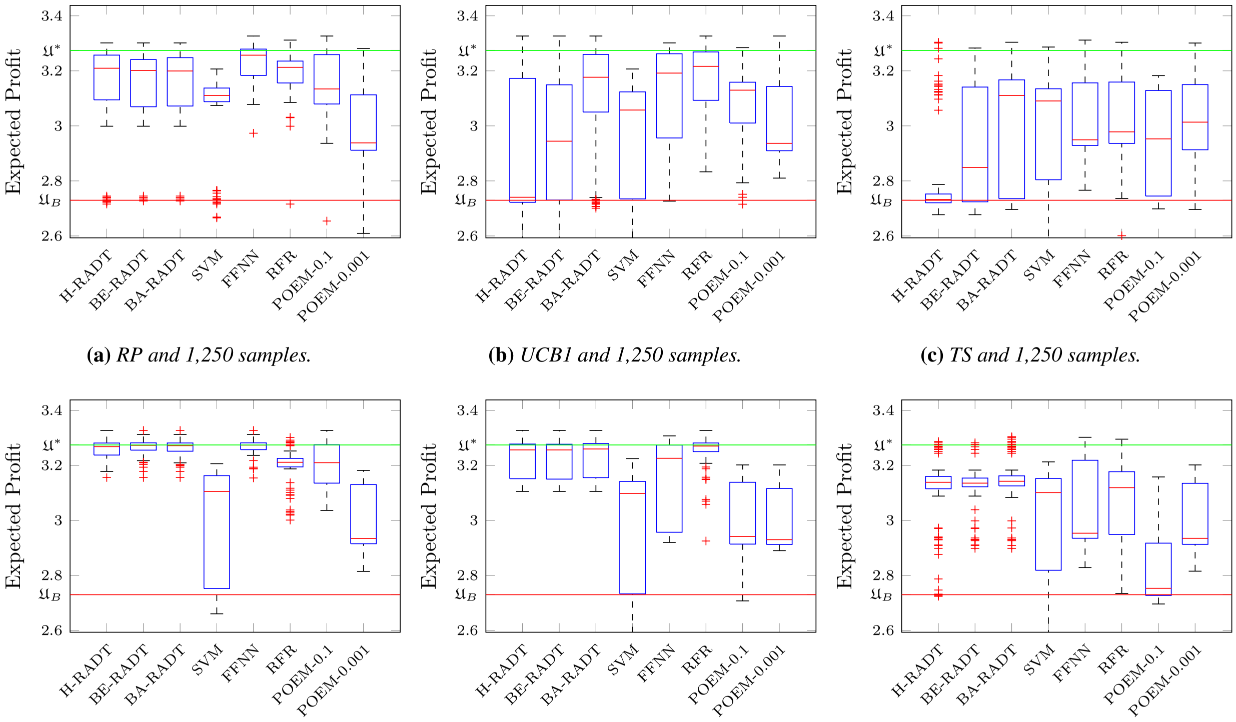
We design synthetic data in which a single product is proposed to the customer that can decide whether to buy it or not. The set of the choices corresponds to the set of prices of the product and we set it as $A = \{1, \dots, 10\}$. A deterministic customer is modeled as a threshold representing her reservation value: any price smaller than or equal to the reservation value will be accepted by the customer (i.e., the user buys the product), while any price strictly larger than the reservation value will be rejected (i.e., the customer does not buy the product). A class of customers is modeled as a probability distribution over the threshold. For simplicity, each class is modeled as a normal distribution with parameters μ and σ. Each class of customers is associated with a set of features modeled as a set of binary variables observable by the algorithms. Basically, each class of customers corresponds to a context the algorithms should be able to identify. All the classes have the same probability of being selected. We assume that a number of data have been collected without any partitioning in contexts by means of three different logging policies, describing which price is returned to the customer. Specifically, we consider: the Random logging Policy (RP), according which each choice of A has the same probability to be chosen (independently of the class of customers), the UCB1 [6] policy, in which the normalized (in $[0, 1]$) profit is considered as reward (and therefore the resulting logging policy is non-stationary), and the policy given by Thompson Sampling (TS) [9] with the same previous definition for the reward (also in this case the logging policy is non-stationary; furthermore Thompson Sampling usually converges to a single option faster than UCB1 and therefore we expect that TS is less explorative than UCB1 that, in its turn, is less explorative than RP). Finally,

[5] Since for $c \in \{1.0, 0.01\}$ POEM is always outperformed by the other algorithms, below we omit the results related to these parametrizations.

we consider four different number of samples for each class of customers in the set $\{1250, 2500, 5000, 10000\}$ to evaluate how the performance of the algorithms varies as the size of the available data varies. For each scenario, characterized by a set of classes of customers, logging policy and number of samples, we randomly generate 100 different Z_N log datasets, and we apply the algorithms to each one. Finally, once obtained the contextual policy returned by the algorithms, we evaluate their performance on a test set Z_M containing $M = 10,000$ samples.

Case with 2 *customer classes.* We consider a case with 2 classes of customers. The first with $\mu = 3$ and the second with $\mu = 8$; $\sigma = 2$ for both. There is only one binary feature observable by the algorithms. Here, we report the most significant results (with $1,250$ and $10,000$ samples per context) in Figure 4.1, provided that experiments on other configurations do not change the final conclusions. The green line reports the average profit of the optimal (clairvoyance) policy $\mathfrak{U}^*$, while the red line reports the average profit of the baseline optimal non-contextual policy $\mathfrak{U}_B$. Although the three RADT algorithms present similar performance, the BA-RADT provides a slightly better performance especially when TS is used and the samples are few. This is reasonable since BA-RADT requires stronger assumptions w.r.t. the other RADT algorithms. Furthermore, as expected, the best performance is obtained with RP, while the worst with TS. Increasing the number of samples allows the algorithms to learn better, but using non-stationary policies may avoid the algorithm to maximize the profit. POEM (for every value of c) is outperformed by RADT both with stationary logging policies (required by the assumption of POEM) and with non-stationary policies (with which an effective functioning of PEOM is not guaranteed). Finally, surprisingly, RFR and FFNN provide good performance usually comparable with that one provided by RADT. However, these methods do not allow a simple interpretation of the policy (e.g., the description of the contexts).

Case with 8 *user classes.* We consider a case with 8 classes of customers. The μs of the normal distributions are $\{1.0, 2.5, 3.5, 4.5, 5.5, 6.5, 8.5, 10.5\}$, while $\sigma = 0.5$ for every class. There are three binary features observable by the algorithms. We report the most significant results (with $1,250$ and $10,000$ samples per context) in Figure 4.2. The red and green lines have the same meaning as in the previous experiment. The most significant result is that the relative performance of POEM w.r.t. the other algorithms degrades importantly. In particular, POEM provides the worst performance for all the pairs logging policy/number of samples. Instead, all the other algorithms perform similarly.



(a) *RP and 1,250 samples.*

(b) *UCB1 and 1,250 samples.*

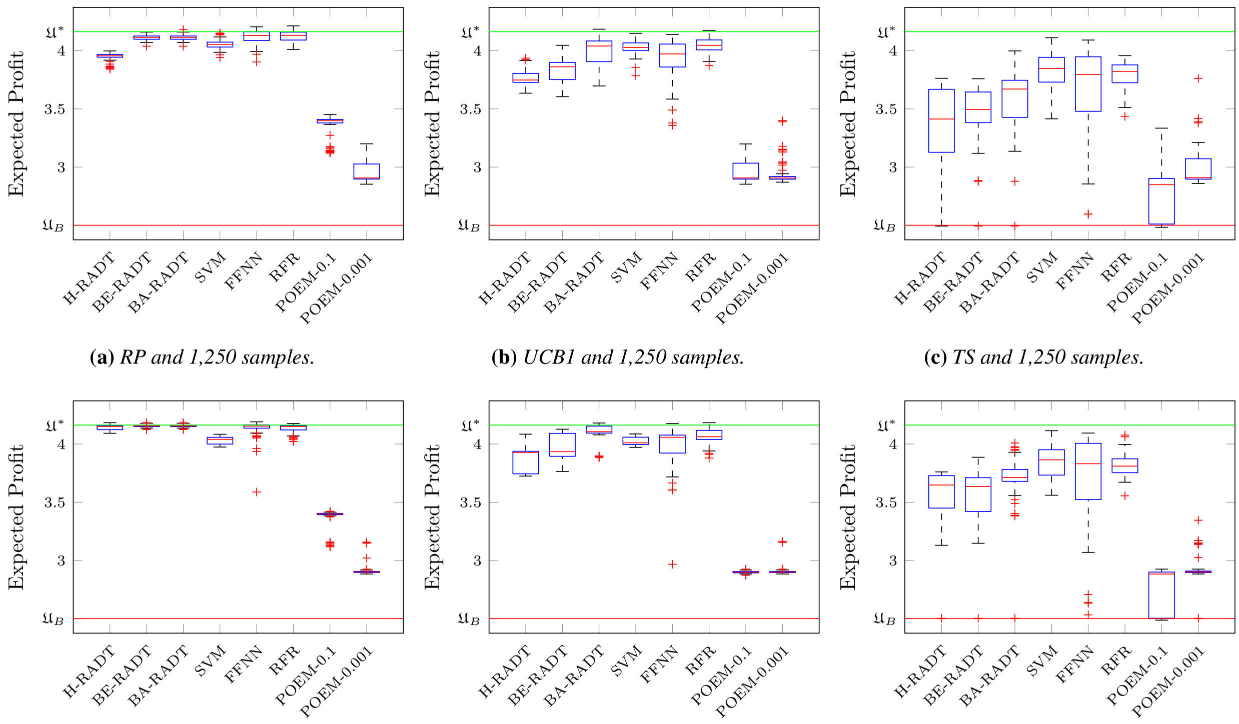
(c) *TS and 1,250 samples.*

(d) *RP and 10,000 samples.*

(e) *UCB1 and 10,000 samples.*

(f) *TS and 10,000 samples.*

Figure 4.1: *Results in the scenarios with 2 contexts.*



(a) *RP and 1,250 samples.*

(b) *UCB1 and 1,250 samples.*

(c) *TS and 1,250 samples.*

(d) *RP and 10,000 samples.*

(e) *UCB1 and 10,000 samples.*

(f) *TS and 10,000 samples.*

Figure 4.2: *Results in the scenarios with 8 contexts.*

4.3.2 Real-world dataset

We use real-world logs generated by an online flight ticket seller (whose name cannot be disclosed, data being confidential). This scenario is characterized by a large number of searches per day and by extremely low conversion rates. The space of choices A is the set of positive monetary markups to apply to a ticket with a discretization of 0.5 Euros. The observable features are, e.g., the departure airport/country/continent, the arrival airport/country/-continent, the number of passengers, and the fact that the flight is one-way or round-trip. We use three different datasets with 3, 5, and 9 millions of samples respectively. Each dataset contains information about: the features associated with the search done by user, the used markup, and the user feedback in terms of purchase or not of the ticket. We apply BA-RADT algorithm with $\delta_0 = 0.1$ to these datasets obtaining three different trees. In this case, we evaluate the percentage of improvement of the policy induced by RADT $\mathfrak{U}$ over the one provided by the non-contextual one $\mathfrak{U}_B$ in terms of lower bound, i.e., $I_{\%}(\mathfrak{U}) := 100 \frac{\underline{P}(\mathfrak{U}, Z_N, \delta_0) - \underline{P}(\mathfrak{U}_B, Z_N, \delta_0)}{\underline{P}(\mathfrak{U}_B, Z_N, \delta_0)}$.

Results. With the first dataset, the algorithm identifies 2 contexts (corresponding to the terminal nodes of the tree) with a minimum assured improvement $I_{\%}(\mathfrak{U}) = 3\%$. With the second dataset, the algorithm identifies 8 contexts with a minimum assured improvement $I_{\%}(\mathfrak{U}) = 11\%$. With the third dataset, the algorithm identifies 27 contexts with a minimum assured improvement $I_{\%}(\mathfrak{U}) = 38\%$.

Figure 4.3 reports the tree generated by RADT with the second real-world dataset. In each internal node (blue circles) it is specified the variable j^* w.r.t. the algorithm performs the split and on each edge is reported the condition a sample has to satisfy to fall in the node below. The leaves (green rectangles), which correspond to contexts X_i, report information about the optimal action a^* chosen by RADT in the context X_i, the lower bound $\underline{\mu}(a^*)$ over the expected reward of action a^* in the context X_i, the lower bound $\underline{p}$ over the probability of a sample to belong to the context X_i and the cardinality $|Z|$ of the data available in the considered context. It is possible to see how the model identifies contexts which are easy to interpret, for instance, the leftmost context identifies all the flight tickets which are arriving in Czech Republic for a single passenger. The high interpretability of the context trees allowed some experts of the field to analyze the partitioning in contexts. Their evaluation has been extremely positive, suggesting new business strategies unexplored so far.

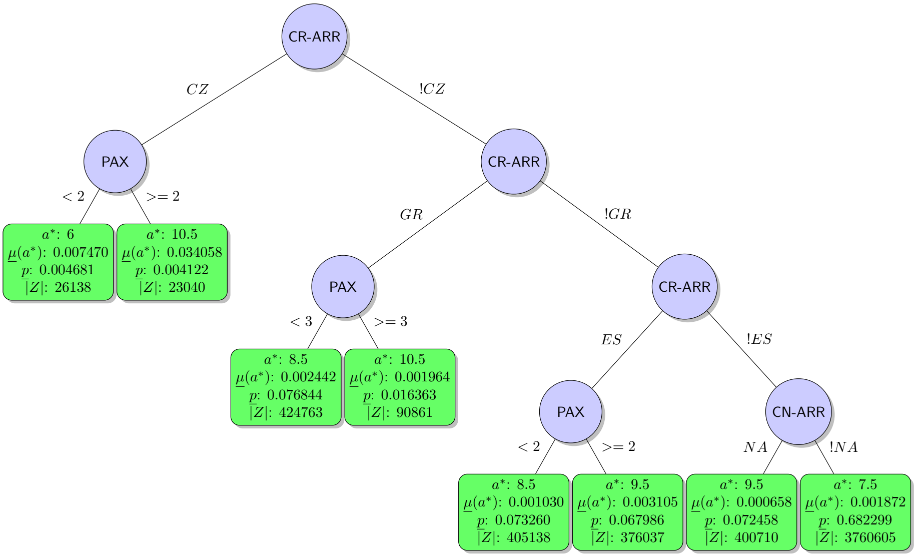

CR-ARR
CZ
$!CZ$
PAX
CR-ARR
< 2
$>= 2$
a^*: 6
$\underline{\mu}(a^*)$: 0.007470
$\underline{p}$: 0.004681
$|Z|$: 26138
a^*: 10.5
$\underline{\mu}(a^*)$: 0.034058
$\underline{p}$: 0.004122
$|Z|$: 23040
GR
$!GR$
PAX
CR-ARR
< 3
$>= 3$
a^*: 8.5
$\underline{\mu}(a^*)$: 0.002442
$\underline{p}$: 0.076844
$|Z|$: 424763
a^*: 10.5
$\underline{\mu}(a^*)$: 0.001964
$\underline{p}$: 0.016363
$|Z|$: 90861
ES
$!ES$
PAX
CN-ARR
< 2
$>= 2$
NA
$!NA$
a^*: 8.5
$\underline{\mu}(a^*)$: 0.001030
$\underline{p}$: 0.073260
$|Z|$: 405138
a^*: 9.5
$\underline{\mu}(a^*)$: 0.003105
$\underline{p}$: 0.067986
$|Z|$: 376037
a^*: 9.5
$\underline{\mu}(a^*)$: 0.000658
$\underline{p}$: 0.072458
$|Z|$: 400710
a^*: 7.5
$\underline{\mu}(a^*)$: 0.001872
$\underline{p}$: 0.682299
$|Z|$: 3760605

Figure 4.3: *Tree generated with the second dataset (5 millions of samples) using BA-RADT, providing a minimum assured improvement $I_{\%}(\mathfrak{U}) = 11\%$.*

CHAPTER 5

Multi-Armed Bandit for Pricing: Frequentist Approach

In this chapter, we study the stochastic MAB setting on a finite number of arms, and we propose techniques to apply to Upper Confidence Bound policies in order to exploit the monotonicity property of conversion rates as well as the *a priori* information on the maximum conversion rate. Section 5.1 provides the formulation for the considered MAB setting. In Section 5.2, we describe the proposed techniques in stationary settings and we prove that the asymptotic regret bounds of our algorithms are of the same order as the state-of-the-art ones. In Section 5.3, we describe the proposed techniques in non-stationary settings, providing regret bound analysis. Section 5.5 provides experimental results in stationary settings, while Section 5.6 provides experimental results in non-stationary settings. Basically, the empirical analysis shows that our algorithms provide significant advantages with respect to general-purpose MAB algorithms in the early stages of the learning process. This is crucial in real pricing scenarios, where very low conversion rates (that require a long exploration phase to have accurate estimations) and non-stationary buyers' demands make the algorithms to work in a never-ending transient. Finally, in Section 5.7 we introduce Thompson Sampling, a Bayesian MAB algorithm, and we compare it to the frequentist methods proposed in the previous sections. The proofs of the theorems are reported in Appendix A and the pseudocode of some algorithms can be found in Appendix B.

5.1 Problem Formulation

We study a scenario where an unlimited non-perishable amount of goods is available to a monopolistic seller, who proposes the product she is selling to some unknown buyers at a chosen price. We model our problem as a MAB problem [6], where at each round $t \in \{1, \dots, N\}$ over a finite horizon N the seller selects an arm, corresponding to a price, among a strictly ordered finite set of K different arms $A = \{a_1, \dots, a_K\}$ with $a_i \in (0, +\infty)$.[1] As customary in microeconomics, each buyer is modeled as a deterministic agent who buys the item only if the proposed price is lower than or equal to a threshold $s \in \mathbb{R}^+$. Thus, all the prices that are lower than s lead to a sale, while all the prices higher than s lead to a non-sale. Since buyers have generally different thresholds s, we model s as realizations of a random variable S with a probability density function (pdf) $\mathcal{S}$ over a finite support $\Omega \subset \mathbb{R}^+$. In stationary settings, the pdf $\mathcal{S}$ is unique for all the rounds, whereas in non-stationary settings each round t presents a potentially different pdf $\mathcal{S}_t$. We assume that the pdfs are unknown to the seller and therefore that the seller needs to estimate them. Furthermore, for the sake of presentation, we assume the costs of the seller to be zero.[2] In that case, the price a_i also represents the reward received by the seller once she sold the product. The seller's goal is the maximization of the total expected revenue over the time horizon N. A MAB *policy* is an algorithm $\mathfrak{U}(h_t)$ that chooses the next arm a_{i_t} to play at round t given history h_t, defined as the sequence of past plays and obtained rewards. At each round t the algorithm observes a single realization of the reward V_{i_t} obtained from the arm $a_{i_t} = \mathfrak{U}(h_t)$.

5.1.1 Stationary Pricing Model

In the case of stationary settings, the reward gained by pulling an arm a_i is a bounded random variable $V_i = a_i X_i$, where $X_i \sim Be(\mu_i)$ is a Bernoulli variable that represents the *outcome* (buy/not buy) of the transaction, where $\mu_i := \mathbb{E}[X_i]$ is the expected value of the outcome corresponding to arm a_i, i.e., the conversion rate. We denote with $V_{i,n}$ and $X_{i,n}$ the random variable of the reward and the outcome of the n-th pull of the i-th arm, respec-

[1] From now on, we will use the terms arm and price interchangeably.

[2] Our work can be easily applied also to the case in which the costs of the seller are arbitrary and known.

tively, and with $v_{i,n}$ and $x_{i,n}$ their realizations. We denote with $T_i(t) = \sum_{m=1}^{t} \mathbb{1}\{\mathfrak{U}(h_m) = a_i\}$ the number of times the arm a_i was pulled in the first t rounds, where $\mathbb{1}\{B\}$ is the indicator function of the event B. The objective of a policy is the maximization of the expected cumulative reward or, equivalently, the minimization of the loss with respect to the optimal decision (in terms of reward). This loss is usually addressed as *(cumulative) pseudo-regret*, whose definition over the time horizon N is:

$$\bar{R}_N = a_{i^*}\mu_{i^*}N - \sum_{i=1}^{K} a_i \mu_i \mathbb{E}[T_i(N)],$$

where $i^* = \arg\max_{i\in\{1,\dots,K\}} a_i\mu_i$ is the optimal arm and $\mathbb{E}[\cdot]$ is the expectation with respect to the stochastic components of the policy.

5.1.2 Non-Stationary Pricing Model

In the case of non-stationary settings, we analyse an *abruptly changing environment*, similarly to what has been studied in [77], where the pdf $\mathcal{S}_j$ describing the buyer behavior is constant during sequences of rounds called *phases* and changes at unknown rounds called *breakpoints*. Thus, differently from the stationary scenario, the expected value of the outcome $\mu_{i,t}$ of an arm a_i at round t changes over the phases and therefore the best arm $a_{i^*,t}$ might change after each breakpoint.

A breakpoint $b \in \{1, \dots, N\}$ is a round such that $\exists i \mid \mu_{i,b-1} \neq \mu_{i,b}$, i.e., a round b where the expected reward of at least one arm changed with respect to the one at round $b-1$. In a non-stationary environment $\mathcal{S}^{(B)}$ with time horizon N we have a set of breakpoints $B = \{b_1, \dots, b_{\Upsilon_N}\}$ of cardinality Υ_N (for sake of notation we define $b_1 = 1$), which determines a set of phases $\{\Phi_\phi\}_{\phi=1}^{\Upsilon_N}$, where $\Phi_\phi = \{t | b_{\phi-1} \leq t < b_\phi\}$, i.e., the set of rounds between two consecutive breakpoints. During phase Φ_ϕ, we denote (with abuse of notation) with $\mu_{i,\phi}$ the expected value of the outcome of the i-th arm a_i and with $\mu_{i^*,\phi}$ the expected conversion probability corresponding to the best arm $a_{i^*,\phi}$. By defining $N_\phi = |\Phi_\phi|$, the cumulative pseudo-regret of a generic policy over a non-stationary environment is:

$$\begin{aligned}
\bar{R}_N &= \mathbb{E}\left[\sum_{t=1}^{N} (a_{i^*,t}\mu_{i^*,t} - a_{i_t}\mu_{i_t,t})\right] \\
&= \sum_{\phi=1}^{\Upsilon_N} a_{i^*,\phi}\mu_{i^*,\phi}N_\phi - \mathbb{E}\left[\textstyle\sum_{t=1}^{N} a_{i_t}\mu_{i_t,t}\right] \\
&= \sum_{\phi=1}^{\Upsilon_N} \left(a_{i^*,\phi}\mu_{i^*,\phi}N_\phi - \mathbb{E}\left[\textstyle\sum_{t\in\Phi_\phi} a_{i_t}\mu_{i_t,t}\right]\right) \\
&= \sum_{\phi=1}^{\Upsilon_N} \left(a_{i^*,\phi}\mu_{i^*,\phi}N_\phi - \textstyle\sum_{i=1}^{K} a_i\mu_{i,\phi}\mathbb{E}[T_i(\Phi_\phi)]\right),
\end{aligned}$$

where $\sum_{\phi=1}^{\Upsilon_N} N_\phi = N$, $T_i(\Phi_\phi) = \sum_{m\in\Phi_\phi} \mathbb{1}\left\{\mathfrak{U}(h_m) = a_i\right\}$ is the number of times the i-th arm a_i has been pulled during phase Φ_ϕ and $\mathbb{E}[\cdot]$ is the expectation with respect to the stochastic components of the policy.

5.1.3 Properties of the Pricing Problem

We exploit two properties of the probability distributions of the random variables $X_{i,n}$ representing the outcomes of the transactions. The first property is the *dependency* between arms. While in the classic MAB setting the rewards produced by different arms are assumed to be drawn from independent probability distributions, in our setting this does not hold anymore, since the realizations at time t (i.e., $x_{1,T_1(t)}, \dots, x_{K,T_K(t)}$) of the outcome variables $X_{1,T_1(t)}, \dots, X_{K,T_K(t)}$ are correlated by the threshold of the buyer that plays at round t. The expected *conversion probability* $\mu_{i,\phi}$ corresponding to price a_i at phase Φ_ϕ is defined as the probability that a user purchases the product or formally:

$$\mu_{i,\phi} := \mathbb{P}_{S_\phi}(s \geq a_i) = 1 - \int_0^{a_i} \mathcal{S}_\phi(x)dx.$$

Notice that, in stationary settings, we have a single probability distribution, thus $\mathcal{S}_\phi = \mathcal{S}$ and $\mu_{i,\phi} = \mu_i$. From the non-negativity of the probability distribution function $\mathcal{S}_\phi$ and from the properties of the integral, it clearly follows that $a_i < a_j \Rightarrow \mu_{i,\phi} > \mu_{j,\phi}$, i.e., the expected conversion probability is *monotonically* decreasing with respect to the price.

The second property concerns the *low conversion rates*, which are common in many e-commerce applications. In this case, the seller knows that only a certain percentage of the buyers $\mu_{\max} \in [0,1]$ (typically $\mu_{\max} \ll 1$) really considers the possibility of purchasing the good, while the remaining part $1 - \mu_{\max}$ would not buy at any price. Such behavior can be introduced in the user model by considering $\mathcal{S}_\phi$ with pdf equal to $\mathcal{S}_\phi(x) =$

$(1 - \mu_{\max}) \cdot \delta(0) + \mu_{\max} \cdot \mathcal{C}_\phi(x), x \in \Omega$, where $\delta(0)$ is a Dirac delta probability distribution centered in 0 and $\mathcal{C}_\phi(\cdot)$ is a pdf defined over Ω.

5.2 Exploiting Pricing Properties in Stationary Settings

In this section, we describe techniques exploiting the pricing problem structure in stationary settings. We use the monotonicity structure of the expected value of the outcome $\{\mu_i\}_{i=1}^K$ of the arms to tighten the UCBs used in the frequentist approach. The proposed techniques are then applied to UCB1 [6] and UCBV [36], as interesting case studies. Furthermore, to exploit the prior knowledge about low conversion rates, we propose the use of a form of the Chernoff's bound [7] which, in this case, is tighter than the Hoeffding's one. Finally, we provide an algorithm that combines both techniques.

5.2.1 Exploiting the Monotonicity Property

Given an arm a_i, the realizations of all the outcomes X_j with $j < i$ provide information that can be exploited for the computation of the UCB on the expected value μ_i. Indeed, since $\mu_i \leq \mu_j$, we can use the realizations drawn so far from X_j as optimistic samples to estimate μ_i. In what follows, we will derive a set of bounds which exploit the samples coming from arms with lower value and consider the tightest among them to design an algorithm for the pricing scenario. Let $\bar{X}_{i,t}$ be the empirical mean, at round t, of the outcomes obtained by pulling arm a_i for $T_i(t-1)$ rounds (i.e., an estimator of the expected conversion rate μ_i of arm a_i) and $\bar{x}_{i,t}$ its realization, or formally:

$$\bar{X}_{i,t} := \frac{1}{T_i(t-1)} \sum_{n=1}^{T_i(t-1)} X_{i,n},$$

$$\bar{x}_{i,t} := \frac{1}{T_i(t-1)} \sum_{n=1}^{T_i(t-1)} x_{i,n}.$$

Similarly, given $1 \leq j \leq i$, let $\bar{X}_{ji,t}$ be the following convex combination of the sample means $\bar{X}_{j,t}, \dots, \bar{X}_{i,t}$ and let $\bar{x}_{ji,t}$ be its realization:

$$\bar{X}_{ji,t} := \frac{\sum_{k=j}^{i} T_k(t-1) \bar{X}_{k,T_k(t-1)}}{T_{ji}(t-1)},$$

$$\bar{x}_{ji,t} := \frac{\sum_{k=j}^{i} T_k(t-1) \bar{x}_{k,t}}{T_{ji}(t-1)},$$

where $T_{ji}(t-1) = \sum_{k=j}^{i} T_k(t-1)$, corresponding to the cumulative number of rounds all the arms from j to i have been pulled. Since, given the monotonicity property, it holds:

$$\mu_{ji,t} = \mathbb{E}\left[\bar{X}_{ji,t}\right] \geq \mu_i,$$

any upper bound on $\mu_{ji,t}$ is also an upper bound on μ_i. This allows us to bound the expected value μ_i of the outcome X_i of arm a_i by using samples drawn from the set of outcomes $\{X_1, \dots, X_i\}$. In other words, we can compute an upper bound on the expected conversion rate associated with arm a_i by taking into account also the experience collected when lower arms were selected. By considering concentration bounds over the aggregated variables X_{ji} with $j \in \{1, \dots, i\}$, we may possibly find a tighter bound also on the expected value of the outcome X_i. In what follows, we apply this idea to the concentration bounds used in UBC1 and UCBV policies.

UCB1 with Monotonic Arms (UCB1-M)

Applying the Hoeffding's inequality [73] to the random variables $\bar{X}_{ji,t}$, with probability at least $1-\frac{p}{i}$ where $p \in [0,1]$, we have the following UCBs (from now on denoted as UCB1-M):

$$u_{ji,t}^{\text{(UCB1-M)}} = \bar{x}_{ij,t} + \sqrt{\frac{\log(i) - \log(p)}{2T_{ji}(t-1)}} > \mu_{ji,t} \geq \mu_i \quad \forall j \in \{1, \dots, i\}. \quad (5.1)$$

Since, for each $j \in \{1, \dots, i\}$, $u_{ji,t}^{\text{(UCB1-M)}}$ is a valid upper bound on μ_i holding with at least probability $1 - \frac{p}{i}$, by setting $u_{i,t}^{\text{(UCB1-M)}} = \min_{j \in \{1,\dots,i\}} u_{ji,t}^{\text{(UCB1-M)}}$ and resorting to a union bound, we have the tightest bound among those provided by Equation (5.1), holding with at least probability $1-p$.

The use of the UCB1-M bound constitutes a potential improvement over the traditional one used by the UCB1 algorithm and obtained by considering realizations coming from a single arm. Indeed, this novel UCB exploits $T_{ji}(t-1) \geq T_i(t-1)$ samples and may be tighter than the UCB1 one. If the observed empirical means are consistent with the monotonicity property (i.e., $\bar{x}_{i,t} < \bar{x}_{j,t}, \ \forall i > j$) the use of a larger number of samples coming from other arms may allow one (specially in the early stages) to tighten the bound. The proposed method is even more advantageous when empirical means are not consistent with the monotonicity property (i.e., $\exists \ i > j$ such that $\bar{x}_{i,t} > \bar{x}_{j,t}$). In this case, the bound $u_{i,t}^{\text{(UCB1-M)}}$ is significantly improved over the original UCB1 bound. Such a situation is exemplified in Figure 5.1, where we have that, in contrast with the monotonicity over $A = \{a_1, a_2\}$,

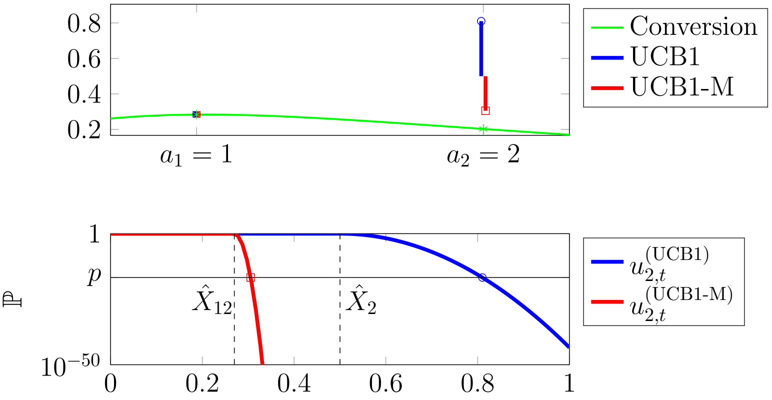


Figure 5.1: *Example of empirical means not consistent with the monotonicity property and UCBs corresponding to UCB1 and UCB1-M. The top figure presents the real conversion rate function (green line) and two bars going from the estimated expected reward and the two bounds (in blue and red). The bottom figure represents the dependence of the two bounds over arm a_2 (in blue and red) with respect to the confidence level one wants to keep $[110^{-50}]$; p is the confidence level used to draw the top figure and the dashed lines are the empirical means of X_2 and X_{12}.*

the empirical mean of the outcome corresponding to arm $a_1 = 1$, i.e., $\bar{x}_{1,t}$, is lower than the one of arm $a_2 = 2$, i.e., $\bar{x}_{2,t}$. This happens because arm a_2 has been selected much less often than arm a_1 and so its empirical mean is more uncertain. The samples drawn from arm a_1 allow to tighten the UCB for arm a_2 from the value denoted by the blue circle to the value denoted by the red square (top). The use of the proposed UCB for arm a_2 does not imply a reduction in the confidence level, since the two values have been obtained from different bounds. Indeed, they share the same confidence level $1 - p$, as shown in Figure 5.1 (bottom).

The algorithm corresponding to the derived UCB, namely *UCB1 with Monotonic arms* (UCB1-M), is presented in Algorithm 5. At first, the algorithm selects each arm once, to have at least one outcome realization coming from each arm. Subsequently, for each round t, it assigns for each arm a_i:

$$u_{i,t}^{(\text{UCB1-M})} = \min_{j \in \{1,\ldots,i\}} \left\{ \bar{x}_{ji,t} + \sqrt{\frac{4\log(t) + \log(i)}{2T_{ji}(t-1)}} \right\},$$

where, we considered $p = t^{-4}$ and we selected the $j \in \{1, \ldots, i\}$ minimizing $u_{ji,t}^{(\text{UCB1-M})}$. Finally, the algorithm selects for the next round t the arm a_{i_t} providing the maximum upper bound $a_{i_t} u_{i_t,t}^{(\text{UCB1-M})}$ over the expected reward $a_i \mu_i$.

Algorithm 5: UCB1-M

```
Initialization
for $t \in \{1, \dots, K\}$ do
    Play arm $a_t$ and observe $x_{t,1}$
Loop
for $t \in \{K+1, \dots, N\}$ do
    for $i \in \{1, \dots, K\}$ do
        Compute:
            $u_{i,t}^{(\text{UCB1-M})} = \min_{j \in \{1,\dots,i\}} \left\{ \bar{x}_{ji,t} + \sqrt{\frac{4\log(t) + \log(i)}{2T_{ji}(t-1)}} \right\}$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} a_i u_{i,t}^{(\text{UCB1-M})}$ and observe $x_{i_t, T_{i_t}(t)}$
```

By using the UCB1-M algorithm we are able to show that:

Theorem 1. *If policy UCB1-M is run over a stationary MAB setting with a monotonic set A, the pseudo-regret after N rounds is at most:*

$$\bar{R}_N \leq \sum_{i | a_i \neq a_{i^*}} \frac{8a_i^2 \log(N)}{\Delta_i} + \sum_{i | a_i \neq a_{i^*}} \frac{2a_i^2 \log(K)}{\Delta_i} + \left(1 + \frac{\pi^2}{3}\right) \sum_{i=1}^{K} \Delta_i,$$

where $\Delta_i := a_{i^*}\mu_{i^*} - a_i \mu_i, \forall i \in \{1, \dots, K\}$.

The previous theorem guarantees that the proposed algorithm has, in the worst case, $O(\log(N))$ regret, as the UCB1 policy. Nevertheless, we show that, empirically, UCB1-M dramatically outperforms UCB1.

UCBV with Monotonic Arms (UCBV-M)

Similarly, by resorting to the bound presented Theorem 1 in [36], it is possible to derive an UCB that also considers the empirical variance $\bar{V}_{ji,t}$ of the variable $X_{ji,t}$ by using its realization $\bar{v}_{ji,t}$, formally defined as, respectively:

$$\bar{V}_{ji,t} = \frac{\sum_{k=j}^{i} \sum_{n=1}^{T_k(t-1)} \left(X_{k,n} - \bar{X}_{ji,t}\right)^2}{T_{ji}(t-1)},$$

$$\bar{v}_{ji,t} = \frac{\sum_{k=j}^{i} \sum_{n=1}^{T_k(t-1)} \left(x_{k,n} - \bar{x}_{ji,t}\right)^2}{T_{ji}(t-1)}.$$

The bound, from now on denoted as UCBV-M, holding with probability at least $1 - 3\left(\frac{p}{i}\right)^{\xi}$, with $p \in [0,1]$ is:

$$\begin{aligned} u_{ji,t}^{(\text{UCBV-M})} =& \bar{x}_{ji,t} + \sqrt{\frac{2\bar{v}_{ji,t}\xi[\log(i) - \log(p)]}{T_{ji}(t-1)}} + \\ &+ \frac{3c\xi[\log(i) - \log(p)]}{T_{ji}(t-1)} > \mu_{ji,t}, \end{aligned}$$

where $\xi, c \in \mathbb{R}, \xi > 1, c \geq 1$; see [36] for details. Note that, if we choose $\xi > 1 - \frac{\log(3)}{\log(p)}$, the previous bound holds with probability at least $1 - \frac{p}{i}$, i.e., with the same confidence the UCB1-M holds.

The algorithm, based on the derived bound and called *UCBV with Monotonic arms* (UCBV-M), is described in Algorithm 6. Similarly to UCB1-M, it chooses each arm once in the initial phase and, after that, it selects the next arm to play on the basis of the upper confidence bounds $u_{i,t}^{(\text{UCBV-M})} = u_{\bar{j}i,t}^{(\text{UCBV-M})}$, where $\bar{j}$ is chosen to minimize $u_{i,t}^{(\text{UCBV-M})}$ and $p = t^{-1}$. It is possible to show that:

Theorem 2. *If policy UCBV-M is run with $\xi = 1.2$ and $c = 1$ over a setting with a monotonic set A, the pseudo-regret after N rounds is at most:*

$$\begin{aligned} \bar{R}_N \leq& \frac{12}{5} \sum_{i | a_i \neq a_{i^*}} a_i^2 \left(\frac{\sigma_i^2}{\Delta_i} + \frac{32}{15} \right) \log(N) + \\ &+ \sum_{i | a_i \neq a_{i^*}} \Delta_i \left[1 + a_i^2 \left(\frac{\sigma_i^2}{\Delta_i^2} + \frac{2}{\Delta_i} \right) \log(K) \right], \end{aligned}$$

where $\sigma_i^2 := Var(X_{i,n})$, $\forall i \in \{1, \dots, K\}, \forall\, n \in \{1, \dots, T_i(N)\}$.

Even in this theorem the asymptotic behaviour is of order of $O(\log(N))$ as the one presented in [36] for the UCBV algorithm.

5.2.2 Exploiting the Low Conversion Rates Property

When it is *a priori* known that the conversion rates of all the arms are upper bounded by a value $\mu_{\max} \leq \frac{1}{2}$, i.e., $\mu_i \leq \mu_{\max}$ for every $i \in \{1, \dots, K\}$, it is possible to exploit probabilistic bounds that achieve better results than the one based on the Hoeffding's inequality [73]. More specifically, one of the approximations used in the derivation of the Hoeffding's inequality for the generic outcome X_i is:

$$\mathbb{P}(\bar{X}_{i,t} + \varepsilon \leq \mu_i) \leq e^{-T_i(t-1)D(\mu_i+\varepsilon||\mu_i)} \leq e^{-2T_i(t-1)\varepsilon^2}, \qquad (5.2)$$

Algorithm 6: UCBV-M

```
**Initialization**
**Input:** $\xi, c$
**for** $t \in \{1, \dots, K\}$ **do**
    Play arm $a_t$ and observe $x_{t,1}$
**Loop**
**for** $t \in \{K+1, \dots, N\}$ **do**
    **for** $i \in \{1, \dots, K\}$ **do**
        Compute:
```

$$u_{i,t}^{\text{(UCBV-M)}} = \min_{j\in\{1,\dots,i\}} \left\{ \frac{3c[\zeta \log(t) + \log(i)]}{T_{ji}(t-1)} + \right. \\ \left. + \sqrt{\frac{2\bar{v}_{ji,t}[\zeta \log(t) + \log(i)]}{T_{ji}(t-1)}} + \bar{x}_{ji,t} \right\}$$

```
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i\in\{1,\dots,K\}} a_i u_{ji,t}^{\text{(UCBV-M)}}$ and observe $x_{i_t,T_{i_t}(t)}$
```

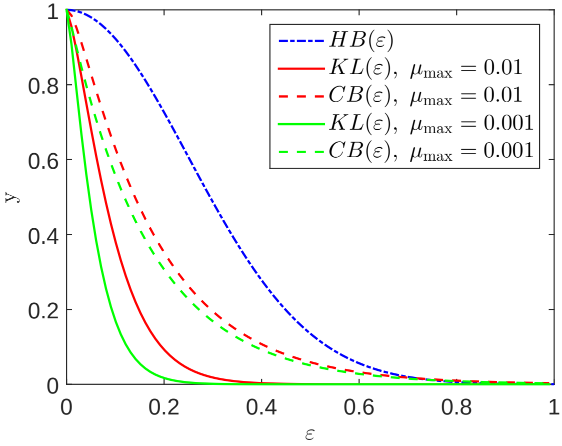


Figure 5.2: *Example of bounds $y = e^{-x(\varepsilon)}$ obtained with different $x(\varepsilon)$: Hoeffding's Bound ($HB(\varepsilon)$), Kullback-Leiber divergence ($KL(\varepsilon)$) and Chernoff's Bound ($CB(\varepsilon)$).*

where $D(\mu_i + \varepsilon||\mu_i)$ is the Kullback-Leibler (KL) divergence between two Bernoulli variables with mean $\mu_i + \varepsilon$ and μ_i, respectively.

As shown in Figure 5.2, the bound based on the KL divergence (solid lines) and the one on Hoeffding's inequality (dash-dotted line) diverge as $\mu_{\max}$ decreases. To reduce the gap, we consider the following result that is one of the formulations of the Chernoff's bound [7]:

Theorem 3 (Theorem 4 in [78], Lower tail)**.** *Given a set of* $T_i(t-1)$ *independent and identically distributed random variables* $\{X_{i,1}, \dots, X_{i,T_i(t-1)}\}$ *such that* $X_{i,s} \sim Be(\mu_i)$*, for any* $\varepsilon > 0$ *we have:*

$$\mathbb{P}(\bar{X}_{i,t} + \varepsilon \leq \mu_i) \leq e^{-\frac{T_i(t-1)\varepsilon^2}{2\mu_i}}.$$

Since μ_i is unknown, the above concentration inequality cannot be used in practice. On the other hand, under the assumption that $\mu_i \leq \mu_{\max}$, we can replace μ_i with $\mu_{\max}$, thus getting an upper confidence bound that is tighter than the Hoeffding's one and gets close to the one obtained by knowing the KL divergence (see dashed lines in Figure 5.2). In order to obtain an upper confidence bound over μ_i with confidence $1-p$, with $p \in [0,1]$, we resort to Theorem 3 and get:

$$\mathbb{P}(\bar{X}_{i,t} + \varepsilon \leq \mu_i) \leq e^{-\frac{T_i(t-1)\varepsilon^2}{2\mu_i}} \leq e^{-\frac{T_i(t-1)\varepsilon^2}{2\mu_{\max}}} = p, \tag{5.3}$$

where the last inequality derives from the trivial fact that $\mu_{\max} \geq \mu_i$ for every $i \in \{1, \dots, K\}$. Thus, with probability at least $1-p$ we have the following UCBs (from now on denoted as UCB-L):

$$u_{i,t}^{(\text{UCB-L})} := \bar{x}_{i,t} + \sqrt{-\frac{2\mu_{\max}\log(p)}{T_i(t-1)}} \geq \mu_i, \tag{5.4}$$

where the square root term is computed by considering the positive root of the second order equality in Equation (5.3). By comparing the two bounds provided by Hoeffding's and Chernoff's inequalities, it is possible to compute a sufficient condition that identifies when the former is tighter than the latter: when $\mu_{\max} > \frac{1}{2}$ the bound in Equation (5.3) is larger than the one in the right hand side of Equation (5.2). As a consequence, if we cannot guarantee low conversion probabilities, it is better to resort to the traditional Hoeffding's bound.

The proposed algorithm, namely *Upper Confidence Bound with Low conversion rates* (UCB-L) is presented in Algorithm 7, where we set $p = t^{-4}$ and we choose the next arm to be pulled by selecting the one having the maximum expected revenue. The execution is analogous to the one already described for UCB1-M and UCBV-M, where we have an initial round robin over all the arms and, after that, the choice of the arm to be played in the next round is based on the upper bound of the regret $a_i u_{i,t}^{(\text{UCB-L})}$.

In this case it is possible to show that:

Theorem 4. *If policy UCB-L is run over a stationary MAB setting with a set of arms* A *in which each arm* $a_i \in A$ *has outcome* $X_{i,t}$ *such that* $\mathbb{E}[X_{i,t}] =$

Algorithm 7: UCB-L

```
**Initialization**
**Input:** $\mu_{\max}$
**for** $t \in \{1, \dots, K\}$ **do**
    Play arm $a_t$ and observe $x_{t,1}$
**Loop**
**for** $t \in \{K+1, \dots, N\}$ **do**
    **for** $i \in \{1, \dots, K\}$ **do**
        Compute:
            $u_{i,t}^{\text{(UCB-L)}} = \bar{x}_{i,t} + \sqrt{\frac{8\mu_{\max}\log(t)}{T_i(t-1)}}$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} a_i u_{i,t}^{\text{(UCB-L)}}$ and observe $x_{i_t,T_{i_t}(t)}$
```

$\mu_i \leq \mu_{\max} \leq \frac{1}{2}$ for each $t \in \{1, \dots, N\}$, the pseudo-regret after N rounds is at most:

$$\bar{R}_N \leq \sum_{i | a_i \neq a_{i*}} \frac{32\mu_{\max} a_i^2 \log(N)}{\Delta_i} + \left[1 + \frac{\pi^2}{6} + \zeta\left(\frac{10}{7}\right)\right] \sum_{i=1}^{K} \Delta_i,$$

where $\zeta(\cdot)$ is the Riemann zeta function.

Even by resorting by this newly designed bound the asymptotic order is $O(\log(N))$, thus we are assured to lose only a logarithmic amount of reward in the learning process.

5.2.3 Exploiting both Properties

Here, we show how to combine both the monotonic and the low conversion rates properties into a single algorithm. Such algorithm, named UCB-LM, simply consists of computing for each arm a_i the minimum upper confidence bound among the ones built using $\bar{X}_{ji,t}$, with $j \in \{1, \dots, i\}$, but, differently from UCB1-M, the UCBs are built exploiting the Chernoff's inequality and the assumption over the maximum conversion rate as it happens in UCB-L.

The resulting algorithm is summarized in Algorithm 8. Also in this case, we can state the following result:

Theorem 5. *If policy UCB-LM is run over a stationary MAB setting with a monotonic set A in which each arm $a_i \in A$ has outcome $X_{i,t}$ such that $\mathbb{E}[X_{i,t}] = \mu_i \leq \mu_{\max} \leq \frac{1}{2}$ for each t, the pseudo-regret after N rounds is at*

Algorithm 8: UCB-LM

```
**Initialization**
**Input:** $\mu_{\max}$
**for** $t \in \{1, \dots, K\}$ **do**
    Play arm $a_t$ and observe $x_{t,1}$
**Loop**
**for** $t \in \{K+1, \dots, N\}$ **do**
    **for** $i \in \{1, \dots, K\}$ **do**
        Compute:
        $$u_{i,t}^{(\text{UCB-LM})} = \min_{j \in \{1,\dots,i\}} \left\{ \bar{x}_{ji,t} + \sqrt{\frac{2\mu_{\max}[4\log(t) + \log(i)]}{T_{ji}(t-1)}} \right\}$$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} a_i u_{i,t}^{(\text{UCB-LM})}$ and observe $x_{i_t, T_{i_t}(t)}$
```

most:

$$\bar{R}_N \leq \sum_{i | a_i \neq a_{i^*}} \frac{32\mu_{\max} a_i^2 \log(N)}{\Delta_i} + \sum_{i | a_i \neq a_{i^*}} \frac{8\mu_{\max} a_i^2 \log(K)}{\Delta_i} + \\ + \left[1 + \frac{\pi^2}{6} + \zeta\left(\frac{10}{7}\right)\right] \sum_{i=1}^{K} \Delta_i,$$

where $\zeta(\cdot)$ is the Riemann zeta function.

This bound presents the same characteristics of the one derived for UCB-L, e.g., $O(\log(N))$ regret and constant dependent from $\mu_{\max}$. The experimental results, presented in Section 5.5, provide empirical evidence that the introduction of the monotonicity assumption is improving the performance of UCB-LM even when we use the Chernoff bound to design MAB policies.

5.3 Exploiting Pricing Properties in Non-Stationary Setting

Since in a non-stationary environment $\mathcal{S}^{(B)}$ the outcome expected values $\mu_{i,\phi}$ might change as a new phase starts, we employ, similarly to [77], a Sliding Window (SW) approach for UCB-like algorithms. This approach takes decisions on the basis of the last τ rounds and, therefore, is capable of forgetting information coming from previous phases. At the same time, we integrate the information coming from the monotonicity property to speed up the learning process.

In what follows, we use the estimator for the outcome average value μ_i over the last $\min\{\tau, t\}$ rounds $\bar{X}_{i,t,\tau}$ and its realization $\bar{x}_{i,t,\tau}$, which are defined as:

$$\bar{X}_{i,t,\tau} := \frac{1}{T_i(t-1,\tau)} \sum_{s=T_i(\max\{t-\tau,1\})}^{T_i(t-1)} X_{i,s},$$

$$\bar{x}_{i,T_i(t-1,\tau),\tau} := \frac{1}{T_i(t-1,\tau)} \sum_{s=T_i(\max\{t-\tau,1\})}^{T_i(t-1)} x_{i,s},$$

where $T_i(t,\tau) = T_i(t) - T_i(\max\{t-\tau+1,1\})$ is the number of rounds the arm a_i has been selected in the last $\min\{\tau, t\}$ ones. Similarly to what has been considered for the UCB1-M algorithm, for each $1 \leq j \leq i$, let $\bar{X}_{ji,t,\tau}$ be the following linear combination of the random variables $\bar{X}_j, \dots, \bar{X}_i$ and $\bar{x}_{ji,t,\tau}$ its realization, defined as:

$$\bar{X}_{ji,t,\tau} := \frac{\sum_{k=j}^{i} T_k(t-1,\tau)\bar{X}_{k,t,\tau}}{T_{ji}(t-1,\tau)},$$

$$\bar{x}_{ji,t,\tau} := \frac{\sum_{k=j}^{i} T_k(t-1,\tau)\bar{x}_{k,t,\tau}}{T_{ji}(t-1,\tau)},$$

where $T_{ji}(t,\tau) = \sum_{k=j}^{i} T_k(t,\tau)$ is the number of rounds one of the arms in $\{a_j, \dots, a_i\}$ has been selected in the last $\min\{\tau, t\}$ ones. Given the monotonicity property and assuming to have samples to compute $\bar{x}_{ji,t,\tau}$ coming from the same phase Φ_ϕ we have:

$$\mu_{ji,\phi} = \mathbb{E}\left[\bar{X}_{ji,t,\tau}\right] \geq \mu_{i,\phi}.$$

Consider the following:

Theorem 6 (Corollary 21 in [77]). *Given a sequence $\{X_1, \dots, X_t\}$ of $t \in \mathbb{N}$ random variables with support $\Omega \subseteq [0,1]$ with expectation $\mu_h := \mathbb{E}[X_h]$ and a sequence $\{\epsilon_1, \dots, \epsilon_t\}$ a previsible sequence of Bernoulli random variables. For all $\tau \in \mathbb{N}$ and $\eta > 0$ it holds:*

$$\mathbb{P}\left(\frac{\sum_{h=\min\{t-\tau+1,1\}}^{t}(X_h - \mu_h)\epsilon_h}{\sum_{h=\min\{t-\tau+1,1\}}^{t} \epsilon_h}\right) \leq$$
$$\leq \left\lceil \frac{\log(\min\{t,\tau\})}{\log(1+\eta)} \right\rceil \exp\left\{-2\delta^2\left(1 - \frac{\eta^2}{16}\right)\right\}.$$

Algorithm 9: SW-UCB-M

```
Initialization
for $t \in \{1, \dots, K\}$ do
  Play arm $a_i$ and observe $x_{t,1}$
Loop
for $t \in \{K+1, \dots, N\}$ do
  for $i \in \{1, \dots, K\}$ do
    Compute:
      $$u_{i,t}^{(\text{SW-UCB-M})} = \min_{j \in \{1,\dots,i\}} \left\{ \bar{x}_{ji,t,\tau} + \sqrt{\frac{\xi\,(4\log(\min\{t,\tau\}) + \log(i))}{T_{ji}(t-1,\tau)}} \right\}$$
  Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} a_i u_{i,t}^{(\text{SW-UCB-M})}$ and observe $x_{i_t, T_{i_t}(t)}$
```

If we apply the previous result to the random variable $\bar{X}_{ji,t,\tau}$ and $\eta = 4\sqrt{1-\frac{2}{\xi}}$, with probability at least $1-\frac{p}{i}$, with $p \in [0,1]$, we have the following UCBs (from now on denoted as SW-UCB-M):

$$u_{ji,t}^{(\text{SW-UCB-M})} = \bar{x}_{ji,t,\tau} + \sqrt{\frac{\xi[\log(i) - \log(p)]}{T_{ji}(t-1,\tau)}} > \mu_{ji,\phi} \geq \mu_{i,\phi}, \tag{5.5}$$

where $\xi \in \mathbb{R}^+$ is a parameter used in the bound in [77].[3] Even in this case, we select $u_{i,t}^{(\text{SW-UCB-M})}$ as the tightest bound for $1 \leq j \leq i$, which holds with at least probability $1-p$, to decide which arm to pull next.

The pseudocode of the algorithm employing such a bound is presented in Algorithm 9 and presents characteristics similar to the bounds we propose in the previous section. Focusing on the SW-UCB-M algorithm, we can show that:

Theorem 7. *If policy SW-UCB-M is run over a non-stationary MAB setting $\mathcal{S}^{(B)}$, for any $\tau \in \mathbb{N}$ and $\xi > \frac{1}{2}$, the pseudo-regret after N rounds is at most:*

$$\bar{R}_N \leq \sum_{i=1}^{K} \left[\frac{N}{\tau} \frac{4a_i^2 \xi[\log(i) + \log(\tau)]}{\Delta_i} + a_i \Upsilon_N \tau + \right.$$
$$\left. + \frac{2N}{\tau} \left\lceil \frac{\log(\tau)}{\log\left(1 + 4\sqrt{1 - \frac{1}{2\xi}}\right)} \right\rceil \right],$$

[3] Here we assume that all the variables used to obtain $\bar{X}_{ji,t,\tau}$ are coming from a single phase Φ_ϕ.

where Υ_N is the number of breakpoints before N and

$$\Delta_i := \min_{\phi\in\{1,\dots,\Upsilon_N\}} \left(a_{i^*_\phi}\mu_{i^*_\phi,\phi} - a_i\mu_{i,\phi}\right)\mathbb{1}\{i \neq i^*_\phi\} \quad \forall i \in \{1,\dots,K\},$$

denotes the minimum, over all the phases Φ_ϕ in which the arm a_i is not optimal, of the difference of the expected reward $a_{i^*_\phi}\mu_{i^*_\phi,\phi}$ of the best arm $a_{i^*_\phi}$ and the expected reward $a_i\mu_{i,\phi}$ of the arm a_i.

5.4 Summary of the Proposed Algorithms

In the previous sections, we study how to exploit two properties of the pricing problem to improve the empiric performance of general-purpose bandit algorithms. We provide some algorithms that we summarize in Table 5.1, to ease the comprehension of the experimental evaluation. The first property is the decreasing monotonicity of the conversion rate in the price. The algorithms exploiting this property are specified with the suffix M (monotonicity). The second property is the *a priori* information about the maximum conversion rate $\mu_{\max}$. The algorithms exploiting this property are specified with the suffix L (low conversion rate). The algorithms exploiting both properties, are specified using the suffix LM. Furthermore, we focus both on stationary settings and non-stationary settings. We make use of a Sliding Window to tackle the non-stationarity of the environment, thus the algorithms are specified with the prefix SW.

	Stationary		Non-Stationary	
	Generic	Monotonic	Generic	Monotonic
$\mu \in [0,1]$	UCB1, UCBV	**UCB1-M,** **UCBV-M**	SW-UCB	**SW-UCB-M,** **SW-UCBV-M**
$\mu \in [0,\mu_{\max}]$	**UCB-L**	**UCB-LM**	**SW-UCB-L**	**SW-UCB-LM**

Table 5.1: *Algorithms for the different assumptions and scenarios analyzed in the chapter. We use the boldface for the algorithms proposed in this work.*

5.5 Experimental Analysis in Stationary Setting

We provide a thorough experimental evaluation of our algorithms in stationary environments, comparing them with the corresponding algorithms that do not exploit the two properties of the pricing problem we study.

5.5.1 Experimental Setting and Performance Indices

We evaluate our algorithms on a wide spectrum of configurations of pricing settings characterized by a different number of arms in A, by different pdfs $\mathcal{S}$, and by a different $\mu_{\max}$. Our configurations are generated from real-world data on the reselling of flight tickets by a European online travel agency.[4]

In particular, we use a number of arms $K \in \{5, 9, 17, 33\}$ evenly spaced over the interval $[1, 17]$, and the threshold pdfs $\mathcal{S}$ are as follows:[5]

- $\mathcal{S}_H \sim \mathcal{N}(20, 6)$, representing a situation where $a_{i^*} \geq 15$, i.e., the optimal price is among the highest values in $[1, 17]$ and for every i we have $\mu_i \in [0.68\mu_{\max}, \mu_{\max}]$, and
- $\mathcal{S}_L \sim \mathcal{N}(3, 5)$, representing a situation where $a_{i^*} \leq 5$, i.e., the optimal price is among the lowest values in $[1, 17]$ and for every i we have $\mu_i \in [0.0025\mu_{\max}, 0.66\mu_{\max}]$.

Configurations $\mathcal{S}_L$ and $\mathcal{S}_H$ represent the two extreme and most significant cases for the class of algorithms that make the assumption of optimism against uncertainty. More precisely, $\mathcal{S}_H$ is an easy configuration independently from the structure of the problem, since any algorithm based on the assumption of optimism against uncertainty can identify the best arm with a little exploration cost. Instead, $\mathcal{S}_L$ is a challenging configuration, since the identification of the best arm requires a large exploration cost. The values of $\mu_{\max}$ in the case of the reselling of flight tickets are usually in $\{1, 10^{-1}, 10^{-2}, 10^{-3}, 10^{-4}\}$, depending on the specific route and market. Let us observe that such a range includes the values of $\mu_{\max}$ of many scenarios different from the one we study, allowing us to provide an experimental evaluation of our algorithms also in other scenarios. More precisely, according to [79], $\mu_{\max} = 10^{-1}$ corresponds to Bing, Google, Yahoo!; $\mu_{\max} = 10^{-2}$ corresponds to Facebook, Pinterest, Twitter; $\mu_{\max} = 10^{-3}$ corresponds to LinkedIn; $\mu_{\max} = 10^{-4}$ corresponds to StumbleUpon. For each combination of $(K, \mathcal{S}, \mu_{\max})$, we average over 100 independent trials of length $N = 10^7$ rounds and in each round the threshold s is independently drawn from $\mathcal{S}$.

We compare our algorithms UCB1-M, UCB-L, and UCB-LM with the corresponding frequentist algorithms that do not exploit the two properties of the pricing problem we study: UCB1, UCBV, and UCBV-M (for the UCBV and UCBV-M algorithms, the parameters we use are $c = \xi = 1$). In our

[4] We do not specify the name of the online travel agency due to confidentiality issues.

[5] Here, we denote with $\mathcal{N}(\mu, \sigma)$ the normal distribution with mean μ and standard deviation σ.

evaluation, we use the following performance indices, for each $t \leq N$:

$$\begin{aligned}
R_{\%}(t) &= \tfrac{\bar{R}_t(\mathfrak{U})}{\bar{R}_t(\text{UCB1})}, \\
\Delta P(t) &= \textstyle\sum_{t'=1}^{t} \mathbb{E}\left[V_{i_{(\mathfrak{U},t')}}\right] - \sum_{t'=1}^{t} \mathbb{E}\left[V_{i_{(\text{UCB1},t')}}\right], \\
\Delta P_{\%}(t) &= \tfrac{\Delta P(t)}{\sum_{t'=1}^{t} \mathbb{E}[V_{i_{(\text{UCB1},t')}}]},
\end{aligned}$$

where $\mathfrak{U}$ is a generic policy, $i_{(\mathfrak{U},t)}$ is the index chosen by policy $\mathfrak{U}$ at time t. $R_{\%}(t)$ is defined as the ratio between the total regret of policy $\mathfrak{U}$ after t rounds and the regret of UCB1 that we use here as the baseline (a value of $R_{\%}$ lower than 1 means that $\mathfrak{U}$ outperforms UCB1 and the lower the value the greater the improvement); $\Delta P(t)$ is the difference between the cumulative expected reward of policy $\mathfrak{U}$ and the one obtained with UCB1; $\Delta P_{\%}$ is defined as the ratio between $\Delta P(t)$ and the cumulative expected reward obtained with UCB1. A value of ΔP (and $\Delta P_{\%}$) greater than 0 means that $\mathfrak{U}$ improves the profit with respect to UCB1 and the higher the value the greater the improvement.

5.5.2 Regret Analysis

The average $R_{\%}(N)$ and the 95% confidence intervals are reported in Table 5.2 and in Table 5.3 (the results of UCB-L and UCB-LM are omitted for $\mu_{\max} = 1$, their bound being theoretically worse than the one of UCB1). We omit the evaluation of $R_{\%}(t)$ for $t < N$, since we provide in the next section a detailed discussion about how the profit provided by the algorithms changes as t changes and we believe this latter evaluation is more significant in practice than the evaluation of the dependency of the regret on time.

We initially focus on the results obtained with $\mathcal{S}_L$. Here, UCBV-M outperforms all the other algorithms, with $R_{\%}(N)$ decreasing from 0.55 to 0.02 of the UCB1 regret. Furthermore, we observe that all the algorithms in the table outperform UCB1. While UCB1-M performs better than UCB-L only in some specific settings, UCB-LM outperforms both UCB1-M and UCB-L in all the configurations. Furthermore, UCB-LM performs usually worse than UCBV, except for very low values of $\mu_{\max}$ and many arms. These results strengthen the evidence that the use of the Chernoff's bound is effective when $\mu_{\max} \ll 1$. Instead, UCBV-M always outperforms UCBV reducing the regret of UCBV by a ratio up to $2/3$. We observe that the (relative) performance of the algorithms exploiting the monotonicity increases as the number of arms increases. This is because these algorithms better exploit the correlation among the arms. Finally, we observe that the best improvement (in terms of reduction of the regret) of our algorithms with respect to the per-

Table 5.2: *Results concerning $R_{\%}(N)$ with $\mathcal{S}_L$ (averaged values over 100 runs, ± 95% confidence intervals). The best results for each configuration are in boldface.*

		S_L				
$\mu_{\max}$	$\lvert A \rvert$	UCB1-M	UCB-L	UCB-LM	UCBV	UCBV-M
1	5	0.81 ± 0.01	——	——	0.22 ± 0.00	$\mathbf{0.20 \pm 0.00}$
	9	0.72 ± 0.01	——	——	0.24 ± 0.00	$\mathbf{0.19 \pm 0.00}$
	17	0.67 ± 0.01	——	——	0.26 ± 0.00	$\mathbf{0.20 \pm 0.00}$
	33	0.61 ± 0.01	——	——	0.31 ± 0.01	$\mathbf{0.23 \pm 0.01}$
10^{-1}	5	0.80 ± 0.00	0.42 ± 0.00	0.34 ± 0.00	0.03 ± 0.00	$\mathbf{0.02 \pm 0.00}$
	9	0.66 ± 0.00	0.45 ± 0.00	0.30 ± 0.00	$\mathbf{0.03 \pm 0.00}$	$\mathbf{0.03 \pm 0.00}$
	17	0.50 ± 0.00	0.50 ± 0.00	0.27 ± 0.00	0.05 ± 0.00	$\mathbf{0.04 \pm 0.00}$
	33	0.32 ± 0.00	0.54 ± 0.00	0.20 ± 0.00	0.06 ± 0.00	$\mathbf{0.04 \pm 0.00}$
10^{-2}	5	0.87 ± 0.00	0.30 ± 0.00	0.24 ± 0.00	$\mathbf{0.02 \pm 0.00}$	$\mathbf{0.02 \pm 0.00}$
	9	0.78 ± 0.00	0.49 ± 0.00	0.31 ± 0.00	0.05 ± 0.00	$\mathbf{0.04 \pm 0.00}$
	17	0.73 ± 0.00	0.65 ± 0.00	0.30 ± 0.00	0.11 ± 0.00	$\mathbf{0.07 \pm 0.00}$
	33	0.70 ± 0.00	0.77 ± 0.00	0.28 ± 0.00	0.17 ± 0.00	$\mathbf{0.08 \pm 0.00}$
10^{-3}	5	0.91 ± 0.00	0.83 ± 0.00	0.71 ± 0.00	0.17 ± 0.00	$\mathbf{0.15 \pm 0.00}$
	9	0.88 ± 0.00	0.88 ± 0.00	0.64 ± 0.00	0.33 ± 0.00	$\mathbf{0.22 \pm 0.00}$
	17	0.86 ± 0.00	0.92 ± 0.00	0.59 ± 0.00	0.47 ± 0.00	$\mathbf{0.22 \pm 0.00}$
	33	0.85 ± 0.00	0.94 ± 0.00	0.58 ± 0.00	0.60 ± 0.00	$\mathbf{0.22 \pm 0.00}$
10^{-4}	5	0.92 ± 0.00	0.96 ± 0.00	0.86 ± 0.00	0.67 ± 0.01	$\mathbf{0.55 \pm 0.01}$
	9	0.89 ± 0.00	0.97 ± 0.00	0.81 ± 0.00	0.73 ± 0.00	$\mathbf{0.50 \pm 0.01}$
	17	0.87 ± 0.00	0.98 ± 0.00	0.78 ± 0.00	0.77 ± 0.00	$\mathbf{0.48 \pm 0.01}$
	33	0.86 ± 0.00	0.98 ± 0.00	0.77 ± 0.00	0.80 ± 0.00	$\mathbf{0.48 \pm 0.01}$

formance of UCB1 is for $\mu_{\max} = 10^{-1}$. This is because when $\mu_{\max} = 1$ all the algorithms converge to the best arm before $N = 10^7$ rounds, minimizing the differences in terms of regret among them; when $\mu_{\max} \in \{10^{-1}, 10^{-2}\}$ our algorithms converge to the best arm before 10^7 rounds, while UCB1 does not, thus maximizing the differences in terms of regret among the algorithms; when $\mu_{\max} \in \{10^{-3}, 10^{-4}\}$ no algorithm converges to the best arm by 10^7 rounds, but some algorithms select the best arm more frequently than others.

Now, we focus on the results obtained with $\mathcal{S}_H$. Here, there is no algorithm that always outperforms the others. We observe that, for large values of $\mu_{\max}$, UCBV is the best algorithm, for intermediate values UCBV-M outperforms the others, and for small values UCB-LM is the best. The best algorithm presents $R_{\%}(N)$ in the range between 0.20 and 0.66. In details, UCB1 performs better than UCB1-M for some cases and, surprisingly, better than UCBV when $\mu_{\max}$ is very small. We observe that UCB-L, UCB-LM, and UCBV-M always perform better than UCB1. In some configurations UCB-LM improves over UCBV, halving the UCBV regret, e.g., in the configuration with $K = 33$ arms and $\mu_{\max} = 10^{-4}$, providing a significant improvement over UCBV performance. Differently from the case with $\mathcal{S}_L$,

Table 5.3: *Results concerning $R_{\%}(N)$ with $\mathcal{S}_H$ (averaged values over* 100 *runs,* ± *95% confidence intervals). The best results for each configuration are in boldface.*

		$\mathcal{S}_H$				
$\mu_{\max}$	$\lvert A\rvert$	UCB1-M	UCB-L	UCB-LM	UCBV	UCBV-M
1	5	1.01 ± 0.02	——	——	$\mathbf{0.20 \pm 0.01}$	$\mathbf{0.21 \pm 0.01}$
	9	1.01 ± 0.03	——	——	$\mathbf{0.28 \pm 0.01}$	0.31 ± 0.01
	17	1.02 ± 0.02	——	——	$\mathbf{0.45 \pm 0.02}$	0.50 ± 0.02
	33	1.02 ± 0.01	——	——	$\mathbf{0.37 \pm 0.01}$	0.42 ± 0.01
10^{-1}	5	1.03 ± 0.02	0.60 ± 0.02	0.60 ± 0.02	$\mathbf{0.23 \pm 0.01}$	$\mathbf{0.24 \pm 0.01}$
	9	0.98 ± 0.01	0.63 ± 0.01	0.63 ± 0.01	$\mathbf{0.22 \pm 0.01}$	$\mathbf{0.23 \pm 0.01}$
	17	0.86 ± 0.01	0.65 ± 0.01	0.59 ± 0.01	$\mathbf{0.31 \pm 0.01}$	$\mathbf{0.29 \pm 0.01}$
	33	0.67 ± 0.01	0.69 ± 0.01	0.54 ± 0.01	0.42 ± 0.01	$\mathbf{0.36 \pm 0.01}$
10^{-2}	5	0.93 ± 0.00	0.30 ± 0.01	0.29 ± 0.01	$\mathbf{0.21 \pm 0.01}$	$\mathbf{0.22 \pm 0.01}$
	9	0.85 ± 0.00	0.38 ± 0.01	0.35 ± 0.01	$\mathbf{0.25 \pm 0.01}$	$\mathbf{0.25 \pm 0.01}$
	17	0.75 ± 0.00	0.37 ± 0.00	0.28 ± 0.00	0.29 ± 0.01	$\mathbf{0.22 \pm 0.01}$
	33	0.67 ± 0.00	0.42 ± 0.00	0.25 ± 0.00	0.37 ± 0.00	$\mathbf{0.21 \pm 0.00}$
10^{-3}	5	1.26 ± 0.00	$\mathbf{0.31 \pm 0.01}$	$\mathbf{0.30 \pm 0.01}$	0.33 ± 0.01	$\mathbf{0.32 \pm 0.01}$
	9	1.28 ± 0.00	0.44 ± 0.01	$\mathbf{0.36 \pm 0.01}$	0.46 ± 0.01	$\mathbf{0.37 \pm 0.01}$
	17	1.30 ± 0.00	0.49 ± 0.01	$\mathbf{0.34 \pm 0.01}$	0.58 ± 0.01	$\mathbf{0.34 \pm 0.01}$
	33	1.30 ± 0.00	0.57 ± 0.00	$\mathbf{0.35 \pm 0.01}$	0.74 ± 0.01	$\mathbf{0.35 \pm 0.01}$
10^{-3}	5	1.46 ± 0.00	$\mathbf{0.55 \pm 0.01}$	$\mathbf{0.54 \pm 0.01}$	0.89 ± 0.02	0.78 ± 0.02
	9	1.51 ± 0.00	0.68 ± 0.01	$\mathbf{0.63 \pm 0.01}$	1.03 ± 0.01	0.83 ± 0.02
	17	1.57 ± 0.00	0.74 ± 0.01	$\mathbf{0.64 \pm 0.01}$	1.20 ± 0.01	0.86 ± 0.02
	33	1.59 ± 0.00	0.79 ± 0.01	$\mathbf{0.66 \pm 0.01}$	1.33 ± 0.01	0.86 ± 0.02

with $\mathcal{S}_H$ the relative improvement of algorithms exploiting the monotonicity does not increase as the number of arms increases. The same holds for the UCBV algorithm, which does not exploit any assumption, suggesting that the performance of the UCB1 algorithm improves as the number of arms increases in the $\mathcal{S}_H$ setting. This is probably due to the fact that UCB1 is able to exclude many arms easily if the optimal values of the expected reward are realized on high arms, e.g., if $\mu_i a_i > a_j$ it will not play arms lower or equal to a_j. Thus, UCB1 is effectively working on a smaller set of arms than A and this leads to low regret even for this policy.

To summarize, our algorithms, specifically UCBV-M and UCB-LM, provide a significant improvement in terms of regret with respect to the algorithms available in the state of the art.

5.5.3 Profit Analysis

The average $\Delta P_{\%}(t)$ and $\Delta P(t)$ for $t \in \{1, \dots, 10^7\}$ obtained with UCB-LM and UCBV-M (with respect to the results obtained with UCB1 and 5

arms) are reported in Figure 5.3 and Figure 5.4, respectively.[6]

Initially, we focus on the results obtained with $\mu_{\max} \in \{1, 10^{-1}\}$ and $\mathcal{S}_L$. The value of $\Delta P_{\%}(t)$ dramatically changes during the time horizon. It reaches a maximum around $t = 10^4$ for $\mu_{\max} = 1$ and $t = 10^5$ for $\mu_{\max} = 10^{-1}$ and then it decreases approaching the value of zero at $t = 10^7$. The improvement is significant, the maximum of $\Delta P_{\%}(t)$ being about 2.2 for UCBV-M (i.e., the profit of UCB1 is more than tripled) and about 1.3 for UCB-LM (i.e., the profit is more than doubled). In the case of $\mathcal{S}_H$, the value of $\Delta P_{\%}(t)$ initially reaches a minimum and subsequently a maximum, and finally approaches zero as t goes to 10^7. In this case, the improvement is less significant than the one we have in the case of $\mathcal{S}_L$ and $\Delta P_{\%}(t)$ is about 0.003 when $\mu_{\max} = 1$ and 0.033 when $\mu_{\max} = 10^{-1}$, meaning an improvement of 0.3% and 3.3% over the UCB1 profit, respectively.

Now we focus on the results obtained with $\mu_{\max} = 10^{-2}$ and $\mathcal{S}_L$. The maximum of $\Delta P_{\%}(t)$ is reached in the range between $t = 10^6$ and $t = 10^7$ (that is, very close to the termination of the time horizon). The improvement is very significant, $\Delta P_{\%}(t)$ achieving values larger than 3.3. The behavior for $\mathcal{S}_H$ is analogous with respect to the one with larger $\mu_{\max}$. Here, we can observe that the minimum is achieved for a larger t than in the setting with $\mu_{\max} \in \{1, 10^{-1}\}$. The maximum of $\Delta P_{\%}(t)$ is about 0.04. Finally, we focus on the results obtained with $\mu_{\max} \in \{10^{-3}, 10^{-4}\}$ and $\mathcal{S}_L$. The $\Delta P_{\%}(t)$ trend suggests that its maximum might be beyond 10^7 rounds. Nevertheless, the improvement is very significant: $\Delta P_{\%}(t)$ is larger than 3 for $\mu_{\max} = 10^{-3}$ and almost 2 for $\mu_{\max} = 10^{-4}$. As for smaller values of $\mu_{\max}$, the improvements with $\mathcal{S}_H$ are less significant. Nevertheless, UCB-LM presents a maximum of $\Delta P_{\%}(t)$ that is almost 0.08 even with $\mu_{\max} = 10^{-4}$.

Furthermore, we observe how the performance of the algorithms varies as the number of arms varies in the two different pdfs. With $\mathcal{S}_L$ the best improvement is achieved when the number of arms is 33 with $\mu_{\max} \leq 10^{-1}$ and 5 otherwise. Instead, with $\mathcal{S}_H$, the best improvement is achieved, in the most cases, when using 33 arms.

To summarize, our algorithms, specifically UCBV-M and UCB-LM, provide a significant improvement in terms of relative profit especially in the early stages of the learning process.

[6] Results for UCB-LM in the case $\mu_{\max} = 1$ are not reported since this algorithm requires $\mu_{\max} < \frac{1}{2}$ to be effective. Moreover, the results for $\Delta P(t)$ for $\mathcal{S}_H$ are not reported since they are less significant.

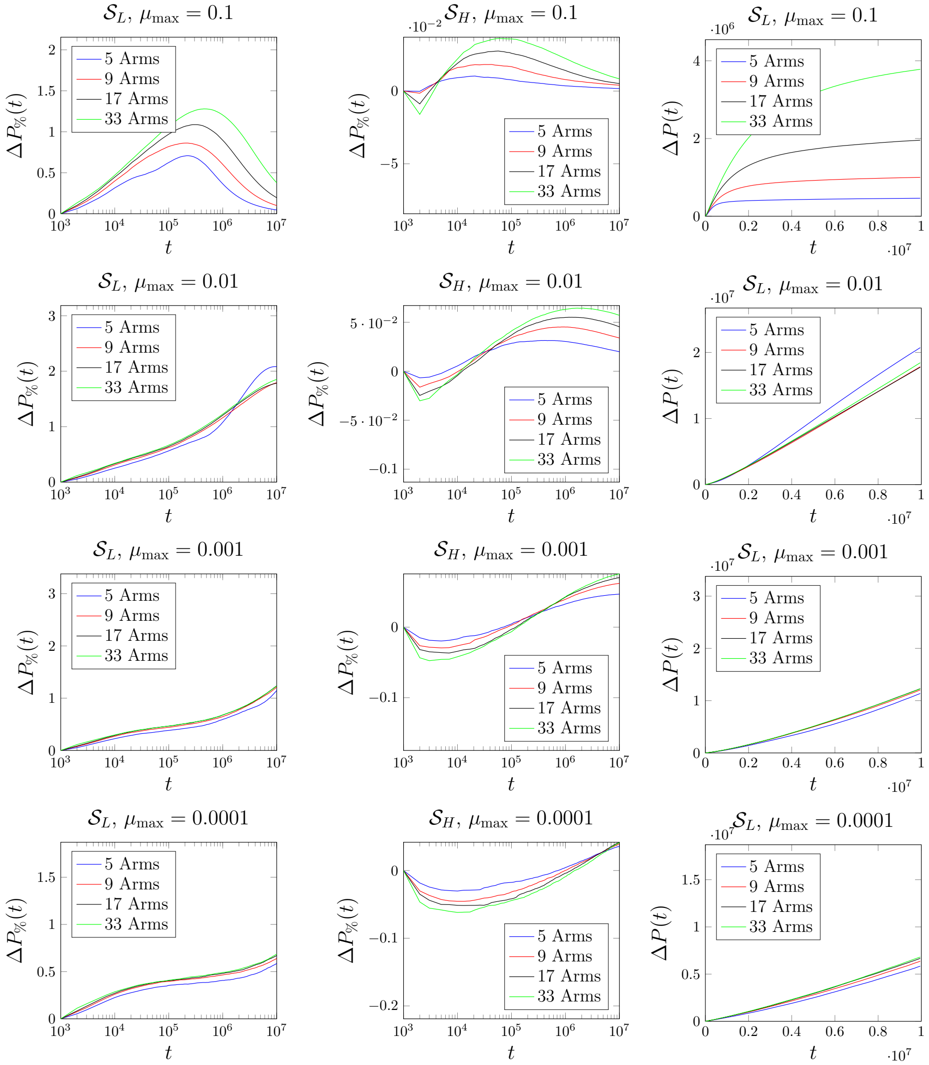


Figure 5.3: *$\Delta P_{\%}(t)$ (first two columns) and $\Delta P(t)$ (third column) obtained with UCB-LM with different configurations.*

5.6 Experimental Analysis in Non-Stationary Setting

We experimentally evaluate the performance of our techniques in an abruptly changing environment, that, as aforementioned, is one of the most common non-stationary settings in e-commerce. We compare the SW-UCB-M algorithm with UCBV-M, as representatives of algorithms exploiting the monotonicity assumption, and SW-UCB from [77], as representatives of frequen-

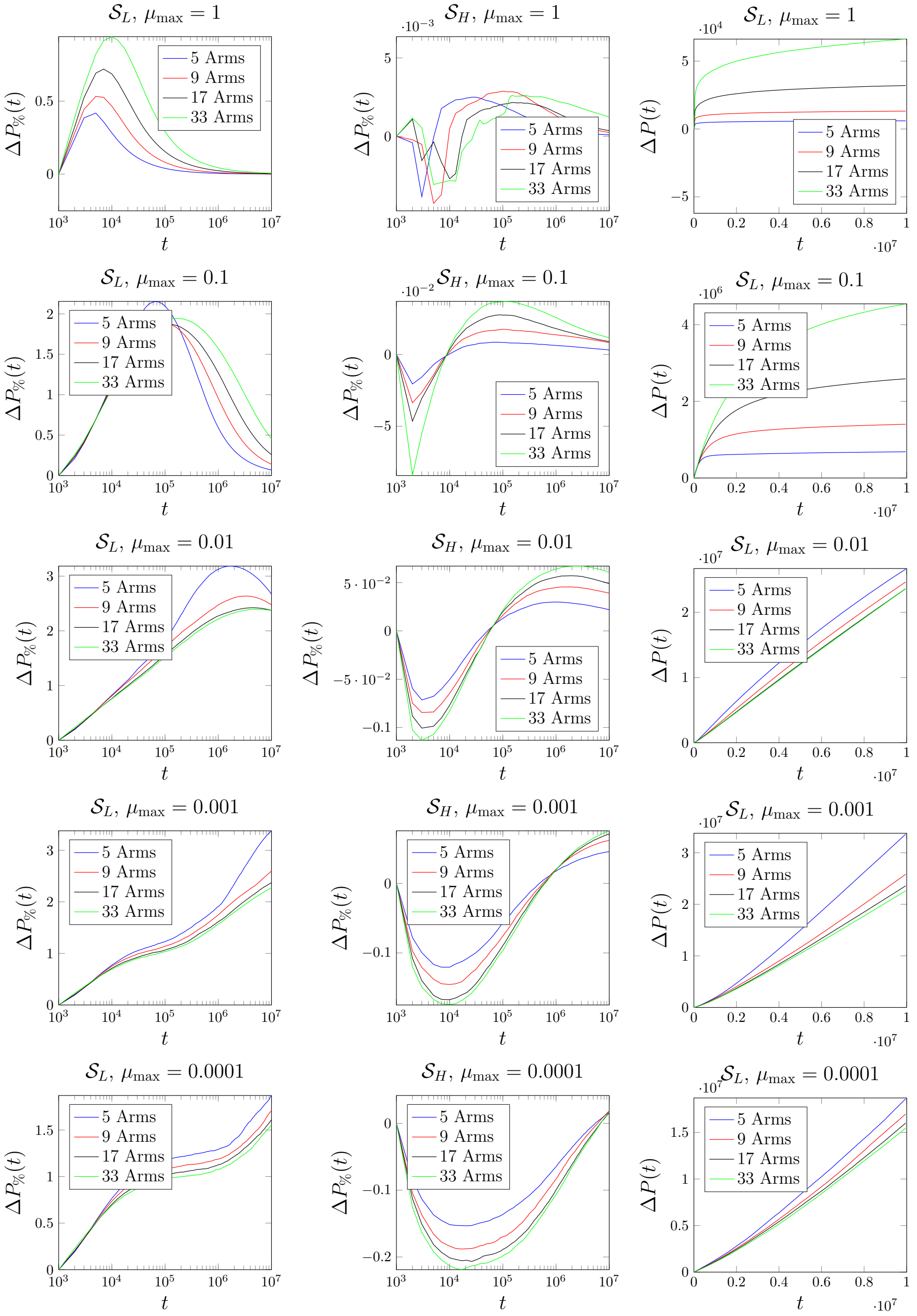


Figure 5.4: *$\Delta P_{\%}(t)$ (first two columns) and $\Delta P(t)$ (third column) obtained with UCBV-M with different configurations.*

tist MAB designed for non-stationary environments. In addition to the already presented SW-UCB-M, we extend the sliding window approach to other algorithms proposing the SW-UCB-LM and the SW-UCBV-M algorithms, to include the maximum probability information and to consider UCBV-like algorithms with the monotonicity information, respectively. The pseudocode of these algorithms is provided in Appendix B.1.

5.6.1 Experimental Setting and Performance Indices

The experimental setting considers a number of rounds of $N = 4 \cdot 10^7$ and uses two different abruptly changing pdfs, denoted with $\mathcal{S}_{LHLH}$ and $\mathcal{S}_{HLHL}$, each of which contains three breakpoints at rounds $t = 10^7$, $t = 2 \cdot 10^7$ and $t = 3 \cdot 10^7$. The threshold pdf switches from $\mathcal{S}_L$ to $\mathcal{S}_H$ or *vice versa* for $\mathcal{S}_{LHLH}$ and $\mathcal{S}_{HLHL}$, respectively, where $\mathcal{S}_L$ and $\mathcal{S}_H$ are defined as in Section 5.5. For instance, $\mathcal{S}_{LHLH}$ starts with $\mathcal{S}_L$ in phase Φ_1, then switches to $\mathcal{S}_H$ in phase Φ_2 and so on. For the sliding window algorithms, we choose a sliding window $\tau = 4\sqrt{N \log(N)}$ and we consider a parameter $\xi = 0.6$ for SW-UCB and SW-UCB-M, as in [77]. We average the results over 100 independent trials.

We redefine the performance indices using SW-UCB as baseline in place of UCB1 as follows:

$$\begin{aligned}
R_{\%}(t) \quad &= \tfrac{\bar{R}_t(\mathfrak{U})}{\bar{R}_t(\text{SW-UCB})}, \\
\Delta P(t) \quad &= \textstyle\sum_{t'=1}^{t} \mathbb{E}\left[V_{i_{(\mathfrak{U},t')}}\right] - \sum_{t'=1}^{t} \mathbb{E}\left[V_{i_{(\text{SW-UCB},t')}}\right], \\
\Delta P_{\%}(t) \quad &= \tfrac{\Delta P(t)}{\sum_{t'=1}^{t} \mathbb{E}[V_{i_{(\text{SW-UCB},t')}}]},
\end{aligned}$$

where $\mathfrak{U}$ is a generic policy, $i_{(\mathfrak{U},t)}$ is the index chosen by policy $\mathfrak{U}$ at time t.

5.6.2 Regret Analysis

The average $R_{\%}(N)$ and the 95% confidence intervals are reported in Table 5.4 and in Table 5.5 (the results of SW-UCB-L and SW-UCB-LM are omitted for $\mu_{\max} = 1$, their bound being always larger than the one used in SW-UCB). As in the stationary case, we omit the evaluation of $R_{\%}(t)$ for $t < N$, since we provide in the next section a detailed discussion about how the profit provided by the algorithms changes as t changes and we believe this latter evaluation is more significant in practice than the evaluation of the dependency of the regret on time.

The first observation we provide is that, except for some specific cases, the performance in terms of regret of each algorithm is similar in the two configurations $\mathcal{S}_{LHLH}$ and $\mathcal{S}_{HLHL}$. This shows that the switches between

Table 5.4: *Results concerning $R_{\%}$ in non-stationary settings with S_{LHLH} (averaged values over 100 runs, ± 95% confidence intervals).*

		S_{LHLH}					
$\mu_{\max}$	$\|A\|$	SW-UCB-M	SW-UCB-L	SW-UCB-LM	SW-UCBV	SW-UCBV-M	UCBV-M
1	5	1.02 ± 0.00	——	——	$\mathbf{0.95 \pm 0.00}$	0.98 ± 0.02	10.67 ± 0.00
	9	1.82 ± 0.26	——	——	$\mathbf{0.94 \pm 0.00}$	1.46 ± 0.18	10.18 ± 0.00
	17	2.29 ± 0.35	——	——	$\mathbf{0.92 \pm 0.00}$	2.44 ± 0.37	9.57 ± 0.00
	33	2.96 ± 0.44	——	——	$\mathbf{0.91 \pm 0.00}$	3.49 ± 0.44	8.60 ± 0.06
10^{-1}	5	0.82 ± 0.01	1.08 ± 0.00	0.88 ± 0.01	0.33 ± 0.00	$\mathbf{0.25 \pm 0.02}$	5.71 ± 0.00
	9	0.72 ± 0.01	1.05 ± 0.00	0.75 ± 0.01	0.51 ± 0.00	$\mathbf{0.40 \pm 0.05}$	4.77 ± 0.00
	17	$\mathbf{0.63 \pm 0.02}$	1.04 ± 0.00	0.64 ± 0.01	0.66 ± 0.00	$\mathbf{0.67 \pm 0.13}$	4.41 ± 0.03
	33	$\mathbf{0.57 \pm 0.02}$	1.04 ± 0.00	$\mathbf{0.57 \pm 0.01}$	0.82 ± 0.00	0.68 ± 0.13	4.01 ± 0.08
10^{-2}	5	0.98 ± 0.00	0.88 ± 0.00	0.79 ± 0.00	0.74 ± 0.00	$\mathbf{0.58 \pm 0.01}$	3.37 ± 0.05
	9	0.98 ± 0.00	0.89 ± 0.00	0.76 ± 0.00	0.88 ± 0.00	$\mathbf{0.59 \pm 0.01}$	3.10 ± 0.03
	17	0.97 ± 0.00	0.90 ± 0.00	0.71 ± 0.00	0.98 ± 0.00	$\mathbf{0.60 \pm 0.01}$	2.97 ± 0.07
	33	0.97 ± 0.00	0.92 ± 0.00	0.69 ± 0.01	1.07 ± 0.00	$\mathbf{0.60 \pm 0.02}$	2.78 ± 0.13
10^{-3}	5	1.10 ± 0.00	0.92 ± 0.00	$\mathbf{0.90 \pm 0.00}$	1.09 ± 0.00	1.08 ± 0.00	3.00 ± 0.11
	9	1.14 ± 0.00	0.93 ± 0.00	$\mathbf{0.92 \pm 0.00}$	1.14 ± 0.00	1.14 ± 0.00	2.65 ± 0.14
	17	1.17 ± 0.00	0.94 ± 0.00	$\mathbf{0.93 \pm 0.00}$	1.19 ± 0.00	1.18 ± 0.00	1.84 ± 0.24
	33	1.19 ± 0.00	0.96 ± 0.00	$\mathbf{0.93 \pm 0.00}$	1.21 ± 0.00	1.20 ± 0.01	1.25 ± 0.23
10^{-4}	5	1.12 ± 0.00	$\mathbf{0.97 \pm 0.00}$	1.04 ± 0.00	1.24 ± 0.00	1.49 ± 0.00	1.34 ± 0.24
	9	1.17 ± 0.00	0.97 ± 0.00	1.08 ± 0.00	1.24 ± 0.00	1.56 ± 0.00	$\mathbf{0.57 \pm 0.01}$
	17	1.21 ± 0.00	0.98 ± 0.00	1.11 ± 0.00	1.26 ± 0.00	1.63 ± 0.00	$\mathbf{0.55 \pm 0.01}$
	33	1.23 ± 0.00	0.98 ± 0.00	1.12 ± 0.00	1.26 ± 0.00	1.64 ± 0.01	$\mathbf{0.54 \pm 0.00}$

L and H and *vice versa* do not significantly affect the performance of the algorithms. Instead, the performance depends on the number of L and H configurations. This holds for all the algorithms, $\mu_{\max}$ values, and number of arms, except for the following special cases:

- UCBV-M: it performs much worse in $\mathcal{S}_{LHLH}$ than in $\mathcal{S}_{HLHL}$ for $\mu_{\max} \in \{10^{-1}, 10^{-2}, 10^{-3}\}$. This result does not depend on the exploitation of the monotonicity, but on the fact that, once UCBV-M has learned a configuration L or H, its bounds do not significantly change after an abrupt change given that it does not exploit any sliding window and the optimal arm in the configuration L has a very small relative reward in configuration H. This does not hold when $\mu_{\max} = 10^{-4}$ since the sliding window is excessively small and the baseline SW-UCB cannot learn anything.

- SW-UCB-M and SW-UCBV-M: they perform worse in $\mathcal{S}_{LHLH}$ than in $\mathcal{S}_{HLHL}$ for $\mu_{\max} = 1$. This is an anomaly of our algorithms. In this specific case, the cost of exploiting the monotonicity is larger than the gain provided by the algorithm.

Table 5.5: *Results concerning $R_{\%}$ in non-stationary settings with S_{HLHL} (averaged values over 100 runs, ± 95% confidence intervals).*

		S_{HLHL}					
$\mu_{\max}$	$\|A\|$	SW-UCB-M	SW-UCB-L	SW-UCB-LM	SW-UCBV	SW-UCBV-M	UCBV-M
1	5	1.01 ± 0.00	——	——	$\mathbf{0.96 \pm 0.00}$	$\mathbf{0.97 \pm 0.01}$	1.44 ± 0.00
	9	0.99 ± 0.00	——	——	$\mathbf{0.95 \pm 0.00}$	$\mathbf{0.96 \pm 0.01}$	1.32 ± 0.00
	17	0.97 ± 0.01	——	——	$\mathbf{0.93 \pm 0.00}$	$\mathbf{0.94 \pm 0.01}$	1.29 ± 0.00
	33	0.92 ± 0.01	——	——	$\mathbf{0.89 \pm 0.00}$	$\mathbf{0.89 \pm 0.01}$	1.18 ± 0.00
10^{-1}	5	0.86 ± 0.01	1.08 ± 0.00	0.90 ± 0.01	0.33 ± 0.00	$\mathbf{0.29 \pm 0.01}$	1.43 ± 0.00
	9	0.75 ± 0.01	1.05 ± 0.00	0.78 ± 0.01	0.51 ± 0.00	$\mathbf{0.43 \pm 0.03}$	1.10 ± 0.00
	17	0.66 ± 0.01	1.04 ± 0.00	0.67 ± 0.01	0.66 ± 0.00	$\mathbf{0.53 \pm 0.07}$	1.04 ± 0.01
	33	$\mathbf{0.59 \pm 0.01}$	1.04 ± 0.00	$\mathbf{0.60 \pm 0.01}$	0.82 ± 0.00	$\mathbf{0.57 \pm 0.07}$	0.95 ± 0.01
10^{-2}	5	0.98 ± 0.00	0.88 ± 0.00	0.79 ± 0.00	0.74 ± 0.00	$\mathbf{0.60 \pm 0.01}$	0.85 ± 0.00
	9	0.98 ± 0.00	0.89 ± 0.00	0.76 ± 0.00	0.88 ± 0.00	$\mathbf{0.64 \pm 0.01}$	0.76 ± 0.01
	17	0.97 ± 0.00	0.90 ± 0.00	0.73 ± 0.00	0.98 ± 0.00	$\mathbf{0.63 \pm 0.01}$	0.74 ± 0.01
	33	0.97 ± 0.00	0.92 ± 0.00	0.71 ± 0.00	1.07 ± 0.00	$\mathbf{0.63 \pm 0.01}$	0.72 ± 0.01
10^{-3}	5	1.10 ± 0.00	0.92 ± 0.00	0.90 ± 0.00	1.09 ± 0.00	1.08 ± 0.00	$\mathbf{0.83 \pm 0.03}$
	9	1.14 ± 0.00	0.93 ± 0.00	0.92 ± 0.00	1.14 ± 0.00	1.14 ± 0.00	$\mathbf{0.76 \pm 0.02}$
	17	1.17 ± 0.00	0.94 ± 0.00	0.93 ± 0.00	1.19 ± 0.00	1.18 ± 0.00	$\mathbf{0.75 \pm 0.02}$
	33	1.19 ± 0.00	0.96 ± 0.00	0.94 ± 0.00	1.21 ± 0.00	1.21 ± 0.00	$\mathbf{0.75 \pm 0.01}$
10^{-4}	5	1.12 ± 0.00	0.97 ± 0.00	1.04 ± 0.00	1.24 ± 0.00	1.49 ± 0.00	$\mathbf{0.85 \pm 0.02}$
	9	1.17 ± 0.00	0.97 ± 0.00	1.08 ± 0.00	1.24 ± 0.00	1.56 ± 0.00	$\mathbf{0.82 \pm 0.01}$
	17	1.21 ± 0.00	0.98 ± 0.00	1.10 ± 0.00	1.26 ± 0.00	1.63 ± 0.00	$\mathbf{0.81 \pm 0.01}$
	33	1.23 ± 0.00	0.98 ± 0.00	1.12 ± 0.00	1.26 ± 0.00	1.65 ± 0.00	$\mathbf{0.80 \pm 0.01}$

Summarily, we can observe that: SW-UCBV is the optimal algorithm for $\mu_{\max} = 1$, SW-UCBV-M is the optimal algorithm for $\mu_{\max} \in \{10^{-1}, 10^{-2}\}$, and UCBV-M is the optimal algorithm for $\mu_{\max} \in \{10^{-3}, 10^{-4}\}$ except in the configuration $\mathcal{S}_{LHLH}$, where, instead, for $\mu_{\max} = 10^{-3}$ SW-UCB-LM is the best one. This is because the exploitation of the monotonicity allows an algorithm to perform better, but it requires a cost, i.e., the one incurred when a union bound over $1 \leq j \leq i$ is performed. When the setting is easy (e.g., $\mu_{\max}$ is very high), the improvement provided by the monotonicity is smaller than the cost needed for its exploitation. Instead, for $\mu_{\max} \in \{10^{-1}, 10^{-2}\}$, the cost required for the exploitation of the monotonicity is much lower than the gain. When $\mu_{\max}$ is smaller, e.g., $\mu_{\max} \in \{10^{-3}, 10^{-4}\}$, the setting is too hard and we suppose that an optimal solution to the problem would require a sliding window longer than that one used here. Indeed, the fact that UCBV-M is the best algorithms shows essentially that abstaining from learning after the first abrupt change is better than trying to learn the change. In these settings that are so hard, a different approach should be used: for instance, one could identify the abrupt change and employ different stationary MAB policies, one per phase.

Finally, we remark that in every configuration it is possible to outperform the baseline and in many cases the reduction of regret is significant.

5.6.3 Profit Analysis

The average $\Delta P_{\%}(t)$ and $\Delta P(t)$ for $t \in \{1, \dots, 10^7\}$ obtained with SW-UCB-LM and SW-UCBV-M (with respect to the results obtained with SW-UCB and 5 arms) are reported in Figure 5.5 and Figure 5.6, respectively.[7]

The main difference between the results in the stationary settings and those in non-stationary settings concerns the trend of $\Delta P_{\%}(t)$. In the case of the stationary settings, $\Delta P_{\%}(t)$ achieves a maximum and subsequently goes asymptotically to zero, showing that our algorithms provide a gain in the early stages of the learning process. Instead, in the case of non-stationary settings, our algorithms repeatedly provide a gain at each abrupt change. This is showed by the fact that $\Delta P_{\%}(t)$ does not go to zero as t increases. Therefore, the $\Delta P(t)$ continuously increases over time. To summarize, these results provide evidence for a promising application of the proposed SW algorithms in the non-stationary setting.

[7] Results for SW-UCB-LM in the case $\mu_{\max} = 1$ are not reported since this algorithm requires $\mu_{\max} < \frac{1}{2}$ to be effective.

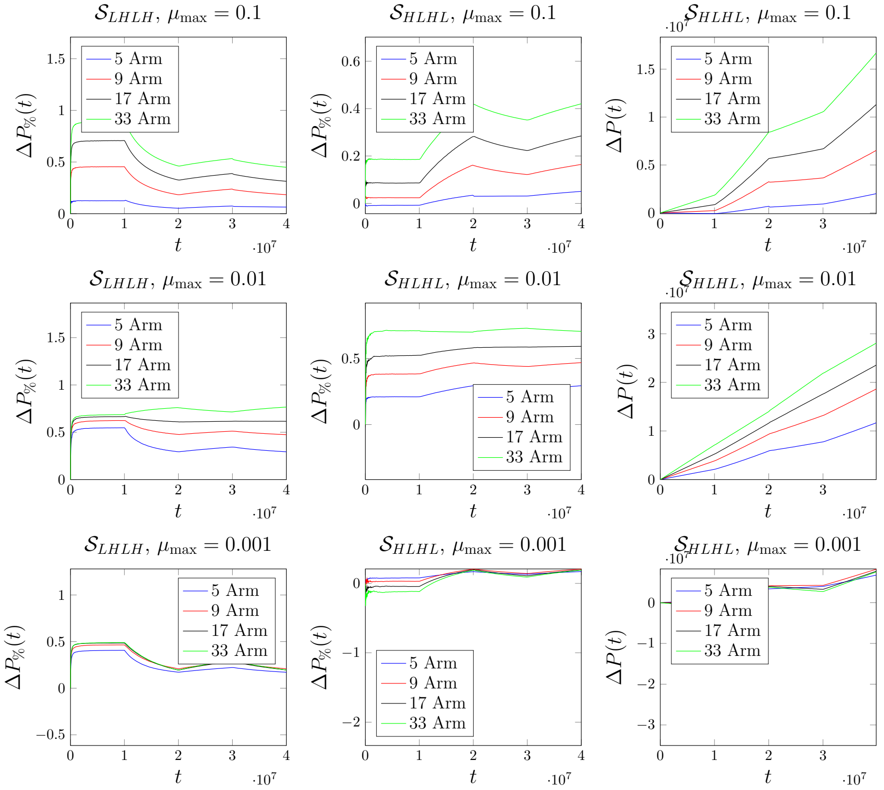


Figure 5.5: $\Delta P_{\%}(t)$ *(first two columns) and* $\Delta P(t)$ *(third column) obtained with SW-UCB-LM with different configurations.*

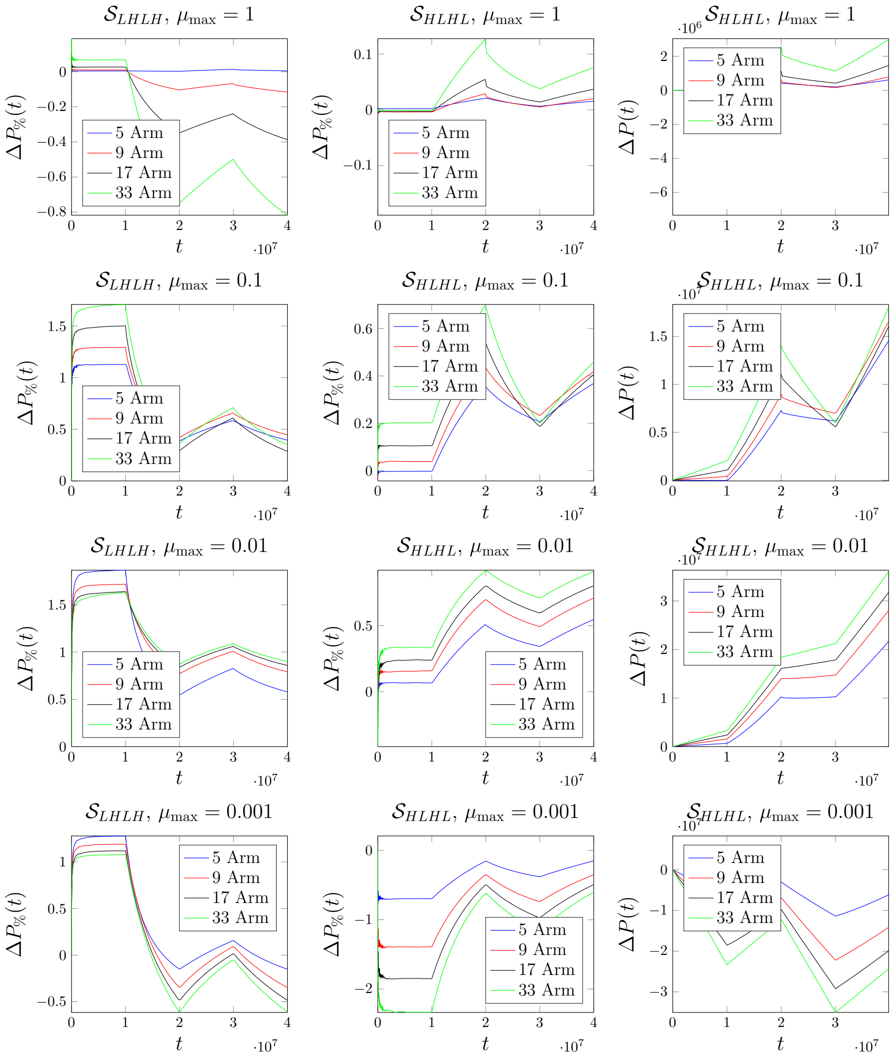


Figure 5.6: *$\Delta P_{\%}(t)$ (first two columns) and $\Delta P(t)$ (third column) obtained with SW-UCBV-M with different configurations.*

5.7 From Frequentist to Bayesian

In the previous sections, we provided a wide experimental evaluation of our algorithms, comparing them with frequentist MAB algorithms present in literature. We evaluated the improvement obtained thanks to the exploitation of the problem characteristics and we elected two algorithms as the best ones: UCBV-M and UCB-LM in stationary settings and SW-UCBV-M and SW-UCB-LM in non-stationary ones. In most of the configurations, our algorithms perform better than the general-purpose ones. However, it is well known that Bayesian MAB algorithms usually suffer of same order of regret as the best frequentist one (e.g., in unstructured settings [10]), but they outperform the frequentist methods in a wide range of problems (e.g., in bandit problems without structure [80] and in bandit problems with budget [81]). Furthermore, in problems with structure, the classical Thompson Sampling in its original formulation may outperform frequentist algorithms exploiting the problem structure. For this reason, in this section, we evaluate the performance of the best frequentist algorithms we elected with the one of Thompson Sampling (TS).

We compare UCBV-M, UCB-LM and TS with the same configurations of Section 5.5, and we compare SW-UCBV-M, SW-UCB-LM and TS with the same configurations of Section 5.6. We use UCB1 as the baseline in non-stationary settings for the performance index $R_{\%}(N)$, while in the non-stationary ones we use SW-UCB.

In Table 5.6, we reported the average $R_{\%}(N)$ and the 95% confidence intervals of the results in the stationary setting. TS outperforms our algorithms in almost all the configurations. Not only TS reaches the best performances, but it also provide a significant improvement in terms of regret with respect to our algorithms. Just in the case with $\mu_{\max} = 10^{-3}$ in configuration S_H, TS is not able to outperform UCB-LM, proving that Chernoff's bound is more efficient when mu_max is really low.

In Table 5.7, we reported the average $R_{\%}(N)$ and the 95% confidence intervals of the results in the non-stationary setting. With configuration S_{LHLH}, TS is not able to outperform our algorithms and its performance are much worser than the one of the baseline SW-UCB. This behavior may be due to the fact that TS is not using a sliding window approach. With configuration S_{HLHL}, instead, TS outperforms our algorithm in the case with $\mu_{\max} \in \{10^{-3}, 10^{-4}\}$. With $\mu_{\max} = 10^{-2}$, even though TS is not able to outperform our algorithms, it provide an improvement in terms of regret with respect to the baseline SW-UCB, showing that, even without the use of a sliding window, it reaches better performances.

We remark that, in both the stationary and non-stationary setting, we used the classical Thompson Sampling, without exploiting the pricing problem structure, i.e., the information about the monotonicity and the low conversion rates. In non-stationary settings, TS is not even using a sliding window to update its posterior distribution. The results show that Thompson Sampling outperforms the frequentist methods in the stationary pricing problem setting and in some of the non-stationary setting we considered. For these reasons, in the next chapter we focus on the design of Thompson Sampling-based algorithms.

Table 5.6: *Results concerning $R_{\%}(N)$ with $\mathcal{S}_L$ and $\mathcal{S}_H$ (averaged values over 100 runs, $\pm$ 95% confidence intervals) of TS versus frequentist algorithms. The best results for each configuration are in boldface.*

			S_L	
$\mu_{\max}$	$\lvert A\rvert$	UCB-LM	UCBV-M	TS
1	5	——	0.20 ± 0.00	$\mathbf{0.06 \pm 0.00}$
	9	——	0.19 ± 0.00	$\mathbf{0.07 \pm 0.00}$
	17	——	0.20 ± 0.00	$\mathbf{0.08 \pm 0.00}$
	33	——	0.23 ± 0.01	$\mathbf{0.13 \pm 0.01}$
10^{-1}	5	0.34 ± 0.00	0.02 ± 0.00	$\mathbf{0.01 \pm 0.00}$
	9	0.30 ± 0.00	0.03 ± 0.00	$\mathbf{0.01 \pm 0.00}$
	17	0.27 ± 0.00	0.04 ± 0.00	$\mathbf{0.02 \pm 0.00}$
	33	0.20 ± 0.00	0.04 ± 0.00	$\mathbf{0.02 \pm 0.00}$
10^{-2}	5	0.24 ± 0.00	0.02 ± 0.00	$\mathbf{0.00 \pm 0.00}$
	9	0.31 ± 0.00	0.04 ± 0.00	$\mathbf{0.01 \pm 0.00}$
	17	0.30 ± 0.00	0.07 ± 0.00	$\mathbf{0.03 \pm 0.00}$
	33	0.28 ± 0.00	0.08 ± 0.00	$\mathbf{0.05 \pm 0.00}$
10^{-3}	5	0.71 ± 0.00	0.15 ± 0.00	$\mathbf{0.03 \pm 0.00}$
	9	0.64 ± 0.00	0.22 ± 0.00	$\mathbf{0.07 \pm 0.00}$
	17	0.59 ± 0.00	0.22 ± 0.00	$\mathbf{0.13 \pm 0.00}$
	33	0.58 ± 0.00	0.22 ± 0.00	$\mathbf{0.21 \pm 0.00}$
10^{-4}	5	0.86 ± 0.00	0.55 ± 0.01	$\mathbf{0.18 \pm 0.01}$
	9	0.81 ± 0.00	0.50 ± 0.01	$\mathbf{0.31 \pm 0.01}$
	17	0.78 ± 0.00	0.48 ± 0.01	$\mathbf{0.43 \pm 0.01}$
	33	0.77 ± 0.00	$\mathbf{0.48 \pm 0.01}$	0.56 ± 0.01

			S_H	
$\mu_{\max}$	$\lvert A\rvert$	UCB-LM	UCBV-M	TS
1	5	——	0.21 ± 0.01	$\mathbf{0.10 \pm 0.05}$
	9	——	0.31 ± 0.01	$\mathbf{0.10 \pm 0.01}$
	17	——	0.50 ± 0.02	$\mathbf{0.25 \pm 0.03}$
	33	——	0.42 ± 0.01	$\mathbf{0.15 \pm 0.01}$
10^{-1}	5	0.60 ± 0.02	0.24 ± 0.01	$\mathbf{0.11 \pm 0.02}$
	9	0.63 ± 0.01	0.23 ± 0.01	$\mathbf{0.09 \pm 0.02}$
	17	0.59 ± 0.01	0.29 ± 0.01	$\mathbf{0.13 \pm 0.01}$
	33	0.54 ± 0.01	0.36 ± 0.01	$\mathbf{0.23 \pm 0.01}$
10^{-2}	5	0.29 ± 0.01	0.22 ± 0.01	$\mathbf{0.11 \pm 0.03}$
	9	0.35 ± 0.01	0.25 ± 0.01	$\mathbf{0.12 \pm 0.01}$
	17	0.28 ± 0.00	0.22 ± 0.01	$\mathbf{0.13 \pm 0.01}$
	33	0.25 ± 0.00	0.21 ± 0.00	$\mathbf{0.18 \pm 0.01}$
10^{-3}	5	0.30 ± 0.01	0.32 ± 0.01	$\mathbf{0.22 \pm 0.02}$
	9	0.36 ± 0.01	0.37 ± 0.01	$\mathbf{0.23 \pm 0.02}$
	17	0.34 ± 0.01	0.34 ± 0.01	$\mathbf{0.27 \pm 0.01}$
	33	$\mathbf{0.35 \pm 0.01}$	$\mathbf{0.35 \pm 0.01}$	$\mathbf{0.37 \pm 0.01}$
10^{-4}	5	0.54 ± 0.01	0.78 ± 0.02	$\mathbf{0.43 \pm 0.04}$
	9	$\mathbf{0.63 \pm 0.01}$	0.83 ± 0.02	$\mathbf{0.59 \pm 0.03}$
	17	$\mathbf{0.64 \pm 0.01}$	0.86 ± 0.02	0.70 ± 0.02
	33	$\mathbf{0.66 \pm 0.01}$	0.86 ± 0.02	0.87 ± 0.02

Table 5.7: *Results concerning $R_{\%}$ in non-stationary settings with S_{LHLH} and S_{HLHL} (averaged values over 100 runs, ± 95% confidence intervals) of TS versus frequentist algorithms. The best results for each configuration are in boldface.*

		S_{LHLH}		
$\mu_{\max}$	$\lvert A\rvert$	SW-UCB-LM	SW-UCBV-M	TS
1	5	——	**0.98 ± 0.02**	8.93 ± 0.68
	9	——	**1.46 ± 0.18**	7.37 ± 0.78
	17	——	**2.44 ± 0.37**	5.11 ± 0.72
	33	——	**3.49 ± 0.44**	4.55 ± 0.65
10^{-1}	5	0.88 ± 0.01	**0.25 ± 0.02**	3.60 ± 0.46
	9	0.75 ± 0.01	**0.40 ± 0.05**	2.61 ± 0.36
	17	0.64 ± 0.01	**0.67 ± 0.13**	1.72 ± 0.29
	33	**0.57 ± 0.01**	0.68 ± 0.13	2.32 ± 0.30
10^{-2}	5	0.79 ± 0.00	**0.58 ± 0.01**	2.14 ± 0.25
	9	0.76 ± 0.00	**0.59 ± 0.01**	1.44 ± 0.22
	17	0.71 ± 0.00	**0.60 ± 0.01**	1.58 ± 0.21
	33	0.69 ± 0.01	**0.60 ± 0.02**	1.89 ± 0.20
10^{-3}	5	**0.90 ± 0.00**	1.08 ± 0.00	1.82 ± 0.23
	9	**0.92 ± 0.00**	1.14 ± 0.00	1.35 ± 0.18
	17	**0.93 ± 0.00**	1.18 ± 0.00	1.60 ± 0.19
	33	**0.93 ± 0.00**	1.20 ± 0.01	1.55 ± 0.17
10^{-4}	5	**1.04 ± 0.00**	1.49 ± 0.00	1.45 ± 0.21
	9	**1.08 ± 0.00**	1.56 ± 0.00	1.28 ± 0.15
	17	**1.11 ± 0.00**	1.63 ± 0.00	**1.12 ± 0.12**
	33	1.12 ± 0.00	1.64 ± 0.01	**0.81 ± 0.07**

		S_{HLHL}		
$\mu_{\max}$	$\lvert A\rvert$	SW-UCB-LM	SW-UCBV-M	TS
1	5	——	**0.97 ± 0.01**	1.71 ± 0.08
	9	——	**0.96 ± 0.01**	1.80 ± 0.11
	17	——	**0.94 ± 0.01**	1.62 ± 0.08
	33	——	**0.89 ± 0.01**	1.42 ± 0.08
10^{-1}	5	0.90 ± 0.01	**0.29 ± 0.01**	1.58 ± 0.06
	9	0.78 ± 0.01	**0.43 ± 0.03**	1.39 ± 0.08
	17	0.67 ± 0.01	**0.53 ± 0.07**	1.17 ± 0.06
	33	**0.60 ± 0.01**	**0.57 ± 0.07**	1.06 ± 0.05
10^{-2}	5	0.79 ± 0.00	**0.60 ± 0.01**	0.94 ± 0.04
	9	0.76 ± 0.00	**0.64 ± 0.01**	0.86 ± 0.04
	17	0.73 ± 0.00	**0.63 ± 0.01**	0.80 ± 0.03
	33	0.71 ± 0.00	**0.63 ± 0.01**	0.80 ± 0.03
10^{-3}	5	0.90 ± 0.00	1.08 ± 0.00	**0.84 ± 0.03**
	9	0.92 ± 0.00	1.14 ± 0.00	**0.79 ± 0.03**
	17	0.93 ± 0.00	1.18 ± 0.00	**0.78 ± 0.02**
	33	0.94 ± 0.00	1.21 ± 0.00	**0.83 ± 0.03**
10^{-4}	5	1.04 ± 0.00	1.49 ± 0.00	**0.86 ± 0.02**
	9	1.08 ± 0.00	1.56 ± 0.00	**0.85 ± 0.02**
	17	1.10 ± 0.00	1.63 ± 0.00	**0.86 ± 0.02**
	33	1.12 ± 0.00	1.65 ± 0.00	**0.87 ± 0.01**

CHAPTER 6

Multi-Armed Bandit for Pricing: Bayesian Approach

In the previous chapter we propose techniques to apply to frequentist policies in order to exploit the pricing problem structure, but we finally show how Thompson Sampling, the most popular Bayesian algorithm, outperforms the designed algorithms even without exploiting the problem structure. In this chapter, we focus on Bayesian MAB and we design novel algorithms based on Thompson Sampling.

First, in Section 6.1 we introduce an update scheme to exploit the monotonicity property for Bayesian policies, showing that it is hard to obtain a closed form solution and to assure theoretical guarantees to a version of TS exploiting the monotonicity property.

Then, we study the Non-Stationary MAB (NS-MAB) settings, proposing an algorithm based on Thompson Sampling which exploits a sliding-window approach to tackle, in a unified fashion, two different forms of non-stationarity studied separately so far: *abruptly changing* and *smoothly changing*. In the former, the reward distributions are constant during sequences of rounds and change at unknown rounds, while, in the latter, the reward distributions smoothly evolve over rounds. Section 6.2 provides the formulation for the NS-MAB setting. In Section 6.3, we describe the proposed algorithm to tackle the NS-MAB setting and we derive upper bounds over the pseudo-regret for the algorithm. In Section 6.4, we empirically show that our algorithm dramatically outperforms the state-of-the-art algorithms even when the forms of non-stationarity are taken separately as previously studied in the literature.

Finally, we focus on the Unimodal MAB (UMAB) setting, in which each arm corresponds to a node of a graph and each edge is associated with a relationship specifying which node of the edge gives the largest expected reward (providing thus a partial ordering over the arm space). While the graph structure may be (not necessarily) known *a priori* by the UMAB algorithm, the relationship defined over the edges is discovered during the learning. Section 6.5 provides the formulation for the UMAB setting. In Section 6.6, we propose a novel Bayesian MAB algorithm and we derive upper bounds over the pseudo-regret for the algorithm. In Section 6.7, we describe a wide experimental campaign showing better performance of our algorithm in applicative scenarios than those of state-of-the-art ones, evaluating also how the performance of the considered algorithms varies as the graph structure properties vary.

6.1 Exploiting the Pricing Problem Structure

In the case of no monotonicity existing on A the Bayesian update due to the outcome $x_{i_t,t}$ at time t is:

$$\mathbb{P}_t(\mu_{i_t}) := \mathbb{P}_{t-1}(\mu_i | x_{i_t,t}) \propto \mathbb{P}_{t-1}(x_{i_t}|\mu_{i_t})\mathbb{P}_{t-1}(\mu_{i_t}),$$

where a single arm a_{i_t} is updated at each time point.[1] On the other hand, when the monotonicity property holds on A as defined in Section 5.1, it is possible to update all the arms at each time point. The corresponding updating scheme due to a realization $x_{i_t,t}$ is:

$$\mathbb{P}_t(\mu_i) \propto \begin{cases} \mathbb{P}_{t-1}(\mu_i)\mathbb{P}_{t-1}(x_{i_t,t}|\mu_i) & i_t = i \\ \mathbb{P}_{t-1}(\mu_i)\dfrac{\int_0^{\mu_i} \mathbb{P}(x_{i_t,t}|\mu_j)\mathbb{P}_{t-1}(M_{i_t} = \mu_j)\mathrm{d}\mu_j}{\int_0^{\mu_i} \mathbb{P}_{t-1}(M_{i_t} = x)\mathrm{d}x} & i_t > i \\ \mathbb{P}_{t-1}(\mu_i)\dfrac{\int_{\mu_i}^1 \mathbb{P}(x_{i_t,t}|\mu_j)\mathbb{P}_{t-1}(M_{i_t} = \mu_j)\,\mathrm{d}\mu_j}{\int_{\mu_i}^1 \mathbb{P}_{t-1}(M_{i_t} = x)\mathrm{d}x} & i_t < i \end{cases}, \quad (6.1)$$

[1] From now on, as usual in probability theory, we will use the notation $\mathbb{P}(x) := \mathbb{P}(X = x)$, whenever there is no ambiguity.

where M_i is the random variable whose distribution is the prior on μ_i. In what follows, we focus on the case where each variable X_i has Bernoulli distribution on $\Omega \in \{0, 1\}$. Notice that, when Equation (6.1) is used as update rule, Beta distribution is no more conjugate prior. Since we do not have a closed form solution, we can resort to a non-parametric scheme, i.e., Sequential Monte Carlo (SMC) technique [82] to represent and to update the prior distribution. More specifically, a scheme where the prior $\mathbb{P}_{t-1}(\mu_i)$ is represented by a finite number $N_p \in \mathbb{N}$ of particles $P_i = \{p_{i1}, \dots p_{iN_p}\}$, $p_{ih} \in [0, 1]$, $\forall h \in \{1, \dots, N_p\}$ and their corresponding weights $W_i = \{w_{i1}, \dots, w_{iN_p}\}$, $w_{ih} \in \mathbb{R}^+$, $\forall h \in \{1, \dots, N_p\}$. In this case the update scheme in Equation (6.1) becomes:

$$w_{ih} \leftarrow \begin{cases} w_{ih} p_{i_t,h}^{x_{i_t,t}} (1 - p_{i_t,h})^{1-x_{i_t,t}} & i_t = i \\ w_{ih} \dfrac{\sum_{h | p_{i_t,h} \leq p_{i,h}} p_{i_t,h}^{x_{i_t,t}} (1 - p_{i_t,h})^{1-x_{i_t,t}} w_{i_t,h}}{\sum_{h | p_{i_t,h} \leq p_{i,h}} w_{i_t,h}} & i_t > i \\ w_{ih} \dfrac{\sum_{h | p_{i_t,h} \geq p_{i,h}} p_{i_t,h}^{x_{i_t,t}} (1 - p_{i_t,h})^{1-x_{i_t,t}} w_{i_t,h}}{\sum_{h | p_{i_t,h} \geq p_{i,h}} w_{i_t,h}} & i_t < i \end{cases} . \qquad (6.2)$$

Since we arrived to an approximated solution, we are not able to provide any theoretical guarantee on the regret bounds. Thus, in the Bayesian framework, the exploitation of the monotonicity turns out to be a not promising research direction. As previously stated, even if an heuristic algorithm might perform better than algorithms with theoretical guarantees, the lack of worst-case guarantees discourages their employment in practice. We are studying an applicative scenario, so we are not interested in evaluating algorithm without theoretical guarantees. For this reason, in the next sections of this chapter, we do not consider the exploitation of the pricing problem properties as done in Chapter 5 with UCB-like algorithms, but we focus on two other interesting features of our problem: the non-stationarity of the environment and the unimodality of the expected profit of the set of arms.

6.2 Non-Stationary MAB: Problem Formulation

We model our problem as a stochastic Non-Stationary MAB (NS-MAB) setting, in which, at each round t over a finite horizon N, the learner selects an arm a_{i_t} among a finite set of K arms $A = \{a_1, \dots, a_K\}$. At each round t the learner observes a realization of the reward $x_{i_t,t}$ obtained from the chosen arm a_{i_t}. The rewards for each arm a_i at round t are modeled by a sequence of i.i.d. random variables $X_{i,t}$ from a distribution unknown to the learner. We denote by $\mu_{i,t} := \mathbb{E}[X_{i,t}]$ the expected value of the reward of arm a_i at round t. As customary in the MAB literature, here we consider Bernoulli distributed rewards, i.e., $X_{i,t} \sim Be(\mu_{i,t})$.[2] A *policy* $\mathfrak{U}$ is a function $\mathfrak{U}(h_t) = a_{i_t}$ that chooses the arm a_{i_t} to play at round t according to history h_t, defined as the sequence of past plays and obtained rewards.

The goal of the learner is to design a policy $\mathfrak{U}$ that minimizes the loss w.r.t. the optimal decision in terms of reward. This loss, usually addressed as cumulative dynamic *pseudo-regret*, is defined as:

$$\bar{R}_N(\mathfrak{U}) = \mathbb{E}\left[\sum_{t=1}^{N} \left(\mu_{i_t^*,t} - \mu_{i_t,t}\right)\right], \tag{6.3}$$

where $\mu_{i_t^*,t} = \max_{i \in \{1,\dots,K\}} \mu_{i,t}$ is the expected reward of the optimal arm $a_{i_t^*}$ at round t and $\mathbb{E}\left[\cdot\right]$ is the expectation w.r.t. the stochasticity of the policy. Differently from the classical (stationary) stochastic MAB setting, where an arm (unique unless degeneracy) is optimal for the all-time horizon ($i_t^* = i^*, \forall t$), in the NS-MAB setting the arms that are optimal might change during time. We recall that when the optimal arm expected value can change without any restriction, the NS-MAB setting is equivalent to an adversarial MAB one, which has been studied in the past [33]. In what follows, we will discuss two different settings where the evolution over time of the arms reward distributions is constrained to change according to specific schemes.

6.2.1 Abruptly Changing Setting

The Abruptly Changing MAB (AC-MAB) setting is introduced in [52]. The reward distributions are constant during sequences of rounds, said *phases*, and change at unknown rounds, said *breakpoints*. Thus, the expected value $\mu_{i,t}$ of the reward of an arm a_i at round t only changes at the beginning of each phase and therefore the best arm $a_{i_t^*}$ remains the same during the whole phase.

Let us define a breakpoint as a round $b \in \{1, \dots, N\}$ s.t. $\exists i \mid \mu_{i,b-1} \neq \mu_{i,b}$, i.e., a round b in which the expected reward of at least one arm a_i changes w.r.t. the one at round $b-1$. In an AC-MAB setting with horizon N we have a set of breakpoints $B = \{b_1, \dots, b_{\Upsilon_N}\}$ of cardinality Υ_N (for sake of notation we define $b_0 = 1$), which determine a set of phases $\{\Phi_\phi\}_{\phi=1}^{\Upsilon_N}$, where each phase is set of rounds between two consecutive breakpoints, namely, $\Phi_\phi = \{t \in \{1, \dots, N\} \text{ s.t. } b_{\phi-1} \leq t < b_\phi\}$. In order to have sublinear pseudo-regret, we upper bound the number of breakpoints over the time horizon. We do that by making the following assumption:

[2] The extension to other distributions is straightforward. Bernoulli variables are considered here for sake of simplicity.

Assumption 1. *There exists $\alpha \in [0, 1)$, independent from N, s.t. the number of breakpoints Υ_N is of order $O(N^\alpha)$. That is, there exist $\alpha \in [0, 1)$ and $\Upsilon \in \mathbb{R}^+$ such that: $\Upsilon_N \leq \Upsilon N^\alpha$.*

During phase Φ_ϕ, with abuse of notation, we denote with $\mu_{i,\phi}$ the expected value of the reward of arm a_i, with $a_{i^*_\phi}$ the optimal arm, and with $\mu_{i^*,\phi}$ the corresponding expected reward. By defining the length of a phase as $N_\phi := |\Phi_\phi|$, a more compact formulation of the cumulative pseudo-regret of a generic policy $\mathfrak{U}$ over an AC-MAB is available:

$$\bar{R}_N(\mathfrak{U}) = \sum_{i=1}^{K} \sum_{\phi=1}^{\Upsilon_N} \Delta_{i,\phi} \mathbb{E}[T_i(\Phi_\phi)],$$

where $\sum_{\phi=1}^{\Upsilon_N} N_\phi = N$, $T_i(\Phi_\phi) = \sum_{t \in \Phi_\phi} \mathbb{1}\{i_t = i\}$ is the number of times the arm a_i has been pulled during phase Φ_ϕ, $\Delta_{i,\phi} := \mu_{i^*,\phi} - \mu_{i,\phi}$ is the difference between the expected reward $\mu_{i^*,\phi}$ of the optimal arm $a_{i^*_\phi}$ of phase Φ_ϕ and the expected reward $\mu_{i,\phi}$ of arm a_i, and $\mathbb{E}[\cdot]$ is the expectation w.r.t. the stochasticity of the policy.[3]

6.2.2 Smoothly Changing Setting

The Smoothly Changing MAB (SC-MAB) setting we study is similar to that one studied in [37], where the expected value $\mu_{i,t}$ of each arm varies smoothly over time. More formally, we make the following Lipschitz assumption:

Assumption 2. *There exists $\sigma > 0$ constant w.r.t time horizon N, such that $|\mu_{i,t} - \mu_{i,t'}| \leq \sigma |t - t'|$ for all $t, t' \in \{1, \ldots, N\}$ and all $i \in \{1, \ldots, K\}$.*

Furthermore, in such a setting, a suboptimal arm a_i might be arbitrarily close to the optimal one $a_{i^*_t}$ in terms of expected reward. Identifying the best arm among those with similar expected expected reward is known to be hard [8]. Indeed, it is known that a learner takes a time of the order of $\frac{1}{(\mu_{i^*_t,t} - \mu_{i,t})}$. Thus, to prevent the regret from being linearly dependent on the horizon N, we assume also that the separation between the expected rewards of two arms is arbitrarily small only for a limited number of rounds.[4] More formally, consider $0 < \Delta < 1$, we define:

$$\Phi_{\Delta,N} := \{t \in \{1, \ldots, N\} \text{ s.t. } \exists i \neq j, |\mu_{i,t} - \mu_{j,t}| < \Delta\}$$

and we assume that:

[3] From now on we denote with $|\cdot|$ the cardinality operator and with $\mathbb{1}\{A\}$ the indicator function of event A.

[4] This is a slightly stronger assumption than the one considered in [37] which allows us to provide, in what follows, a regret of order $\widetilde{O}(\sqrt{N})$.

Assumption 3. *There exist $\beta \in [0, 1)$, $\mathcal{F} \in \mathbb{R}^+$ and $\Delta_0 \in (0, 1)$, all independent from N, s.t. for all $\Delta < \Delta_0$ it holds $|\Phi_{\Delta,N}| \leq \mathcal{F}\Delta N^\beta$.*

6.2.3 Abruptly and Smoothly Changing Setting

Finally, in a quite straightforward way, it is possible to study also a scenario, from now on addressed as Abruptly and Smoothly Changing MAB (ASC-MAB) setting, in which the two forms of non-stationarity (abrupt changes and smooth ones) occur over a finite time period. In this setting, we have that Assumption 1 and Assumption 3 hold, and we also have that:

Assumption 4. *There exists $\sigma > 0$ constant w.r.t the time horizon N, such that:*

$$|\mu_{i,t} - \mu_{i,t'}| \leq \sigma \, |t - t'|$$

for all $i \in \{1, \ldots, K\}$ and all $t, t' \in \{1, \ldots, N\} \setminus B$, i.e., the expected value of the reward function is Lipschitz continuous for all the rounds except the breakpoints.

This newly defined assumption is the natural extension of Assumption 2 to this new setting, in which the smoothness assumption might be violated if the process is at a breakpoint.

6.3 The Sliding-Window Thompson Sampling Algorithm

We propose an algorithm that exploits a Sliding-Window (SW) approach to forget past information during the learning process which could provide a bias to the estimation process. More precisely, at round t, we take into account only the rewards obtained in the last τ rounds.[5] Based on these realizations, we apply a TS-based algorithm to decide which arm should be selected in the next round. More specifically, each arm expected value is coupled with a posterior distribution from which we extract samples. By choosing the sample with the highest value we decide which is the next arm to play. For sake of clarity, at first, we describe the algorithm and provide the finite-time analysis of its pseudo-regret $R_N(\mathfrak{U})$ for Bernoulli distributed rewards separately for the AC-MAB and SC-MAB and, after that, we analyze the ASC-MAB settings in which both the non-stationary processes (abrupt and smoothly changing) are present at the same time.[6]

[5] As showed in the following, the optimal τ depends on the parameters of the problem. In Appendix B.2, we provide the sensitivity analysis of τ, showing how the algorithm performance degrades when τ is not optimal.

[6] We report the complete proofs of the analysis in Appendix A.2.3, Appendix A.2.4, and Appendix A.2.5.

Algorithm 10: SW-TS

```
Input: $\{\pi_{i,0}\}_i$ prior distributions, $N$ time horizon, $A$ arm set
for $t \in \{1, \dots, N\}$ do
    for $i \in \{1, \dots, K\}$ do
        Compute $\pi_{i,t} = \text{Beta}(S_{i,t,\tau} + 1, T_{i,t,\tau} - S_{i,t,\tau} + 1)$
        Sample $\theta_{i,t}$ from $\pi_{i,t}$
    Play arm $a_{i_t}$ s.t.: $i_t = \arg\max_{i \in \{1,\dots,K\}} \theta_{i,t}$ and observe $x_{i_t,t}$
```

6.3.1 The SW-TS Pseudo-code

The pseudocode of SW-TS for Bernoulli distributed rewards is presented in Algorithm 10. Assume to have a prior $\pi_{i,0}$ on each reward expected value $\mu_{i,t}$ and let $\pi_{i,t}$ be the posterior distribution for the parameter $\mu_{i,t}$ after t rounds. In the case we consider a uniform uninformative prior, we choose $\pi_{i,0} := \text{Beta}(1, 1)$, where we denote with $\text{Beta}(a, b)$ the Beta distribution with parameters a and b, and the posterior becomes $\pi_{i,t} := \text{Beta}(S_{i,t,\tau} + 1, T_{i,t,\tau} - S_{i,t,\tau} + 1)$, where $T_{i,t,\tau} := \sum_{s=\max\{t-\tau+1,1\}}^{t} \mathbb{1}\{i_s = i\}$ is the number of times the arm a_i has been selected in the last $\min\{t, \tau\}$ rounds, and $S_{i,t,\tau} := \sum_{s=\max\{t-\tau+1,1\}}^{t} x_{i,s}\mathbb{1}\{i_s = i\}$ is the cumulative reward of the arm a_i in the last $\min\{t, \tau\}$ rounds.[7] Once computed the posterior distributions $\pi_{i,t}$, we draw a random sample $\theta_{i,t}$, also known as *Thompson sample*, from each distribution. Finally, we select the arm a_i with the highest sample $\theta_{i,t}$ for this round. The extension of the SW-TS algorithm to the case where the rewards $X_{i,t}$ are not Bernoulli distributed is similar to what proposed for the classical TS algorithm [80], given that conjugate prior/posterior distributions are available.

6.3.2 Finite-Time Analysis in the Abruptly Changing Setting

We provide a finite-time analysis of the pseudo-regret achieved by SW-TS algorithm, in the AC-MAB setting introduced in Section 6.2.1.

Theorem 8. *If policy SW-TS is run over an AC-MAB setting with* $X_{i,t} \sim$

[7] In what follows, we omit to explicitly state the dependence on τ with a subscript when there is no ambiguity in doing so.

Be$(\mu_{i,t})$*, for any* $\tau \in \mathbb{N}$*, the pseudo-regret after* N *rounds is at most:*

$$\bar{R}_N(\mathfrak{U}) \leq \sum_{i=1}^{K} \Bigg[\tau \Upsilon N^{\alpha} + \\ + \sum_{\phi=1}^{\Upsilon_N} \Delta_{i,\phi} \frac{N_\phi}{\tau} \left(\frac{56 \log \tau}{\Delta_{i,\phi}^2} + \log \tau + 4 + \frac{22}{\log \tau} + \frac{1}{\tau^{\frac{1}{2}}} \right) \Bigg],$$

where Υ *and* α *are defined in Assumption 1 and* $\Delta_{i,\phi} := \mu_{i^*,\phi} - \mu_{i,\phi}$ *is the difference between the expected reward* $\mu_{i^*,\phi}$ *of the best arm* $a_{i^*_\phi}$ *and the expected reward* $\mu_{i,\phi}$ *of arm* a_i*. By defining:*

$$\Delta_i := \min_{\phi \in \{1,\dots,\Upsilon_N\}} \Delta_{i,\phi} \mathbb{1}\{i \neq i^*_\phi\},$$

for all $i \in \{1, \dots, K\}$*, i.e., the minimum over all the phases* Φ_ϕ *of the difference of the expected rewards* $\Delta_{i,\phi}$*, the pseudo-regret becomes:*

$$\bar{R}_N(\mathfrak{U}) \leq \tau K \Upsilon N^{\alpha} + \frac{N}{\tau} \sum_{i=1}^{K} \left(\frac{56 \log \tau}{\Delta_i^2} + \log \tau + 4 + \frac{22}{\log \tau} + \frac{1}{\tau^{\frac{1}{2}}} \right).$$

By using a sliding window $\tau \propto N^{\frac{1-\alpha}{2}}$ in Theorem 8, the pseudo-regret $\bar{R}_N(\mathfrak{U})$ of the SW-TS is of the order $\tilde{O}(N^{\frac{1+\alpha}{2}})$. In particular, if Assumption 1 holds for $\alpha = 0$, meaning that the number of breakpoints is constant w.r.t. the time horizon, and we use a sliding window $\tau \propto \sqrt{N}$, the order of the pseudo-regret is $\tilde{O}(\sqrt{N})$. Conversely, if Assumption 1 does not hold ($\alpha \geq 1$), the above bound would provide a linear upper bound on the pseudo-regret over the time horizon. Notice that the sliding window approach outperforms classical MAB algorithms for stationary settings, e.g., UCB1, even with a single breakpoint ($\alpha = 0$ and $\Upsilon_N = 1$) and two arms. Those algorithms would suffer from $\Omega(\sqrt{N})$ regret in the second phase, in addition to the customary regret due to the first phase.[8]

Before providing a sketch of the proof (the complete proof is provided in Appendix A.2.2), we present a lemma used in what follows, which might be of independent interest to the reader.

Lemma 1. *Consider a random variable* B *with Beta distribution Beta*$(S + 1, T - S + 1)$*, where* $S := \sum_{s=1}^{T} X_s$ *is the sum of* $T \in \mathbb{N}$ *Bernoulli trials*

[8] For instance, suppose we have 2 arms a_1, a_2 and a breakpoint $b_1 = N/2$ in which the expected values of the arms switch (a_1 is better than a_2 before $N/2$ and worse after). After $N/2$, $O(\sqrt{N})$ pulls of a_1 are required before the upper confidence bound of a_2 overcomes the one of a_1, leading to a regret of at least $\Omega(\sqrt{N})$.

$X_s \sim Be(\mu)$ with same parameter $\mu \in [0,1]$. Consider a finite integer $\tau \in \mathbb{N}, \tau > T$, a parameter $\varepsilon > \frac{1}{2}$ and:

$$u_T := \frac{S}{T} + \sqrt{\frac{\varepsilon \log \tau}{T}},$$
$$q_T := Q\left(1 - \frac{1}{\tau}\right),$$

where $Q(\alpha)$ is the α-quantile of the random variable B. We have that $q_T \leq u_T$.

This lemma is used in what follows to bound the number of times a Thompson sample $\theta_{i,t}$ is drawn from a high quantile of the Beta distribution by using a UCB-like bound u_T.

Sketch of the proof. To prove the bound provided in Theorem 8, we consider one phase at a time and we upper bound the number of rounds a suboptimal arm has been selected. Let us focus on the phase Φ_ϕ: we exclude the first τ rounds during which the SW-TS algorithm is using data about rewards coming from two different distributions. More precisely, they are drawn either from Be$(\mu_{i,\phi-1})$ or Be$(\mu_{i,\phi})$. The contribution of these rounds to the pseudo-regret is bounded by $\tau \Upsilon N^\alpha$, thanks to Assumption 1. After that, we analyse the remaining rounds Φ'_ϕ of phase Φ_ϕ: during these rounds the pseudo-regret increases when a suboptimal arm is pulled. The probability of this event is upper bounded by the summation of the probability of event E_1 that the optimal arm $a_{i^*_\phi}$ is under-estimated and the probability of event E_2 that the optimal arm $a_{i^*_\phi}$ is not under-estimated, but a sub-optimal arm a_i is played.

The event E_1 occurs when a sample from a Beta distribution is lower that a certain lower bound and can be transformed, by resorting to the Beta/Binomial trick [83], into the event that a Binomial is lower than a lower bound. After this transformation, we use the Hoeffding inequality [73] over a bounded martingale difference to bound the probability of event E_1.

Event E_2 probability is further divided into the one of the event E_{2a} of drawing a sample which is higher than the quantile $q_{T_{i,t,\tau}}$ in Lemma 1 and the one of the complementary event E_{2b}. The former event has low probability (less than $\frac{1}{\tau}$) and in the latter one we considered Lemma 1 with $\varepsilon = 2$ to consider the UCB1 bound $u_{T_{i,t,\tau}}$ instead of the quantile. By choosing a suitable number of pulls of the suboptimal arm and of the optimal one s.t. the estimated expected value is well concentrated around the real mean $\mu_{i,\phi}$, we bound again the probability of the event E_{2b} resorting to the Hoeffding inequality. The guarantee that the arms have been pulled a sufficient number of times is provided by a general result in [84, 37]. Summing the aforementioned probabilities over the arms and the phases concludes the proof. □

6.3.3 Finite-Time Analysis in the Smoothly Changing Setting

We provide a finite-time analysis of the pseudo-regret achieved under SW-TS algorithm, in the SC-MAB setting introduced in Section 6.2.2.

Theorem 9. *If policy SW-TS is run over a SC-MAB setting with $X_{i,t} \sim Be(\mu_{i,t})$, Lipschitz constant $\sigma > 0$ and there exists $\Delta_0 \in (0,1)$ as in Assumption 3, for any $\tau \in \mathbb{N}$ s.t. $2\sigma\tau < \Delta \leq 3\sigma\tau \leq \Delta_0$, the expected pseudo-regret after N rounds is at most:*

$$\begin{aligned}\bar{R}_N(\mathfrak{U}) \leq \left(3\sigma\mathcal{F}N^\beta + 1\right)\tau \\ + \frac{NK}{\tau}\left[\frac{56\log\tau}{(\Delta - 2\sigma\tau)^2} + \log\tau + 4 + \frac{22}{\log\tau} + \frac{1}{\tau^{\frac{1}{2}}}\right].\end{aligned}$$

Sketch of the proof. At first, we divide the rounds in $\Phi_{\Delta,N}$, in which the pseudo-regret is bounded trivially by $3\sigma\tau\mathcal{F}N^\beta$ thanks to Assumption 3, and the remaining ones $\Phi_{\Delta^C,N} := \{1,\dots,N\} \setminus \Phi_{\Delta,N}$, in which the absolute difference in terms of expected reward between pair of arms is greater than $\Delta > 2\sigma\tau$. Over $\Phi_{\Delta^C,N}$ we define an arm $\underline{a}_{i_t^*}$ with reward $\widetilde{X}_{i^*,s} := X_{i^*,s} + \mu_{i^*,t} - \mu_{i^*,s} - \sigma\tau$, i.e., a pessimistic version of the optimal arm $a_{i_t^*}$. Assumption 2 guarantees that this newly defined reward is optimal in the last τ rounds and has constant expected value equal to $\mu_{i_t^*,t}$, allowing us to use a reasoning similar to what has been done for Theorem 8 to bound the pseudo-regret. Summing the two pseudo-regret contributions and over the arms concludes the proof. □

The dependence of the pseudo-regret to the factor $\frac{N}{\tau}$ is similar to what has been obtained in [37, 52], where frequentist algorithms have been considered. In the case Assumption 3 holds with $\beta = 0$, an order optimal choice of the sliding window is $\tau \propto \sqrt{N}$ which provides a pseudo-regret $\bar{R}_N(\mathfrak{U})$ of order $\tilde{O}(\sqrt{N})$. In the case Assumption 3 hold for $\beta > 0$, considerations similar to what has been discussed in the AC-MAB setting can be used to derive the order optimal sliding window τ and the corresponding upper bound over the pseudo-regret.

6.3.4 Finite-Time Analysis in the Abruptly and Smoothly Changing Setting

Once we proved theoretical guarantees of SW-TS in both the AC-MAB and SC-MAB settings, it is quite straightforward to show that:

Theorem 10. *If policy SW-TS is run over an ASC-MAB setting with $X_{i,t} \sim Be(\mu_{i,t})$, Lipschitz constant $\sigma > 0$ as in Assumption 4 and there exists $\Delta_0 \in (0,1)$ as in Assumption 3, for any $\tau \in \mathbb{N}$ s.t. $2\sigma\tau < \Delta \leq 3\sigma\tau \leq \Delta_0$, the expected pseudo-regret after N rounds is at most:*

$$\bar{R}_N(\mathfrak{U}) \leq \left(3\sigma\mathcal{F}N^{\beta} + \Upsilon N^{\alpha}\right)\tau \\ + \frac{NK}{\tau}\left[\frac{56\log\tau}{(\Delta - 2\sigma\tau)^2} + \log\tau + 4 + \frac{22}{\log\tau} + \frac{1}{\tau^{\frac{1}{2}}}\right],$$

where Υ and α are defined in Assumption 1.

As pointed out in the other settings, in the case $\alpha = 0$ and $\beta = 0$ and by setting a window of $\tau = \sqrt{N}$, we would have an upper bound over the regret of order $\tilde{O}(\sqrt{N})$. The asymptotic order of SW-TS in the ASC-MAB setting upper bound reduces to the one of Theorem 9 in the case we have $\Upsilon = 0$, i.e., we are in a SC-MAB setting. If we apply the bound in Theorem 10 for the AC-MAB setting, by fixing $\Delta = \min_i \Delta_i$ we have $|\Phi_{\Delta,N}| = 0$, thus $\mathcal{F} = 0$, and we obtain a slightly less accurate bound, which presents the same order in terms of N and τ of the one in Theorem 8.

6.4 Experimental Evaluation of SW-TS

We experimentally evaluate our algorithm w.r.t. the state-of-the-art algorithms with theoretical guarantees in terms of pseudo-regret performance in the AC-MAB and SC-MAB settings. In particular, we compare SW-TS with Thompson Sampling (TS) [9] to evaluate the improvement obtained thanks to the employment of a sliding window τ. Furthermore, we compare SW-TS with REXP3 [51], SW-UCB [52], SW-KL-UCB [37] and SER4 [57] to evaluate the improvement obtained thanks to the adoption of Bayesian methods vs. frequentist ones in non-stationary settings. The figures of merit we consider is the pseudo-regret $\bar{R}_N(\mathfrak{U})$, as defined in Equation (6.3), and the corresponding 95% confidence intervals. Here we report only the most significant results, provided that experiments on the other tested configurations do not change the final conclusions.

6.4.1 Abruptly Changing MAB Setting

Experimental Setting We consider a time horizon $N \in \{10^4, 10^5, 10^6\}$ and a number of arms $K \in \{5, 10, 20, 30\}$. We split the time horizon N into four phases of equal length. During each phase, we select randomly the expected value $\mu_{i,\phi}$ for each arm i. After each breakpoint, we randomly change the expected value $\mu_{i,\phi}$ of each arm a_i, making sure that there is never the

Table 6.1: *AC-MAB: Results concerning* $\bar{R}_N(\mathfrak{U})$.

			N		
			10^4	10^5	10^6
K	5	TS	1317±52.89	12857±425.68	114476±4836.98
		SER4	2494±37.63	25601±513.99	238034±4323.34
		REXP3	1451±13.70	8448±55.21	42561±212.75
		SW-UCB	824±66.80	5687±814.94	32939±7587.28
		SW-KL-UCB	**344±7.57**	1570±31.51	6248±145.51
		SW-TS	437±13.37	**1467±30.45**	**4904±39.00**
	10	TS	1251±26.90	10927±315.30	98312±4168.24
		SER4	3151±34.63	31454±499.91	279232±6504.65
		REXP3	1913±17.85	12170±108.38	61978±345.25
		SW-UCB	1116±68.46	8143±872.37	49537±6191.14
		SW-KL-UCB	**469±7.98**	2197±45.54	8601±162.32
		SW-TS	**470±8.82**	**1632±32.85**	**5493±92.67**
	20	TS	1130±30.91	8864±139.77	69919±2447.98
		SER4	3684±26.76	33293±167.89	293844±3038.42
		REXP3	2480±17.27	16134±93.65	83042±337.96
		SW-UCB	1405±57.44	11789±503.34	68751±6651.74
		SW-KL-UCB	652±6.70	3086±48.22	11921±315.74
		SW-TS	**536±10.26**	**1858±26.82**	**6156±149.64**
	30	TS	1016±35.55	7714±170.92	61979±2001.15
		SER4	3922±19.23	33622±212.29	285382±1727.97
		REXP3	2712±22.37	18432±100.09	96851±378.67
		SW-UCB	1566±60.42	12271±804.93	82006±8424.70
		SW-KL-UCB	770±19.79	3858±84.94	15287±233.75
		SW-TS	**575±12.20**	**2067±35.65**	**7123±96.46**

same optimal arm in two different phases, i.e., $a_{i^*_\phi} \neq a_{i^*_{\phi'}}, \forall \phi, \phi'$ with $\phi \neq \phi'$. For sake of comparison, we choose a sliding window $\tau = 4\sqrt{N \log(N)}$ as in [52]. We generate 10 configurations for each combination of N and K as described above and we provide the results averaged over the configurations and over 100 independent trials for each of them.

Results The numerical results in terms of $\bar{R}_N(\mathfrak{U})$ are reported in Table 6.1. For each combination of N and K, we highlight in bold the smallest value of $\bar{R}_N(\mathfrak{U})$ achieved. SW-TS outperforms the other algorithms in all the configurations except for the setting with $N = 10^4$ and $K = 5$ where SW-KL-UCB outperforms SW-TS. In the setting with $N = 10^4$ and $K = 10$ there is no statistical evidence to determine which algorithm is the best between SW-TS and SW-KL-UCB, since the 95% confidence intervals overlap. In Figure 6.1, we report the results for settings with $K = 10$ as t varies. It can be observed

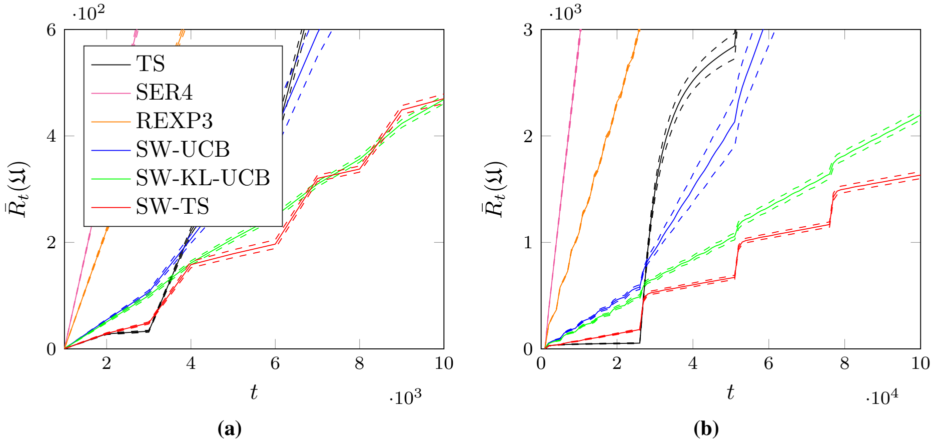


Figure 6.1: *AC-MAB: Results as t varies for the pseudo-regret* $\bar{R}_t(\mathfrak{U})$ *with* $K = 10$*, in the settings with* $N = 10^4$ *(a) and with* $N = 10^5$ *(b).*

that with $N = 10^4$ (Figure 6.1a), the performance of SW-TS and SW-KL-UCB are similar. However, the regret obtained by the algorithms is almost linear, suggesting that the algorithms are not able to learn since the problem is excessively hard. With a longer time horizon of $N = 10^5$ (Figure 6.1b), the sliding window τ becomes larger (we recall that we use a τ depending on N) as well as the phases length and thus SW-TS outperforms SW-KL-UCB. The SW-TS suffers from a larger regret when we enter a new phase, e.g., around $t = 5 \cdot 10^4$, but once the sliding window discards the samples coming from the previous phase, SW-TS is able to learn faster than other algorithms, which is exemplified by the lower slope of the regret between $t = 6 \cdot 10^4$ and $t = 7 \cdot 10^4$.

6.4.2 Smoothly Changing MAB Setting

Experimental Setting We consider a time horizon $N \in \{10^4, 10^5, 10^6\}$ and a number of arms $K \in \{5, 10, 20, 30\}$. We consider the experimental settings of [37], where the expected value $\mu_{i,t}$ of arm a_i changes according to the following function:

$$\mu_{i,t} = \frac{K-1}{K} - \frac{|w(t) - i|}{K}, \quad w(t) = 1 + \frac{(K-1)(1+\sin(t\sigma))}{2}.$$

We used a sliding window $\tau = \sqrt{N}$ and, in order to satisfy the assumption on the value of Δ in Theorem 9 for all values of N, we choose $\sigma = 0.0001$.[9]

[9] A discussion on the conditions for which the analysed SC-MAB setting satisfies Assumption 3 is provided in Appendix A.2.6.

Table 6.2: *SC-MAB: Results concerning* $\bar{R}_N(\mathfrak{U})$.

			N		
			10^4	10^5	10^6
K	5	TS	**218±41.94**	11995±562.73	161933±3767.54
		SER4	1787±6.61	22398±61.49	212095±309.93
		REXP3	957±16.58	11141±58.95	111403±169.79
		SW-UCB	1624±62.40	6560±49.31	34615±160.23
		SW-KLUCB	987±13.79	7407±50.99	40190±165.42
		SW-TS	608±15.43	**3330±40.60**	**16403±117.46**
	10	TS	**520±42.19**	13253±579.05	169850±4434.77
		SER4	2206±14.66	26094±145.80	242464±498.22
		REXP3	1253±19.36	14391±63.09	144581±199.42
		SW-UCB	3424±105.37	36622±314.36	80256±4375.85
		SW-KLUCB	1289±12.56	11009±50.77	64518±179.97
		SW-TS	922±16.18	**5529±49.82**	**28258±149.82**
	20	TS	**549±26.74**	12843±390.73	173140±2772.27
		SER4	2361±32.85	27286±259.41	258380±1039.26
		REXP3	1470±16.91	17334±67.77	174065±219.54
		SW-UCB	4466±202.74	45089±353.93	448649±155.59
		SW-KLUCB	1330±12.17	13630±38.60	89442±157.88
		SW-TS	1180±15.48	**7971±49.70**	**44186±154.32**
	30	TS	**581±26.29**	12483±297.63	172305±2205.39
		SER4	2480±23.17	27872±354.54	279190±1680.75
		REXP3	1607±14.59	18854±59.28	189734±178.99
		SW-UCB	4348±329.65	47586±911.96	462611±443.20
		SW-KLUCB	1638±11.92	14603±37.31	102707±126.91
		SW-TS	1339±11.49	**9595±39.76**	**54298±124.20**

We average the results over 100 independent trials for each combination of N, K and σ.

Results The numerical results in terms of $\bar{R}_N(\mathfrak{U})$ are reported in Table 6.2. It can be observed that SW-TS outperforms all the other algorithms except for the case with $N = 10^4$: SW-TS achieves the best performance w.r.t. the other sliding window algorithms, but it is not able to outperform TS. The reason behind this behaviour lies in the fact that with such a small value of σ the optimal arm remains the same until round $t = 5 \cdot 10^3$ and it is not convenient to use a sliding window approach. Conversely, if we have longer time horizons, the optimal arm changes more often: with $N = 10^5$ we have 14 changes of the optimal arm and the performance of TS becomes the worst. In Figure 6.2a, we report the results as t varies in the case with $N = 10^4$. It can be observed that, when the optimal arm changes, there is a worsening in the regret performance of TS. However, no sliding window algorithm can

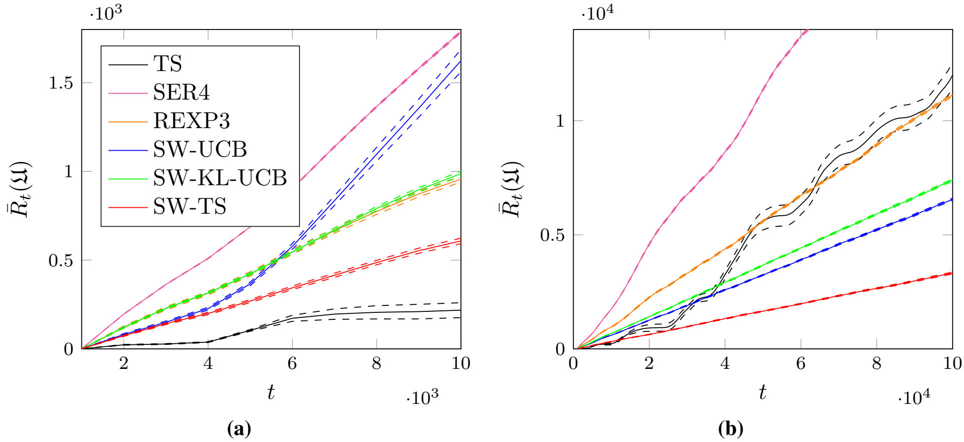


Figure 6.2: *SC-MAB: Results as t varies for the pseudo-regret $\bar{R}_t(\mathfrak{U})$ with $K = 5$, with $N = 10^4$ (a) and $N = 10^5$ (b).*

reach its performance. In the case with $N = 10^5$, reported in Figure 6.2b, TS is not able to limit its regret. Even if in the very first rounds TS and SW-TS share similar behaviours, the use of a sliding window assures the best performance to SW-TS.

6.4.3 Abruptly and Smoothly Changing MAB Setting

Experimental Setting We consider a time horizon $N \in \{10^4, 10^5, 10^6\}$ and a number of arms $K \in \{5, 10, 20, 30\}$. For each setting, we split the time horizon N into four phases of equal duration. During each phase, the expected value $\mu_{i,t}$ of arm a_i changes according to the following function:

$$\mu_{i,t} = \frac{K-1}{K} - \frac{|w(t) - i|}{K},$$
$$w(t) = 1 + \frac{(K-1)(1 + \sin(t\sigma))}{2}.$$

After each breakpoint, in order to abruptly change the optimal arm, we shift the $\sin(t\sigma)$ in the expected value $\mu_{i,\phi}$ of each arm i of an amount of rounds proportional to the time horizon N: after the first breakpoint, we shift of an amount of rounds of 25% of N; after the second one, 50% of N; after the third one, 75% of N. At first, we consider a sliding window $\tau = \sqrt{N}$ and, in order to satisfy, for all values of N, the assumption on the value of Δ in Theorem 10, we choose $\sigma = 0.0001$. After that, we choose a sliding window $\tau = \sigma^{-\frac{4}{5}}$ and we set $\sigma \in \{0.001, 0.002, \ldots, 0.01\}$, as in [37]. In both cases,

Table 6.3: *ASC-MAB with $\tau = \sigma^{-\frac{4}{5}}$: Results concerning $\bar{R}_N(\mathfrak{U})$ in the settings with $\sigma = 10^{-3}$.*

			N		
			10^4	10^5	10^6
K	5	TS	1252±39.02	23163±305.47	211871±380.91
		SER4	2253±9.44	23133±55.87	226316±294.62
		REXP3	1661±14.75	17517±42.07	175080±115.16
		SW-UCB	684±15.74	7451±74.99	90968±7174.24
		SW-KLUCB	754±9.62	8232±50.02	83436±359.15
		SW-TS	**470±11.90**	**4645±54.85**	**47197±298.63**
	10	TS	1370±39.51	26078±384.97	266289±283.49
		SER4	2745±23.31	29277±99.78	299947±456.21
		REXP3	2063±12.45	21719±49.39	217066±140.36
		SW-UCB	4736±62.98	41371±76.94	423199±99.35
		SW-KLUCB	1090±11.15	11958±48.21	120388±218.01
		SW-TS	**728±11.76**	**7630±47.36**	**77302±207.19**
	20	TS	1394±41.43	26573±341.85	292356±328.06
		SER4	3107±21.22	34016±46.53	341214±164.83
		REXP3	2389±14.57	25122±46.73	251113±143.90
		SW-UCB	5309±4.98	45305±77.67	459211±75.73
		SW-KLUCB	1345±9.73	14479±36.84	145437±133.78
		SW-TS	**980±9.92**	**10420±34.53**	**104146±107.82**
	30	TS	1401±27.51	26713±326.58	301882±317.67
		SER4	3252±13.63	35227±22.46	352542±78.03
		REXP3	2558±12.68	26911±40.75	268738±126.13
		SW-UCB	5513±4.21	46429±132.47	471276±0.00
		SW-KLUCB	1418±11.11	15161±37.74	151120±143.85
		SW-TS	**1136±10.84**	**12102±32.18**	**120625±115.00**

we average the results over 100 independent trials for each combination of N, K and σ.

Results The results for both settings are similar to the ones presented for the SC-MAB setting, suggesting that the abrupt changes do not affect the regret if the expected values of the arms are smoothly changing.

The numerical results in terms of $\bar{R}_N(\mathfrak{U})$ with $\tau = \sqrt{N}$ are reported in Table 6.4 for the experiments with $\sigma = 10^{-4}$. As in the SC-MAB setting, it can be observed that SW-TS outperforms all the other algorithms except for the case with $N = 10^4$, in which it is not able to outperform TS. The reason behind this behaviour lies again in the fact that with such a small value of σ the optimal arm remains the same until round $t = 5 \cdot 10^3$ and it is not convenient to use a sliding window approach. With longer time horizons, the optimal arm changes more often and the performance of TS becomes the

Table 6.4: *ASC-MAB with $\tau = \sqrt{N}$: Results concerning $\bar{R}_N(\mathfrak{U})$ in the settings with $\sigma = 10^{-4}$.*

			N		
			10^4	10^5	10^6
K	5	TS	**416±42.09**	11598±294.55	155795±3325.74
		SER4	2136±10.24	22894±63.04	211904±386.48
		REXP3	984±18.45	11428±55.26	111657±166.40
		SW-UCB	1424±80.93	6565±58.97	34832±162.74
		SW-KLUCB	991±16.66	7367±51.78	40395±171.51
		SW-TS	587±15.28	**3376±51.61**	**16546±143.17**
	10	TS	**513±33.02**	13322±399.70	166233±4166.45
		SER4	2677±28.61	26590±163.43	243132±652.63
		REXP3	1391±20.59	14652±65.30	144851±231.14
		SW-UCB	3807±146.66	51669±13.16	82394±5545.59
		SW-KLUCB	1434±15.94	10994±42.74	64783±166.04
		SW-TS	967±15.41	**5512±47.74**	**28539±143.63**
	20	TS	**598±30.29**	13099±243.57	172319±3124.95
		SER4	2832±56.99	28280±242.18	258039±1055.98
		REXP3	1664±18.84	17838±62.91	173867±198.21
		SW-UCB	5085±229.91	56948±35.71	450598±200.35
		SW-KLUCB	1544±12.00	13718±40.59	89463±139.87
		SW-TS	1280±12.89	**8017±52.56**	**44306±115.80**
	30	TS	**589±21.37**	12854±288.97	171534±2879.85
		SER4	2929±48.64	29651±457.64	278093±1590.27
		REXP3	1833±18.58	19585±53.51	189882±189.03
		SW-UCB	4819±418.89	59078±38.20	464238±565.19
		SW-KLUCB	1882±14.16	14718±33.40	102637±138.04
		SW-TS	1471±13.06	**9612±41.48**	**54363±134.34**

worst.

The results in terms of $\bar{R}_N(\mathfrak{U})$ with $\tau = \sigma^{\frac{4}{5}}$ are reported in Table 6.3 for the experiments with $\sigma = 10^{-3}$. As in the SC-MAB setting, SW-TS outperforms all the other algorithms, providing in every setting the smallest value for $\bar{R}_N(\mathfrak{U})$.

6.5 Unimodal MAB: Problem Formulation

In the Unimodal MAB setting, a learner receives in input a finite undirected graph MAB setting $G = (A, E)$, whose vertices $A = \{a_1, \dots, a_K\}$ with $K \in \mathbb{N}$ correspond to the arms and an edge $(a_i a_j) \in E$ exists only if there is a direct partial order relationship between the expected rewards of arms a_i and a_j. The leaner knows *a priori* the nodes and the edges (i.e., she knows the graph), but, for each edge, she does not know *a priori* which is the node of the edge with the largest expected reward (i.e., she does not know the ordering relationship). At each round t over a time horizon of $N \in \mathbb{N}$ the learner selects an arm a_i and gains the corresponding reward $x_{i,t}$. This reward is drawn from an i.i.d. random variable $X_{i,t}$ (i.e., we consider a stochastic MAB setting) characterized by an unknown distribution $\mathcal{D}_i$ with finite known support $\Omega \subset \mathbb{R}$ (as customary in MAB settings, from now on we will consider $\Omega \subseteq [0, 1]$) and by unknown expected value $\mu_i := \mathbb{E}[X_{i,t}]$. We assume that there is a single optimal arm, i.e., there exists a unique arm a_{i^*} s.t. its expected value $\mu^* := \mu_{i^*} = \max_i \mu_i$ with $\mu^* \geq \mu_i$ for each $i \in \{1, \ldots, K\}$.

Here we analyze a graph bandit setting with unimodality property, defined as:

Definition 1. *A graph* unimodal MAB *(UMAB) setting $G = (A, E)$ is a graph bandit setting G s.t. for each sub-optimal arm $a_i, i \neq i^*$ it exists a finite path $p = (i_1 = i, \ldots, i_m = i^*)$ s.t. $\mu_{i_k} < \mu_{i_{k+1}}$ and $(a_{i_k}, a_{i_{k+1}}) \in E$ for each $k \in \{1, \ldots, m-1\}$.*

This definition assures that if one is able to identify a non-decreasing path in G of expected rewards, she will be able to reach the optimum arm, without getting stuck in local optima. We would like to point out that the unimodality property implies that the graph G is connected, thus we will consider only connected graphs from this point on.

A policy $\mathfrak{U}$ over a UMAB setting is a procedure able to select at each round t an arm a_{i_t} by basing on the history h_t, i.e., the sequence of past selected arms and past rewards gained. The pseudo-regret $\bar{R}_N(\mathfrak{U})$ of a generic policy $\mathfrak{U}$ over a UMAB setting is defined as:

$$\bar{R}_N(\mathfrak{U}) := N\mu^* - \mathbb{E}\left[\sum_{t=1}^{N} X_{i_t,t}\right], \tag{6.4}$$

where the expected value $\mathbb{E}[\cdot]$ is taken w.r.t. the stochasticity of the gained rewards $X_{i_t,t}$ and of the policy $\mathfrak{U}$.

Let us define the neighborhood of arm a_i as $\mathcal{N}(i) := \{j | (a_i a_j) \in E\}$, i.e., the set of each index j of the arm a_j connected with an edge $(a_i a_j) \in E$ to the arm a_i. It has been shown in [37] that the problem of learning in a UMAB setting presents a lower bound over the regret $\bar{R}_N(\mathfrak{U})$ of the following form:

Theorem 11. *Let $\mathfrak{U}$ be a uniformly good policy, i.e., a policy s.t. $\bar{R}_N(\mathfrak{U}) =$*

Algorithm 11: UTS

```
Input: UMAB setting $G = (V, E)$, Horizon $T$, Priors $\{\pi_i\}_{i=1}^{K}$
for $t \in \{1, \dots, T\}$ do
  Compute $\hat{\mu}_{i,T_{i,t}}$ for each $i \in \{1, \dots, K\}$
  Find the leader $a_{l(t)}$
  if $L_{l(t),t} \bmod |N^{+}(l(t))| = 0$ then
    Collect reward $x_{l(t),t}$
  else
    Draw $\theta_{i,t}$ from $\pi_{i,t}$ for each $i \in N^{+}(l(t))$
    Collect reward $x_{i_t,t}$ where $i_t = \arg\max_i \theta_{i,t}$
```

$o(N^c)$ for each $c > 0$. Given a UMAB setting $G = (A, E)$ we have:

$$\liminf_{N \to \infty} \frac{\bar{R}_N(\mathfrak{U})}{\log(N)} = \sum_{i \in \mathcal{N}(i^*)} \frac{\mu^* - \mu_i}{KL(\mu_i, \mu^*)} \tag{6.5}$$

where $KL(p, q) = p \log\left(\frac{p}{q}\right) + (1-p) \log\left(\frac{1-p}{1-q}\right)$, i.e., the Kullaback-Leibler divergence of two Bernoulli distributions with means p and q, respectively.

This result is similar to the one provided in [8], with the only difference that the summation is restricted to the arms laying in the neighborhood of the optimal arm $\mathcal{N}(i^*)$ and reduces to it when the optimal arm is connected to all the others (i.e., $\mathcal{N}(i^*) \equiv \{1, \dots, K\}$) or the graph is completely connected (i.e., $\mathcal{N}(i) \equiv \{1, \dots, K\}, \forall i$). We would like to point out that by relying on the assumption of having a single maximum of the expected rewards, we also assure that the optimal arm neighborhood $\mathcal{N}(i^*)$ is uniquely defined and, thus, the lower bound inequality in Equation (6.5) is well defined.

6.6 The Unimodal Thompson Sampling Algorithm

We describe the UTS algorithm and we show that its regret is asymptotically optimal, i.e., it asymptotically matches the lower bound of Theorem 11. The algorithm is an extension of the Thompson Sampling [9] that exploits the graph structure and the unimodal property of the UMAB setting. Basically, the rationale of the algorithm is to apply a simple variation of the TS algorithm to only the arms associated with the nodes that compose the neighborhood of the arm with the highest empirical mean reward, called *leader*.

6.6.1 The UTS Pseudo-code

The pseudo-code of the UTS algorithm is presented in Algorithm 11. The algorithm receives in input the graph structure G, the time horizon N, and a Bayesian prior π_i for each expected reward μ_i. At each round t, the algorithm computes the empirical expected reward for each arm (Line 3):

$$\hat{\mu}_{i,t} := \begin{cases} \dfrac{S_{i_t}}{T_{i,t}} & \text{if } T_{i,t} > 0 \\ 0 & \text{otherwise} \end{cases},$$

where $S_{i,t} = \sum_{h=1}^{t-1} X_{i,h} \mathbf{1}\{\mathfrak{U}(h) = a_i\}$ is the cumulative reward of arm a_i up to round t and $T_{i,t} = \sum_{h=1}^{t-1} \mathbf{1}\{\mathfrak{U}(h) = a_i\}$ is the number of times the arm a_i has been pulled up to round t.[10] After that, UTS selects the arm denoted as the leader $a_{l(t)}$ for round t, i.e., the one having the maximum empirical expected reward:

$$a_{l(t)} = \arg\max_{a_i \in A} \hat{\mu}_{i,t}. \tag{6.6}$$

Once the leader has been chosen, we restrict the selection procedure to it and its neighborhood, considering only arms with indexes in $\mathcal{N}^+(l(t)) := \mathcal{N}(l(t)) \cup \{l(t)\}$. Denote with $L_{i,t} := \sum_{h=1}^{t-1} \mathbf{1}\{l(h) = i\}$ the number of times the arm a_i has been selected as leader before round t. If $L_{l(t),t}$ is a multiple of $|\mathcal{N}^+(l(t))|$, then the leader is pulled and reward $x_{l(t),t}$ is gained (Line 6).[11] Otherwise, the TS algorithm is performed over arms a_i s.t. $i \in \mathcal{N}^+(l(t))$ (Lines 8-9).

Basically, under the assumption of having a prior π_i, we can compute the posterior distribution $\pi_{i,t}$ for μ_i after t rounds, using the information gathered from the rounds in which a_i has been pulled. We denote with $\theta_{i,t}$ a sample drawn from $\pi_{i,t}$, called Thompson sample. For instance, for Bernoulli rewards and by assuming uniform priors we have that $\pi_{i,t} = Beta(1 + S_{i,t}, 1 + T_{i,t} - S_{i,t})$, where $Beta(\alpha, \beta)$ is the beta distribution with parameters α and β. Finally, the UTS algorithm pulls the arm with the largest Thompson sample $\theta_{i,n}$ and collects the corresponding reward $x_{i_t,t}$. See [10] for further details.

Remark 1. Assuming that the UTS algorithm receives in input the whole graph G is unnecessary. The algorithm just requires an oracle that, at each round t, is able to return the neighborhood $\mathcal{N}(l(t))$ of the arm which is currently the leader $a_{l(t)}$. This is crucial in all the applications in which the graph is discovered by means of a series of queries and the queries have a

[10] We here denote with $\mathbf{1}\{\cdot\}$ the indicator function.

[11] We here denote with $|\cdot|$ the cardinality operator.

non-negligible cost (e.g., in social networks a query might be computationally costly). Finally, we remark that the frequentist counterpart of our algorithm (i.e., the OSUB algorithm) requires the computation of the maximum node degree $\gamma := \max_i |\mathcal{N}(i)|$, thus requiring at least an initial analysis of the entire graph G.

6.6.2 Finite-time analysis of UTS

Theorem 12. *Given a UMAB setting $G = (A, E)$, the expected pseudo-regret of the UTS algorithm satisfies, for every $\varepsilon > 0$:*

$$\bar{R}_N(\mathit{UTS}) \leq (1+\varepsilon) \sum_{i \in \mathcal{N}(i^*)} \frac{\mu^* - \mu_i}{KL(\mu_i, \mu^*)} [\log(N) + \log\log(N)] + \tilde{C},$$

where $\tilde{C} > 0$ is a constant depending on ε, the number of arms K and the expected rewards $\{\mu_1, \dots, \mu_K\}$.

Sketch of the proof. The complete version of the proof is reported in the appendices. At first, we remark that a straightforward application of the proof provided for OSUB is not possible in the case of UTS. Indeed, the use of frequentist upper bounds over the expected reward in OSUB implies that in finite time and with high probability the bounds are ordered as the expected values. Since we are using a Bayesian algorithm, we would require the same assurance over the Thompson samples $\theta_{i,t}$, but we do not have a direct bound over $\mathbb{P}(\theta_{i,t} > \theta_{i',t})$ where $a_{i'}$ is the optimal arm in the neighborhood $\mathcal{N}^+(i)$. This fact requires to follow a completely different strategy when we analyze the case in which the leader is not the optimal arm.

The regret of the UTS algorithm $\bar{R}_N(\text{UTS})$ can be divided in two parts: the one obtained during those rounds in which the optimal arm a^* is the leader, called $\mathcal{R}_1$, and the summation of the regrets in the rounds in which the leader is the arm $a_i \neq a^*$, called $\mathcal{R}_i$. $\mathcal{R}_1$ is obtained when i^* is the leader, thus, the UTS algorithms behaves like Thompson Sampling restricted to the optimal arm and its neighborhood $\mathcal{N}^+(i^*)$, and the regret upper bound is the one derived in [10] for the TS algorithm.

$\mathcal{R}_i$ is upper bounded by the expected number of rounds the arm a_i has been selected as leader $\mathbb{E}[L_{i,N}]$ over the horizon N. Let us consider $\hat{L}_{i,N}$ defined as the number of rounds spent with a_i as leader when restricting the problem to its neighborhood $\mathcal{N}^+(i)$. $\mathbb{E}[\hat{L}_{i,N}]$ is an upper bound over $\mathbb{E}[L_{i,N}]$, since there is nonzero probability that the UTS algorithm moves in another neighborhood. Since $i \neq i^*$ and the setting is unimodal, there exists an optimal arm $a_{i'}, i' \neq i$ among those in the neighborhood $\mathcal{N}(i)$ s.t. $\mu_{i'} = \max_{i | a_i \in \mathcal{N}(i)} \mu_i$ and $\hat{\mu}_{i,t} \geq \hat{\mu}_{i'}$. Thus:

$$
\begin{aligned}
\mathcal{R}_i \leq \mathbb{E}[\hat{L}_{i,N}] &= \sum_{t=1}^{N} \mathbb{E}\left[\mathbf{1}\{\hat{\mu}_{i,t} = \max_{a_j \in \mathcal{N}^+(i)} \hat{\mu}_{j,t}\}\right] \\
&= \sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i,t} \geq \max_{a_j \in \mathcal{N}^+(i)} \hat{\mu}_{j,t}\right) \leq \sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i,t} \geq \hat{\mu}_{i',t}\right) \\
&= \sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i,t} - \mu_i - \frac{\Delta_i}{2} - \hat{\mu}_{i',t} + \mu_{i'} - \frac{\Delta_i}{2} \geq 0\right) \\
&\leq \underbrace{\sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i,t} - \mu_i - \frac{\Delta_i}{2} \geq 0\right)}_{\mathcal{R}_{i1}} + \underbrace{\sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i',t} - \mu_{i'} + \frac{\Delta_i}{2} \leq 0\right)}_{\mathcal{R}_{i2}},
\end{aligned}
$$

where $\Delta_i = \max_{i' | a_i \in \mathcal{N}(i)} \mu_{i'} - \mu_i$ is the expected loss incurred in choosing a_i instead of its best adjacent one $a_{i'}$.

$\mathcal{R}_{i1}$ can be upper bounded by a constant by relying on conditional probability definition and the Hoeffding inequality [73]. Specifically, we rely on the fact that the leader is chosen at least $\left\lfloor \frac{L_{l(t),t}}{|\mathcal{N}^+(l(t))|} \right\rfloor$ times. Upper bounding $\mathcal{R}_{i2}$ by a constant term requires the use of Proposition 1 in [10], which limits the expected number of times the optimal arm is pulled less than t^b times by TS, where $b \in (0, 1)$ is a constant, and the use of a technique already used on $\mathcal{R}_{i1}$. Summing up the regret over $i \neq i^*$ and considering the three obtained bounds concludes the proof. □

6.7 Experimental Evaluation of UTS

In this section, we compare the empirical performance of the proposed algorithm UTS with the performance of a number of algorithms. We study the performance of the state-of-the-art algorithm OSUB [37] to evaluate the improvement due to the employment of Bayesian approaches w.r.t. frequentist approaches. Furthermore, we study the performance of TS [9] to evaluate the improvement in Bayesian approaches due to the exploitation of the problem structure. For completeness, we study also the performance of KLUCB [85], being a frequentist algorithm that is optimal for Bernoulli distributions.

6.7.1 Figures of merit

Given a policy $\mathfrak{U}$, we evaluate the average and 95%-confidence intervals of the following figures of merit:

- the pseudo-regret $\bar{R}_N(\mathfrak{U})$ as defined in Equation (6.4); the lower $\bar{R}_N(\mathfrak{U})$ the better the performance;
- the regret ratio $R_{\%}(\mathfrak{U}_1, \mathfrak{U}_2) = \frac{\bar{R}_N(\mathfrak{U}_1)}{\bar{R}_N(\mathfrak{U}_2)}$ showing the ratio between the total regret of policy $\mathfrak{U}_1$ after N rounds and the one obtained with $\mathfrak{U}_2$; the lower $R_{\%}(\mathfrak{U}_1, \mathfrak{U}_2)$ the larger the relative improvement of $\mathfrak{U}_1$ w.r.t. $\mathfrak{U}_2$.

6.7.2 Line graphs

We initially consider the same experimental settings are in [37], composed of line graphs. They consider graphs with $K \in \{17, 129\}$ arms, where the arms are ordered on a line from the arm with smallest index to the arm with the largest index and with Bernoulli rewards whose averages have a triangular shape with the maximum on the arm in the middle of the line. More precisely, the minimum average is 0.1, associated with arms a_1 and a_{17} when $K = 17$ and with arms a_1 and a_{129} with $K = 129$, while the maximum average reward is $\mu^* = 0.9$, associated with arm a_9 when $K = 17$ and with arm a_{65} with $K = 129$. The averages decrease linearly from the maximum one to the minimum one.

For both the experiments, we average the regret over 100 independent trials of length $N = 10^5$. We report $R_t(\mathfrak{U})$ for each policy $\mathfrak{U}$ as t varies in Figure 6.3a, for $K = 17$, and in Figure 6.3b, for $K = 129$. The UTS algorithm outperforms all the other algorithms along the whole time horizon, providing a significant improvement in terms of regret w.r.t. the state-of-the-art algorithms. In order to have a more precise evaluation of the reduction of the regret w.r.t. OSUB algorithm, we report $R_{\%}(\mathfrak{U}, \text{OSUB})$ in Table 6.5. As also confirmed below by a more exhaustive series of experiments, in line graphs the relative improvement of performance due to UTS w.r.t. OSUB reduces as the number of arms increases, while the relative improvement of performance due to UTS w.r.t. TS increases as the number of arms increases.

6.7.3 Erdős-Rényi graphs

To provide a thorough experimental evaluation of the considered algorithms in settings in which the space of arms has a graph structure, we generate graphs using the model proposed by Erdős and Rényi [86], which allows us

Table 6.5: *Results concerning $R_{\%}(\mathfrak{U}, OSUB)$ in the setting with $K = 17$ and $K = 129$ and a line graph.*

	K	
	17	129
KLUCB	3.08 ± 0.05	6.51 ± 0.07
TS	1.34 ± 0.07	2.68 ± 0.05
UTS	$\mathbf{0.52 \pm 0.07}$	$\mathbf{0.76 \pm 0.15}$

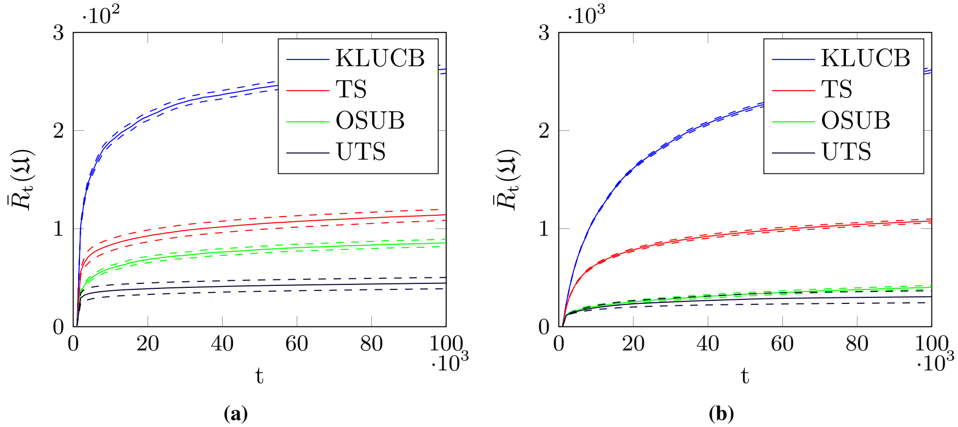


Figure 6.3: *Results for the pseudo-regret $\bar{R}_t(\mathfrak{U})$ in line graphs settings with $K = 17$ (a) and $K = 129$ (b) as defined in [37].*

to simulate graph structures more complex than a simple line. An Erdős-Rényi graph is generated by connecting nodes randomly: each edge is included in the graph with probability p, independently from existing edges. We consider connected graphs with $K \in \{5, 10, 20, 50, 100, 1000\}$ and with probability $p \in \{1, \frac{1}{2}, \frac{\log(K)}{K}, \ell\}$, where $p = 1$ corresponds to have a fully connected graph and therefore the graph structure is useless, $p = \frac{1}{2}$ corresponds to have a number of edges that increases linearly in the number of nodes, $p = \frac{\log(K)}{K}$ corresponds to have a few edges w.r.t. the nodes, and we use $p = \ell$ to denote line graphs (these line graphs differ from those used for the experimental evaluation discussed above for the reward function, as discussed in what follows). We use different values of p in order to see how the performance of UTS changes w.r.t. the number of edges in the graph; we remark that such an analysis is unexplored in the literature so far. The optimal arm is chosen randomly among the existing arms and its reward is given by a Bernoulli distribution with expected value 0.9. The rewards of the suboptimal arms are given by Bernoulli distributions with expected value depending

Table 6.6: *Results concerning $\bar{R}_N(\mathfrak{U})$ ($N = 10^5$) in the setting with Erdős-Rényi graphs.*

			p			
			1	$1/2$	$\log(K)/K$	ℓ
K	5	KLUCB	34 ± 0.4	50 ± 1.5	52 ± 3.7	56 ± 2.2
		TS	18 ± 0.2	23 ± 0.6	24 ± 1.3	25 ± 0.7
		OSUB	34 ± 0.3	32 ± 7.2	35 ± 5.8	31 ± 4.1
		UTS	$\mathbf{17 \pm 0.1}$	$\mathbf{15 \pm 2.4}$	$\mathbf{16 \pm 2.2}$	$\mathbf{14 \pm 1.3}$
	10	KLUCB	77 ± 0.5	107 ± 5.5	127 ± 11.2	159 ± 7.0
		TS	40 ± 0.2	50 ± 2.0	56 ± 3.8	67 ± 2.5
		OSUB	77 ± 0.3	76 ± 8.1	57 ± 5.6	70 ± 8.1
		UTS	$\mathbf{39 \pm 0.2}$	$\mathbf{35 \pm 3.2}$	$\mathbf{27 \pm 2.1}$	$\mathbf{34 \pm 2.4}$
	20	KLUCB	163 ± 0.7	217 ± 6.2	262 ± 16.2	386 ± 21.3
		TS	84 ± 0.5	102 ± 2.3	117 ± 5.7	157 ± 6.9
		OSUB	163 ± 0.8	148 ± 14.9	86 ± 14.6	124 ± 11.7
		UTS	$\mathbf{83 \pm 0.3}$	$\mathbf{70 \pm 6.0}$	$\mathbf{44 \pm 4.8}$	$\mathbf{65 \pm 8.8}$
	50	KLUCB	420 ± 0.7	560 ± 15.0	686 ± 30.5	1132 ± 49.2
		TS	217 ± 0.5	262 ± 4.4	303 ± 10.0	454 ± 19.9
		OSUB	420 ± 1.0	382 ± 35.6	162 ± 13.9	240 ± 15.8
		UTS	$\mathbf{216 \pm 0.7}$	$\mathbf{182 \pm 14.2}$	$\mathbf{89 \pm 5.5}$	$\mathbf{156 \pm 30.1}$
	100	KLUCB	846 ± 2.0	1134 ± 17.8	1313 ± 59.7	2327 ± 63.5
		TS	$\mathbf{436 \pm 1.1}$	528 ± 4.9	586 ± 18.4	973 ± 31.8
		OSUB	846 ± 2.7	786 ± 39.0	226 ± 27.1	369 ± 10.7
		UTS	$\mathbf{437 \pm 0.5}$	$\mathbf{372 \pm 15.2}$	$\mathbf{141 \pm 9.1}$	$\mathbf{290 \pm 42.3}$
	1000	KLUCB	8505 ± 12.2	11247 ± 60.1	12024 ± 464.7	10640 ± 291.5
		TS	$\mathbf{4391 \pm 3.4}$	5262 ± 23.0	5478 ± 151.3	6554 ± 115.2
		OSUB	8493 ± 13.6	7761 ± 153.4	1151 ± 45.0	$\mathbf{1165 \pm 20.7}$
		UTS	$\mathbf{4388 \pm 5.2}$	$\mathbf{3718 \pm 62.9}$	$\mathbf{1000 \pm 14.2}$	$\mathbf{1165 \pm 41.8}$

on their distance from the optimal one. More precisely, let d_i^* be the shortest path from the i-th arm to the optimal arm and let $d_{\max}^* = \max_{i \in \{1,\dots,K\}} d_i^*$ be the maximum shortest path of the graph. The expected reward of the i-th arm is $\mu_i = 0.9 - d_i^* \frac{(0.9-0.1)}{d_{\max}^*}$, i.e., the arm with $d_{\max}^*$ has a value equal to 0.1 and the expected rewards of the arms along the path from it to the optimal arm are evenly spaced between 0.1 and 0.9. We generate 10 different graphs for each combination of K and p and we run 100 independent trials of length $N = 10^5$ for each graph. We average the regret over the results of the 10 graphs.

In Table 6.6, we report $\bar{R}_N(\mathfrak{U})$ for each combination of policy $\mathfrak{U}$, K, and p. It can be observed that the UTS algorithm outperforms all the other algorithms, providing in every case the smallest regret except for $K = 1000$ and $p = \ell$. Below we discuss how the relative performance of the algorithms vary as the values of the parameters K and p vary.

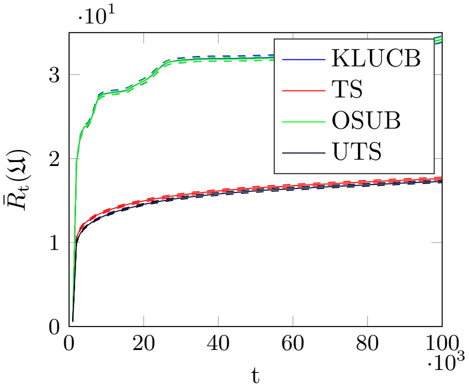


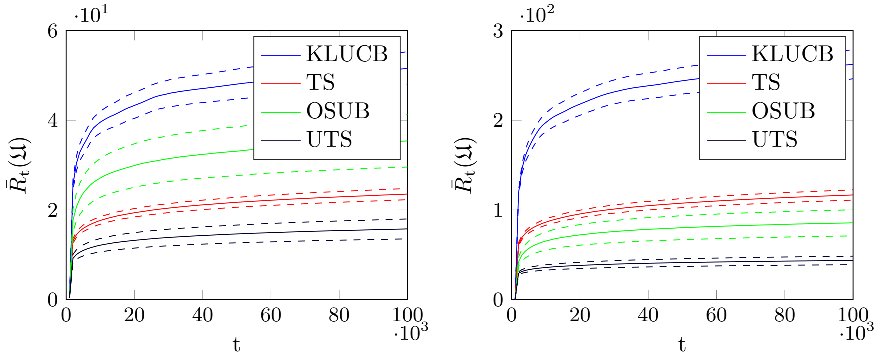
Figure 6.4: *Results for the pseudo-regret $\bar{R}_t(\mathfrak{U})$ in the setting with $K = 5$ and $p = 1$.*

Consider the case with $p = 1$. The performance of UTS and TS are approximately equal and the same holds for the performance of OSUB and KLUCB. This is due to the fact that the neighborhood of each node is composed by all the arms, the graphs being fully connected, and therefore UTS and OSUB cannot take any advantage from the structure of the problem. We notice, however, that UTS and TS have not the same behavior and that UTS always performs slightly better than TS. It can be observed in Figure 6.4 with $K = 5$ and $p = 1$ that the relative improvement is mainly at the beginning of the time horizon and that it goes to zero as K increases (the same holds for OSUB w.r.t. KLUCB). The reason behind this behavior is that UTS reduces the exploration performed by TS in the first rounds, forcing the algorithm to pull the leader (chosen as the arm maximizing the empirical mean) for a larger number of rounds.

Consider the case with $p = \frac{1}{2}$. In the considered experimental setting, the relative performance of the algorithms does not depend on K. The ordering, from the best to the worst, over the performance of the algorithms is: UTS, TS, OSUB, and finally KLUCB. Surprisingly, even the dependency of the following ratios on K is negligible: $R_{\%}(\text{UTS}, \text{TS}) = 0.68 \pm 0.03$, $R_{\%}(\text{UTS}, \text{OSUB}) = 0.47 \pm 0.01$, and $R_{\%}(\text{OSUB}, \text{KLUCB}) = 0.68 \pm 0.03$. This shows that the relative improvement due to UTS is constant w.r.t. TS and OSUB as K varies. These results raise the question whether the relative performance of OSUB and TS would be the same, except for the numerical values, for every p constant w.r.t. K. To answer to this question, we run additional experiments, considering the case in which $p = 0.1$, corresponding to the case in which the number of edges is linear in K, but it is smaller than the case with $p = \frac{1}{2}$. The results in terms of $\bar{R}_N(\mathfrak{U})$ show that OSUB outperforms TS for $K \geq 10$, suggesting that, when p is constant in K, OSUB may or may not outperform TS depending on the specific pair (p, K).

Consider the case with $p = \frac{\log(K)}{K}$. The ordering over the performance of the algorithms changes as K varies. More precisely, while UTS keeps to be the best algorithm for every K and KLUCB the worst algorithm for every K, the ordering between TS and OSUB changes. When $K \leq 10$ TS performs better than OSUB, instead when $K \geq 20$ OSUB outperforms TS, see Figure 6.5. This is due to the fact that, with a small number of arms, exploiting the graph structure is not sufficient for a frequentist algorithm to outperform the performance of TS, while with many arms exploiting the graph structure even with a frequentist algorithm is much better than employing a general-purpose Bayesian algorithm. The ratio $R_{\%}(\text{UTS}, \text{TS})$ monotonically decreases as K increases, from 0.66 when $K = 5$ to 0.19 when $K = 1000$, suggesting that exploiting the graph structure provides advantages as K increases. Instead, the ratio $R_{\%}(\text{UTS}, \text{OSUB})$ monotonically increases as K increases, from 0.45 when $K = 5$ to 0.94 when $K = 1000$, suggesting that the improvement provided by employing Bayesian approaches reduces as K increases as observed above in line graphs.

Consider the case with $p = \ell$. As in the case discussed above, OSUB is outperformed by TS for a small number of arms ($K \leq 10$), while it outperforms TS for many arms ($K \geq 20$). The reason is the same. Similarly, the ratio $R_{\%}(\text{UTS}, \text{TS})$ monotonically decreases as K increases, from 0.58 when $K = 5$ to 0.18 when $K = 1000$, and the ratio $R_{\%}(\text{UTS}, \text{OSUB})$ monotonically increases as K increases, from 0.45 when $K = 5$ to 1.00 when $K = 1000$. This confirms that the performance of UTS and the one of OSUB asymptotically match as K increases when $p = \ell$ (as well as $p = \frac{\log(K)}{K}$). In order to investigate the reasons behind such a behavior, we produce an additional experiment with the line graphs of Combes and Proutiere [37] except that the maximum expected reward is set to 0.108 when $K = 17$ and 0.165 when $K = 129$ (thus, given any edge with terminals i and $i + 1$, we have $|\mu_i - \mu_{i+1}| = 0.001$). What we observe is that, on average, OSUB outperforms UTS at $N = 10^5$ suggesting that, when it is necessary to repeatedly distinguish between three arms that have very similar expected rewards, frequentist methods may outperform the Bayesian ones. This is no longer true when N is much larger, e.g., $N = 10^7$, where UTS outperforms OSUB (interestingly, differently from what happens in the other topologies, in line graphs with very small $|\mu_i - \mu_{i+1}|$, the average $\bar{R}_N(\text{UTS})$ and $\bar{R}_N(\text{OSUB})$ cross a number of times during the time horizon). Furthermore, we evaluate how the relative performance of OSUB w.r.t. UTS varies for $|\mu_i - \mu_{i+1}| \in \{0.001, 0.002, 0.005\}$, observing it improves as $|\mu_i - \mu_{i+1}|$



(a)

(b)

Figure 6.5: *Results for the pseudo-regret* $\bar{R}_t(\mathfrak{U})$ *in the setting with* $K = 5$ *(a) and* $K = 20$ *(b) and* $p = \frac{\log(K)}{K}$.

decreases. Finally, we evaluate whether this behavior emerges also in Erdős-Rényi graphs in which $p = \frac{c}{K}$ where c is a constant (we use $p = \frac{5}{K}, \frac{10}{K}$) and we observe that UTS outperforms OSUB, suggesting that line graphs with very small $|\mu_i - \mu_{i+1}|$ are pathological instances for UTS.

CHAPTER 7

Conclusions and Future Works

In this thesis, we investigate the pricing problem in the setting of online sales of digital goods from the point of view of an e-commerce, such as an Online Travel Agency (OTA), selling its products in an environment where it is not possible to use information about the customers, like the metasearch environment. We study the problem of optimal pricing, that is the search for the optimal price to set on an item to maximize the expected profit. Metasearch scenario presents a large profitability, but also many characteristics which make the problem very challenging. In the specific, we have a huge catalog of items to price, we have no information about our customers, the environment is non-stationary, and we have really low conversion rates. Thus, the problem of building a user model and learning the optimal price becomes very hard and long in time. We tackle the problem of finding the best price as an online learning problem, with a particular focus on Multi-Armed Bandit (MAB) techniques. The solution we propose consists in dividing the problem into two sub-problems.

Clustering problem The first sub-problem is the one of clustering, that is the partitioning of the catalog in contexts of items sharing similar features. This goal can be achieved by learning from historical data collected directly with the system, recording the interactions with the customers. We propose a novel algorithm to deal with the problem of Learning with Logged Bandit Feedback (LLBF). The proposed methodology can learn a risk-averse policy to maximize the expected profit gained in this setting. It makes use of lower confidence bounds to build a decision tree over the context space, which provides both a decisional tool over future samples and an instrument to highlight the features that influence the profit the most. Indeed, our method provide a clear interpretability of the resulting model, useful for business analysis, that allows to easily identify the most relevant features for the definition of the contexts. With a wide experimental campaign, we present empirical evidence for the improved performance of our algorithm over the state-of-the-art and we show promising results on real-world datasets.

Optimization problem The second sub-problem is the study of algorithms to learn the optimal price that maximizes the expected profit of each context. Making use of Multi-Armed Bandit techniques, we exploit some properties of the pricing problem to improve the performance of the classical algorithms.

The first two properties are the decreasing monotonicity of the conversion rate on the price and the *a priori* information about the maximum conversion rate. We study how to exploit two properties of the pricing problem to improve the empiric performance of general-purpose bandit algorithms without losing their theoretical guarantees on the regret. We propose a methodology to apply to Upper Confidence Bound (UCB) policies, such as the well-known UCB1 and UCBV. We provide a wide experimental evaluation of our algorithms, comparing them with other frequentist MAB algorithms with theoretical guarantees that do not exploit the two aforementioned properties. In this way, we show the improvement obtained thanks to the exploitation of the problem characteristics.

Furthermore, we focus on non-stationary settings. We make use of a sliding window to tackle the non-stationarity of the environment. We propose algorithms both for the frequentist and the Bayesian case, we derive upper bounds on the regret of the proposed algorithms, and we show that our algorithms empirically outperform the state-of-the-art approaches in most of the considered configurations.

Finally, we study the Unimodal Multi-Armed Bandit (UMAB), characterized by a graph structure in which each arm corresponds to a node of a graph and each edge is associated with a relationship in terms of expected reward between its arms. We propose, to the best of our knowledge, the first Bayesian algorithm for the UMAB setting. We derive a tight upper bound that asymptotically matches the lower bound for the UMAB setting and we present a thorough experimental analysis showing that our algorithm outperforms the state-of-the-art methods.

Proposed solution The final solution consists of making the two sub-problems to work together. The feedback collected from the interaction between the users and the system, which implements bandits algorithms to price the items, is used as input for the clustering algorithm. The data collected with the newly generated model will be used to improve the clustering. Thus, the two sub-problems work in a cycle, continuously improving the performance of the system.

Future works A number of extensions are possible to this work. Future developments may study the exploitation of the pricing properties in continuous MAB setting, that is the case of a continuous decision space. Furthermore, we may study the finite-time lower bound of the regret and the gap-independent bounds of the proposed algorithms.

Future extensions for our clustering algorithm may concern settings with a continuous space of actions or a continuous space of contexts (without requiring to discretize it). Furthermore, another interesting future work is the use of metrics different from the expected profit to partition the context space suited for a subsequent learning process, for instance considering also non-stationary policies over the newly coming data.

We may also study a way to better integrate the two sub-problems into each other. For example, the clustering problem could be online and forms a significant part of the optimization as well.

Bibliography

[1] eMarketer. Worldwide retail and ecommerce sales: emarketer's estimates for 2016-2021. https://www.emarketer.com/Report/Worldwide-Retail-Ecommerce-Sales-eMarketers-Estimates-2 2002090. Accessed: August 2017.

[2] A. V. Den Boer. Dynamic pricing and learning: Historical origins, current research, and new directions. *Surveys in Operations Research and Management Science*, 20:1–18, 2015.

[3] C. Walsh and M. Gasdia. Influencing travelers in the new digital funnel. http://www.amadeus.com/documents/otas/wp-influ-trav-amadeus-web.pdf, 2015.

[4] A. Swaminathan and T. Joachims. Batch learning from logged bandit feedback through counterfactual risk minimization. *Journal of Machine Learning Research*, 16:1731–1755, 2015.

[5] A. Beygelzimer and J. Langford. The offset tree for learning with partial labels. In *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 129–138, 2009.

[6] P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2):235–256, 2002.

[7] H. Chernoff. A note on an inequality involving the normal distribution. *The Annals of Probability*, 9(3):533–535, 1981.

[8] T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Advances in Applied Mathematics*, 6(1):4–22, 1985.

[9] W. R. Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[10] E. Kaufmann, N. Korda, and R. Munos. Thompson sampling: An asymptotically optimal finite-time analysis. In *International Conference on Algorithmic Learning Theory (ALT)*, volume 7568 of *Lecture Notes in Computer Science*, pages 199–213. Springer, 2012.

[11] S. Agrawal and N. Goyal. Further optimal regret bounds for thompson sampling. *arXiv preprint arXiv:1209.3353*, 2012.

[12] Sébastien Bubeck, Nicolo Cesa-Bianchi, et al. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–122, 2012.

[13] R. Kleinberg and T. Leighton. The value of knowing a demand curve: Bounds on regret for online posted-price auctions. In *IEEE Symposium on Foundations of Computer Science*, pages 594–605. IEEE Computer Society, 2003.

[14] A. Mas-Colell, M. D. Whinston, J. R. Green, et al. *Microeconomic Theory*, volume 1. Oxford university press, 1995.

[15] N. Gatti, A. Lazaric, and F. Trovò. A truthful learning mechanism for contextual multi-slot sponsored search auctions with externalities. In *ACM Conference on Electronic Commerce*, pages 605–622, 2012.

[16] A. Piccolboni and C. Schindelhauer. Discrete prediction games with arbitrary feedback and loss. In *Conference on Learning Theory (COLT)*, pages 208–223. Springer, 2001.

[17] W. W. Moe and P. S. Fader. Dynamic conversion behavior at e-commerce sites. *Management Science*, 50(3):326–335, 2004.

[18] N. Alon, N. Cesa-Bianchi, C. Gentile, and Y. Mansour. From bandits to experts: A tale of domination and independence. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1610–1618, 2013.

[19] S. Mannor and O. Shamir. From bandits to experts: On the value of side-observations. In *Advances in Neural Information Processing Systems (NIPS)*, pages 684–692, 2011.

[20] G. Bartók, D. P. Foster, D. Pál, A. Rakhlin, and C. Szepesvári. Partial monitoring-classification, regret bounds, and algorithms. *Mathematics of Operations Research*, 39(4):967–997, 2014.

[21] N. Cesa-Bianchi, G. Lugosi, and G. Stoltz. Regret minimization under partial monitoring. *Mathematics of Operations Research*, 31(3):562–580, 2006.

[22] G. Bartók, D. Pál, and C. Szepesvári. Minimax regret of finite partial-monitoring games in stochastic environments. In *Conference on Learning Theory (COLT)*, volume 2011, pages 133–154. JMLR, 2011.

[23] G. Bartók, N. Zolghadr, and C. Szepesvári. An adaptive algorithm for finite stochastic partial monitoring. In *International Conference on Machine Learning (ICML)*, pages 1727–1734, 2012.

[24] D. P Foster and A. Rakhlin. No internal regret via neighborhood watch. In *Artificial Intelligence and Statistics (AISTATS)*, pages 382–390, 2012.

[25] A. Agarwal, P. Bartlett, and M. Dama. Optimal allocation strategies for the dark pool problem. In *Artificial Intelligence and Statistics (AISTATS)*, pages 9–16, 2010.

[26] J. D. Abernethy, K. Amin, and R. Zhu. Threshold bandits, with and without censored feedback. In *Advances in Neural Information Processing Systems (NIPS)*, pages 4889–4897, 2016.

[27] O. Besbes and A. Zeevi. Dynamic pricing without knowing the demand function: Risk bounds and near-optimal algorithms. *Mathematics of Operations Research*, 57(6):1407–1420, 2009.

[28] M. Chhabra and S. Das. Learning the demand curve in posted-price digital goods auctions. In *Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, pages 63–70, 2011.

[29] J. Broder and P. Rusmevichientong. Dynamic pricing under a general parametric choice model. *Mathematics of Operations Research*, 60(4):965–980, 2012.

[30] N. B. Keskin and A. Zeevi. Dynamic pricing with an unknown demand model: Asymptotically optimal semi-myopic policies. *Mathematics of Operations Research*, 62(5):1142–1167, 2014.

[31] O. Besbes and A. Zeevi. On the (surprising) sufficiency of linear models for dynamic pricing with demand learning. *Management Science*, 61(4):723–739, 2015.

[32] A. Blum, V. Kumar, A. Rudra, and F. Wu. Online learning in online auctions. *Theoretical Computer Science*, 324(2):137–146, 2004.

[33] P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM Journal on Computing*, 32(1):48–77, 2002.

[34] R. Cole and T. Roughgarden. The sample complexity of revenue maximization. In *ACM symposium on Theory of computing*, pages 243–252, 2014.

[35] J. H. Morgenstern and T. Roughgarden. On the pseudo-dimension of nearly optimal auctions. In *Advances in Neural Information Processing Systems (NIPS)*, pages 136–144, 2015.

[36] JY. Audibert, R. Munos, and C. Szepesvári. Exploration-exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science*, 410(19):1876–1902, 2009.

[37] R. Combes and A. Proutiere. Unimodal bandits: Regret lower bounds and optimal algorithms. In *International Conference on Machine Learning (ICML)*, pages 521–529, 2014.

[38] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, pages 415–444, 2001.

[39] D. Crandall, D. Cosley, D. Huttenlocher, J. Kleinberg, and S. Suri. Feedback effects between similarity and social influence in online communities. In *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 160–168. ACM, 2008.

[40] Y. Y. Jia and S. Mannor. Unimodal bandits. In *International Conference on Machine Learning (ICML)*, pages 41–48, 2011.

[41] B. Edelman and M. Ostrovsky. Strategic bidder behavior in sponsored search auctions. *Decision Support Systems*, 43(1):192–198, 2007.

[42] R. Combes and A. Proutiere. Unimodal bandits without smoothness. *arXiv preprint arXiv:1406.7447*, 2014.

[43] R. Kleinberg, A. Slivkins, and E. Upfal. Multi-armed bandits in metric spaces. In *ACM Symposium on Theory of Computing*, pages 681–690, 2008.

[44] M. Valko, R. Munos, B. Kveton, and T. Kocak. Spectral bandits for smooth graph functions. In *International Conference on Machine Learning (ICML)*, pages 46–54, 2014.

[45] S. Caron and S. Bhagat. Mixing bandits: A recipe for improved cold-start recommendations in a social network. In *ACM Workshop on Social Network Mining and Analysis*, page 11. ACM, 2013.

[46] J. Eliashberg and A. P. Jeuland. The impact of competitive entry in a developing market upon dynamic pricing strategies. *Marketing Science*, 5(1):20–36, 1986.

[47] M. E. Gorre, M. Mohammed, K. Ellwood, N. Hsu, R. Paquette, P. N. Rao, and C. L. Sawyers. Clinical resistance to sti-571 cancer therapy caused by bcr-abl gene mutation or amplification. *Science*, 293(5531):876–880, 2001.

[48] N. Gatti, A. Lazaric, M. Rocco, and F. Trovò. Truthful learning mechanisms for multi-slot sponsored search auctions with externalities. *Artificial Intelligence*, 227:93–139, 2015.

[49] B. Kitts and B. Leblanc. Optimal bidding on keyword auctions. *Electronic Markets*, 14(3):186–201, 2004.

[50] C. Hartland, S. Gelly, N. Baskiotis, O. Teytaud, and M. Sebag. Multi-armed bandit, dynamic environments and meta-bandits. Working Paper, Nov 2006.

[51] O. Besbes, Y. Gur, and A. Zeevi. Stochastic multi-armed bandit problem with non-stationary rewards. In *Advances in Neural Information Processing Systems (NIPS)*, pages 199–207, 2014.

[52] A. Garivier and E. Moulines. On upper-confidence bound policies for non-stationary bandit problems. *arXiv preprint arXiv:0805.3415*, 2008.

[53] L. Kocsis and C. Szepesvári. Discounted ucb. In *PASCAL Challenges Workshop*, pages 784–791, 2006.

[54] C. Y. Wei, Y. T. Hong, and C. J. Lu. Tracking the best expert in non-stationary stochastic environments. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3972–3980, 2016.

[55] O. C. Granmo and S. Berg. Solving non-stationary bandit problems by random sampling from sibling Kalman filters. In *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE)*, pages 199–208, 2010.

[56] J. Mellor and J. Shapiro. Thompson sampling in switching environments with Bayesian online change point detection. In *Artificial Intelligence and Statistics (AISTATS)*, pages 442–450, 2013.

[57] R. Allesiardo, R. Féraud, and O. A. Maillard. The non-stationary stochastic multi-armed bandit problem. *International Journal of Data Science and Analytics*, pages 1–17, 2017.

[58] A. Slivkins and E. Upfal. Adapting to a changing environment: the brownian restless bandits. In *Conference on Learning Theory (COLT)*, pages 343–354, 2008.

[59] V. Raj and S. Kalyani. Taming non-stationary bandits: A bayesian approach. *arXiv preprint arXiv:1707.09727*, 2017.

[60] D. L. St-Pierre and L. Jialin. Differential evolution algorithm applied to non-stationary bandit problem. In *IEEE Congress on Evolutionary Computation (CEC)*, pages 2397–2403, 2014.

[61] L. Tang, R. Rosales, A. Singh, and D. Agarwal. Automatic ad format selection via contextual bandits. In *ACM International Conference on Information and Knowledge Management (CIKM)*, pages 1587–1594, 2013.

[62] L. Li, W. Chu, J. Langford, and R. E. Schapire. A contextual-bandit approach to personalized news article recommendation. In *ACM International Conference on World Wide Web (WWW)*, pages 661–670, 2010.

[63] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer-Verlag, 2006.

[64] A. Swaminathan and T. Joachims. Counterfactual risk minimization: Learning from logged bandit feedback. In *International Conference on Machine Learning (ICML)*, pages 814–823, 2015.

[65] W. Chu, L. Li, L. Reyzin, and R. E. Schapire. Contextual bandits with linear payoff functions. In *Artificial Intelligence and Statistics (AISTATS)*, pages 208–214, 2011.

[66] R. Féraud, R. Allesiardo, T. Urvoy, and F. Clérot. Random forest for the contextual bandit problem. In *Artificial Intelligence and Statistics (AISTATS)*, pages 93–101, 2016.

[67] A. Sani, A. Lazaric, and R. Munos. Risk-aversion in multi-armed bandits. In *Advances in Neural Information Processing Systems (NIPS)*, pages 3275–3283, 2012.

[68] L. Rokach and O. Maimon. *Data Mining with Decision Trees: Theory and Applications*. World Scientific, 2014.

[69] E. Ikonomovska, J. Gama, and S. Džeroski. Learning model trees from evolving data streams. *Data Mining and Knowledge Discovery*, 23(1):128–168, 2011.

[70] G. Hulten, L. Spencer, and P. Domingos. Mining time-changing data streams. In *ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD)*, pages 97–106, 2001.

[71] R. De Rosa and N. Cesa-Bianchi. Splitting with confidence in decision trees with application to stream mining. In *International Joint Conference on Neural Networks (IJCNN)*, pages 1–8, 2015.

[72] I. Kononenko and M. Kukar. *Machine Learning and Data Mining: Introduction to Principles and Algorithms*. Horwood Publishing, 2007.

[73] W. Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963.

[74] E. Kaufmann, O. Cappé, and A. Garivier. On Bayesian upper confidence bounds for bandit problems. In *Artificial Intelligence and Statistics (AISTATS)*, pages 592–600, 2012.

[75] C. McDiarmid. On the method of bounded differences. *Surveys in Combinatorics*, 141(1):148–188, 1989.

[76] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[77] A. Garivier and E. Moulines. On upper-confidence bound policies for non-stationary bandit problems. *arXiv preprint arXiv:0805.3415*, 2008.

[78] F. Chung and L. Lu. Concentration inequalities and martingale inequalities: A survey. *Internet Mathematics*, 3(1):79–127, 2006.

[79] Monetate. Monetate ecommerce quarterly. `http://www.monetate.com/resources/research/`, 2015.

[80] O. Chapelle and L. Li. An empirical evaluation of thompson sampling. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2249–2257, 2011.

[81] Y. Xia, H. Li, T. Qin, N. Yu, and T. Liu. Thompson sampling for budgeted multi-armed bandits. In *International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[82] M. Sanjeev Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Transactions on Signal Processing*, 50(2):174–188, 2002.

[83] S. Agrawal and N. Goyal. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on Learning Theory (COLT)*, pages 39.1–39.26, 2012.

[84] A. Garivier and E. Moulines. On upper-confidence bound policies for switching bandit problems. In *International Conference on Algorithmic Learning Theory (ALT)*, pages 174–188, 2011.

[85] A. Garivier and O. Cappé. The kl-ucb algorithm for bounded stochastic bandits and beyond. In *Conference on Learning Theory (COLT)*, pages 359–376, 2011.

[86] P. Erdős and A. Rényi. On random graphs i. *Publicationes Mathematicae, Debrecen*, 6:290–297, 1959.

APPENDIX $\mathcal{A}$

Proofs of Theorems

A.1 Frequentist Approach

A.1.1 Proof of Theorem 1

Theorem 1. *If policy UCB1-M is run over a stationary MAB setting with a monotonic set A, the pseudo-regret after N rounds is at most:*

$$\bar{R}_N \leq \sum_{i|a_i \neq a_{i^*}} \frac{8a_i^2 \log(N)}{\Delta_i} + \sum_{i|a_i \neq a_{i^*}} \frac{2a_i^2 \log(K)}{\Delta_i} + \left(1 + \frac{\pi^2}{3}\right) \sum_{i=1}^{K} \Delta_i,$$

where $\Delta_i := a_{i^*}\mu_{i^*} - a_i\mu_i, \forall i \in \{1, \dots, K\}$.

Proof. Let us remind that we denote with $i^* := \arg\max_{i \in \{1,\dots,K\}} a_i\mu_i$ the index corresponding to the optimal arm a_{i^*}. Similarly to [6], we want to compute the expected number of times the policy UCB1-M does not pick the optimal arm a_{i^*} or, more formally, $\mathbb{E}[T_i(N)],\ \forall a_i \neq a_{i^*}$ and compute the regret as:

$$\bar{R}_N = \sum_{i|a_i \neq a_{i^*}} \Delta_i \mathbb{E}[T_i(N)].$$

Consider the round of the learning process at which a specific arm a_i has been selected for s rounds and define:

- $\bar{j}(i,t) := \bar{j}$ (with abuse of notation) as the index $j \in \{1, \dots, i\}$ minimizing the quantity $\bar{x}_{ji,t} + \sqrt{\frac{4\log(t)+\log(i)}{2T_{ji}(t-1)}}$, i.e., the upper bound of arm a_i;
- $\bar{j}^* := \bar{j}(i^*, t)$ as the index $j \in \{1, \dots, i^*\}$ minimizing the quantity $\bar{x}_{ji^*,t} + \sqrt{\frac{4\log(t)+\log(i^*)}{2T_{ji^*}(t-1)}}$, i.e., the upper bound of arm a_{i^*};
- $\bar{X}_{i,(s)}$ is the unbiased estimate of μ_i in the case we collected a total of s samples from arm a_i;
- $\bar{X}_{\bar{j}i,(s)}$ is the unbiased estimate of $\mu_{\bar{j}i,t,s} = \mathbb{E}\left[\bar{X}_{\bar{j}i,(s)}\right]$, in the case we collected a total of s samples from arm a_i (and thus we use $s' \geq s$ samples to estimate $\mu_{\bar{j}i,s}$);
- $c_{i,t,s} := \sqrt{\frac{4\log(t)+\log(i)}{2s}}$ as the Hoeffding bound with confidence $\frac{t^{-4}}{i}$ for $\bar{X}_{i,(s)}$ after t rounds;
- $c_{ji,t,s} := \sqrt{\frac{4\log(t)+\log(i)}{2s'}}$ as the Hoeffding bound with confidence $\frac{t^{-4}}{i}$ for $\bar{X}_{ji,(s)}$ after t rounds, in the case arm a_i has been pulled a total of s times and the arms $\{a_j, \dots, a_i\}$ have been chosen in total $s' > s$ times.

We have that, for each $l > 0$:

$$\begin{aligned}
T_i(N) &= 1 + \sum_{t=K+1}^{N} \mathbb{1}\{i_t = i\} \leq l + \sum_{t=K+1}^{N} \mathbb{1}\{i_t = i, T_i(t-1) \geq l\} \\
&\leq l + \sum_{t=K+1}^{N} \mathbb{1}\left\{a_{i^*}\bar{X}_{\bar{j}^* i^*,t} + a_{i^*} c_{\bar{j}^* i^*,t,T_{i^*}(t-1)} \leq a_i \bar{X}_{\bar{j}i,t} + \right. \\
&\quad \left. + a_i\ c_{\bar{j}i,t,T_i(t-1)}, T_i(t-1) \geq l\right\} \\
&\leq l + \sum_{t=K+1}^{N} \mathbb{1}\left\{ \min_{0<s<t} \left(a_{i^*}\bar{X}_{\bar{j}^* i^*,(s)} + a_{i^*} c_{\bar{j}^* i^*,t,s}\right) \leq \max_{l<s_i<t} \left(a_i \bar{X}_{\bar{j}i,(s_i)} + \right.\right. \\
&\quad \left.\left. + a_i c_{\bar{j}i,t,s_i}\right) \right\} \\
&\leq l + \sum_{t=1}^{\infty} \sum_{s=1}^{t-1} \sum_{s_i=l}^{t-1} \mathbb{1}\left\{a_{i^*}\bar{X}_{\bar{j}^* i^*,(s)} + a_{i^*} c_{\bar{j}^* i^*,t,s} \leq a_i \bar{X}_{\bar{j}i,(s_i)} + a_i c_{\bar{j}i,t,s_i}\right\}.
\end{aligned}$$

where we denoted with $\mathbb{1}\left\{B\right\}$ the indicator function of the event B.

If we consider:

$$\begin{aligned}
&a_{i^*}\bar{X}_{\bar{j}^*i^*,(s)} + a_{i^*}c_{\bar{j}^*i^*,t,s} \leq a_i\bar{X}_{\bar{j}i,(s_i)} + a_i c_{\bar{j}i,t,s_i},\\
&a_{i^*}\bar{X}_{\bar{j}^*i^*,(s)} + a_{i^*}c_{\bar{j}^*i^*,t,s} - a_i\bar{X}_{\bar{j}i,(s_i)} - a_i c_{\bar{j}i,t,s_i} \leq 0,\\
&a_{i^*}\bar{X}_{\bar{j}^*i^*,(s)} - a_{i^*}\mu_{i^*} + a_{i^*}c_{\bar{j}^*i^*,t,s} - a_i\bar{X}_{i,t,(s_i)} + a_i\mu_i + a_i c_{i,t,s_i} +\\
&\quad + a_{i^*}\mu_{i^*} - a_i\mu_i + a_i\bar{X}_{i,(s_i)} - a_i\bar{X}_{\bar{j}i,(s_i)} - a_i c_{\bar{j}i,t,s_i} - a_i c_{i,t,s_i} \leq 0,
\end{aligned}$$

we have that that, if the previous inequality is satisfied, at least one of the following inequalities is satisfied:

$$a_{i^*}\bar{X}_{\bar{j}^*i^*,(s)} \leq a_{i^*}\mu_{i^*} - a_{i^*}c_{\bar{j}^*i^*,t,s} \tag{A.1}$$

$$a_i\bar{X}_{i,(s_i)} \geq a_i\mu_i + a_i c_{i,t,s_i} \tag{A.2}$$

$$a_{i^*}\mu_{i^*} - a_i\mu_i + a_i\bar{X}_{i,(s_i)} - a_i\bar{X}_{\bar{j}i,(s_i)} - a_i c_{\bar{j}i,t,s_i} - a_i c_{i,t,s_i} \leq 0. \tag{A.3}$$

We need to bound the probabilities that the each one of the previous events occurs.

Probability of Event (A.1) By considering the fact that $\bar{X}_{\bar{j}^*i^*,(s)} + c_{\bar{j}^*i^*,t,s}$ is an upper bound for $\mu_{\bar{j}i,t,s}$ and thanks to the monotonicity assumption over μ_{i^*}, we can bound the probability of the events in Equation (A.1) as follows:

$$\begin{aligned}
&\mathbb{P}\left(a_{i^*}\bar{X}_{\bar{j}^*i^*,(s)} \leq a_{i^*}\mu_{i^*} - a_{i^*}c_{\bar{j}^*i^*,t,s}\right)\\
&= \mathbb{P}\left(\bar{X}_{\bar{j}^*i^*,(s)} \leq \mu_{i^*} - c_{\bar{j}^*i^*,t,s}\right)\\
&\leq \mathbb{P}\left(\bar{X}_{\bar{j}^*i^*,(s)} + c_{\bar{j}^*i^*,t,s} \leq \mu_{i^*}\right)\\
&\leq \mathbb{P}\left(\bar{X}_{\bar{j}^*i^*,(s)} + c_{\bar{j}^*i^*,t,s} \leq \mu_{\bar{j}i,t,s}\right) \leq e^{-4\log t} = t^{-4},
\end{aligned}$$

where the i term disappeared with the union bound over $\bar{X}_{ji^*,(s)}$ such that $1 \leq j \leq i^*$.

Probability of Event (A.2) By considering the Hoeffding bound we have that the event Equation (A.2) is bounded by:

$$\begin{aligned}
&\mathbb{P}(a_i\bar{X}_{i,(s_i)} \geq a_i\mu_i + a_i c_{i,t,s_i})\\
&= \mathbb{P}\left(\bar{X}_{i,(s_i)} \geq \mu_i + c_{i,t,s_i}\right) \leq e^{-4\log t - \log i} = \frac{t^{-4}}{i} \leq t^{-4}.
\end{aligned}$$

Probability of Event (A.3) Note that since the algorithm chooses the tightest bound among the set $\bar{X}_{ji,(s)} + c_{ji,t,s}$ with $j \leq i$ we have:

$$\begin{aligned}
a_i\bar{X}_{\bar{j}i,(s_i)} + a_i c_{\bar{j}i,t,s_i} &\leq a_i\bar{X}_{i,(s_i)} + a_i c_{i,t,s_i},\\
a_i\bar{X}_{\bar{j}i,(s_i)} - a_i\bar{X}_{i,(s_i)} + a_i c_{\bar{j}i,t,s_i} &\leq a_i c_{i,t,s_i},\\
a_i\bar{X}_{i,(s_i)} - a_i\bar{X}_{\bar{j}i,(s_i)} - a_i c_{\bar{j}i,t,s_i} &\geq -a_i c_{i,t,s_i}
\end{aligned}$$

If we consider $l = \left\lceil \frac{2a_i^2[4\log(t)+\log(i)]}{\Delta_i^2} \right\rceil$ the event in Equation (A.3) is not possible since:

$$0 \geq a_{i^*}\mu_{i^*} - a_i\mu_i + \underbrace{a_i\bar{X}_{i,(s_i)} - a_i\bar{X}_{\bar{j}i,(s_i)} - a_i c_{\bar{j}i,t,s_i}}_{\geq -a_i c_{i,t,s_i}} - a_i c_{i,t,s_i} \tag{A.4}$$

$$\geq \Delta_i - 2a_i\sqrt{\frac{4\log(t)+\log(i)}{2l}} > \Delta_i - \Delta_i = 0, \tag{A.5}$$

where we recall that $\Delta_i := a_{i^*}\mu_{i^*} - a_i\mu_i$.

Thus, since $\log(t) \leq \log(N)$ and $\log(i) \leq \log(K)$, $\forall i$ we have:

$$\begin{aligned}\mathbb{E}[T_i(N)] &\leq \left\lceil \frac{2a_i^2[4\log(N)+\log(K)]}{\Delta_i^2} \right\rceil + \sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_i=l}^{t-1} 2t^{-4} \\ &\leq \frac{8a_i^2\log(N)}{\Delta_i^2} + \frac{2a_i^2\log(K)}{\Delta_i^2} + 1 + \frac{\pi^2}{3}\end{aligned}$$

and the total regret becomes (since $\sum_{i=1}^K \mathbb{E}[T_i(N)] = N$):

$$\begin{aligned}\bar{R}_N &= a_{i^*}\mu_{i^*}N - \sum_{i=1}^K \mathbb{E}[T_i(N)]a_i\mu_i = \sum_{i=1}^K (a_{i^*}\mu_{i^*} - a_i\mu_i)\mathbb{E}[T_i(N)] \\ &\leq \sum_{i|a_i\neq a_{i^*}} \frac{8a_i^2\log(N)}{\Delta_i} + \sum_{i|a_i\neq a_{i^*}} \frac{2a_i^2\log(K)}{\Delta_i} + \left(1+\frac{\pi^2}{3}\right)\sum_{i=1}^K \Delta_i,\end{aligned}$$

which concludes the proof. □

A.1.2 Proof of Theorem 2

Theorem 2. *If policy UCBV-M is run with $\xi = 1.2$ and $c = 1$ over a setting with a monotonic set A, the pseudo-regret after N rounds is at most:*

$$\begin{aligned}\bar{R}_N \leq & \frac{12}{5} \sum_{i|a_i \neq a_{i^*}} a_i^2 \left(\frac{\sigma_i^2}{\Delta_i} + \frac{32}{15} \right) \log(N) + \\ & + \sum_{i|a_i \neq a_{i^*}} \Delta_i \left[1 + a_i^2 \left(\frac{\sigma_i^2}{\Delta_i^2} + \frac{2}{\Delta_i} \right) \log(K) \right],\end{aligned}$$

where $\sigma_i^2 := Var(X_{i,n})$, $\forall i \in \{1, \dots, K\}, \forall\, n \in \{1, \dots, T_i(N)\}$.

Proof. In what follows we make use of the notation used in Theorem 1. By following the proof of Theorem 3 in [36] we would like to bound the number of times a suboptimal arm is played:

$$\begin{aligned}\mathbb{E}[T_i(N)] \leq l_i + \sum_{t=l_i+1}^{N} \underbrace{\sum_{s=l_i}^{t-1} \mathbb{P}\left(a_i \bar{X}_{\bar{j}i,(s)} + a_i c_{\bar{j}i,t,s} \geq a_{i^*} \mu_{i^*} \right) +}_{T_{i1}} \\ + \sum_{t=l_i+1}^{N} \underbrace{\sum_{s=1}^{t-1} \mathbb{P}\left(a_{i^*} \bar{X}_{\bar{j}^* i^*,(s)} + a_{i^*} c_{\bar{j}^* i^*,t,s} \leq a_{i^*} \mu_{i^*} \right),}_{T_{i2}}\end{aligned}$$

where the inequality is due to Theorem 2 in [36]. Let us consider the two contribution to the regret separately.

Bound over $\mathbf{T_{i1}}$ The first contribution can be bounded as follows:

$$\begin{aligned}T_{i1} = & \sum_{s=l_i}^{t-1} \mathbb{P}\Big(a_i \bar{X}_{\bar{j}i,(s)} - a_i \bar{X}_{i,(s)} + a_i c_{\bar{j}i,t,s} - a_{i^*} \mu_{i^*} + a_i \mu_i + \\ & + a_i c_{i,t,s} + a_i \bar{X}_{i,(s)} - a_i \mu_i - a_i c_{i,t,s} \geq 0 \Big) \\ \leq & \sum_{s=l_i}^{t-1} \mathbb{P}\left(a_i \bar{X}_{\bar{j}i,(s)} - a_i \bar{X}_{i,(s)} + a_i c_{\bar{j}i,t,s} - a_{i^*} \mu_{i^*} + a_i \mu_i + a_i c_{i,t,s} > 0 \right) + \end{aligned} \tag{A.6}$$

$$+ \sum_{s=l_i}^{t-1} \mathbb{P}(a_i \bar{X}_{i,(s)} - a_i \mu_i - a_i c_{i,t,s} \geq 0). \tag{A.7}$$

By considering $s = l_i = \left\lceil 2a_i^2 \left(\frac{\sigma_i^2}{\Delta_i^2} + \frac{2}{\Delta_i} \right) \max\{c, 1\} \xi(log(t) + log(i)) \right\rceil$, where $\sigma_i^2 := Var(X_{it}),\ \forall i \in \{1, \dots, K\}, t \in \{1, \dots, N\}$, we have that:

$$\begin{aligned} 0 &< \underbrace{a_i \bar{X}_{\bar{j}i,(s)} - a_i^2 \bar{X}_{i,(s)} + a_i c_{\bar{j}i,t,s}}_{\leq a_i c_{i,t,s}} - a_{i^*} \mu_{i^*} + a_i \mu_i + a_i c_{i,t,s} \\ &\leq 2a_i c_{i,t,s} - \Delta_i \leq \Delta_i - \Delta_i = 0, \end{aligned}$$

where we used the fact that, by the choice made by the proposed algorithm, we have $a_i \bar{X}_{\bar{j}i,(s)} + a_i c_{\bar{j}i,t,s} \leq a_i \bar{X}_{i,(s)} + a_i c_{i,t,s}$for each $j \in \{1, \dots, i\}$. Thus, the contribution of the term in Equation (A.6) to the regret is null since the aforementioned event is impossible.

The term in Equation (A.7) can be bounded by Theorem 1 in [36] in the following way:

$$\begin{aligned} &\sum_{s=l_i}^{t-1} \mathbb{P}(a_i \bar{X}_{i,(s)} - a_i \mu_i - a_i c_{i,t,s} \geq 0) \\ &\sum_{s=l_i}^{t-1} \mathbb{P}(a_i \bar{X}_{i,(s)} - a_i \mu_i - a_i c_{i,t,s} \geq 0) \leq \beta(t, c, i) \leq \beta(t, c) \end{aligned}$$

where $\beta(t, c, i) := 3 \min\{c, 1\} \inf_{1 < \alpha \leq 3} \left[\left(\min \left\{ \frac{\log(t)}{\log(\alpha)}, t \right\} \right) (ti)^{-\frac{\xi}{\alpha}} \right]$ and $\beta(t, c) := \beta(t, c, 1)$.

Bound over $\mathbf{T_{i2}}$ By exploiting the monotonicity, i.e., since $\mu_{i^*} \leq \mu_{\bar{j}^* i^*, t, s}$ and by considering Theorem 1 in [36] we have:

$$\begin{aligned} T_{i2} &= \sum_{s=1}^{t-1} \mathbb{P} \left(a_{i^*} \bar{X}_{\bar{j}^* i^*, (s)} + a_{i^*} c_{\bar{j}^* i^*, t, s} \leq a_{i^*} \mu_{i^*} \right) \\ &= \sum_{s=1}^{t-1} \mathbb{P} \left(\bar{X}_{\bar{j}^* i^*, (s)} + c_{\bar{j}^* i^*, t, s} \leq \mu_{i^*} \right) \\ &\leq \sum_{s=1}^{t-1} \mathbb{P} \left(\bar{X}_{\bar{j}^* i^*, (s)} + c_{\bar{j}^* i^*, t, s} \leq \mu_{\bar{j}^* i^*} \right) \leq \beta(t, c), \end{aligned}$$

where for the monotonicity $\mu_{\bar{j}^* i^*} \geq \mu_{i^*}$ and we used a union bound over all the considered bounds ($j \in \{1, \dots, i\}$).

Regret $\mathbf{\bar{R}_N}$: Summing up, since $\log(t) \leq \log(N)$ and $\log(i) \leq \log(K)$, we have:

$$
\begin{aligned}
\bar{R}_N &= \sum_{i=1}^{K} \mathbb{E}[T_i(N)]\Delta_i \leq \sum_{i|a_i \neq a_{i^*}} (l_i + \sum_{t=l_i+1}^{N} T_{i1} + T_{i2}) \\
&\leq \sum_{i|a_i \neq a_{i^*}} \left[1 + 2a_i^2 \left(\frac{\sigma_i^2}{\Delta_i^2} + \frac{2}{\Delta_i}\right) \max\{c, 1\} \xi (log(t) + log(i)) + \right. \\
&\quad \left. + 2 \sum_{t=l_i+1}^{N} \beta(t, c)\right] \Delta_i \\
&\leq \sum_{i|a_i \neq a_{i^*}} \left[\frac{12}{5} a_i^2 \left(\frac{\sigma_i^2}{\Delta_i} + 2\right) \log(N) + 4c' \log(N)\right] + \\
&\quad + \sum_{i|a_i \neq a_{i^*}} \Delta_i \left[1 + a_i^2 \left(\frac{\sigma_i^2}{\Delta_i^2} + \frac{2}{\Delta_i}\right) \log(K)\right] \\
&\leq \frac{12}{5} \sum_{i|a_i \neq a_{i^*}} a_i^2 \left(\frac{\sigma_i^2}{\Delta_i} + \frac{32}{15}\right) \log(N) + \\
&\quad + \sum_{i|a_i \neq a_{i^*}} \Delta_i \left[1 + a_i^2 \left(\frac{\sigma_i^2}{\Delta_i^2} + \frac{2}{\Delta_i}\right) \log(K)\right],
\end{aligned}
$$

where by choosing $\xi = 1.2$ and $c = 1$ we have $\sum_{t=l_i+1}^{N} \beta(t, c) \leq c' \frac{2\log(N)}{\Delta_k}$ with $c' \leq 0.08$ (see proof of Theorem 4 in [36] for details). This concludes the proof.

□

A.1.3 Proof of Theorem 4

Let us recall that thanks to the Chernoff's theorem we have:

Theorem 3 (Theorem 4 in [78], Lower tail)**.** *Given a set of $T_i(t-1)$ independent and identically distributed random variables $\{X_{i,1}, \dots, X_{i,T_i(t-1)}\}$ such that $X_{i,s} \sim Be(\mu_i)$, for any $\varepsilon > 0$ we have:*

$$\mathbb{P}(\bar{X}_{i,t} + \varepsilon \leq \mu_i) \leq e^{-\frac{T_i(t-1)\varepsilon^2}{2\mu_i}}.$$

and also:

Theorem 13 (Theorem 4 in [78], Upper tail)**.** *Given a set of $T_i(t-1)$ independent and identically distributed random variables $\{X_{i,1}, \dots, X_{i,T_i(t-1)}\}$ such that $X_{i,s} \sim Be(\mu_i)$, for any $\varepsilon > 0$ we have:*

$$\mathbb{P}(\bar{X}_{i,t} - \varepsilon \geq \mu_i) \leq e^{-\frac{T_i(t-1)\varepsilon^2}{2\mu_i + \frac{\varepsilon}{3}}}.$$

Theorem 4. *If policy UCB-L is run over a stationary MAB setting with a set of arms A in which each arm $a_i \in A$ has outcome $X_{i,t}$ such that $\mathbb{E}[X_{i,t}] = \mu_i \leq \mu_{\max} \leq \frac{1}{2}$ for each $t \in \{1, \dots, N\}$, the pseudo-regret after N rounds is at most:*

$$\bar{R}_N \leq \sum_{i | a_i \neq a_{i^*}} \frac{32 \mu_{\max} a_i^2 \log(N)}{\Delta_i} + \left[1 + \frac{\pi^2}{6} + \zeta\left(\frac{10}{7}\right)\right] \sum_{i=1}^{K} \Delta_i,$$

where $\zeta(\cdot)$ is the Riemann zeta function.

Proof. In what follows we make use of the notation used in Theorem 1. Let us recall that $\mu_{\max} \geq \mu_i,\ \forall i \in \{1, \dots, K\}$. By defining:

$$\varepsilon_{i,t,T_i(t-1)} := \sqrt{\frac{8\mu_{max} \log(t)}{T_i(t-1)}},$$

we have that, similarly to what has been derived in Theorem 1, for each $l > 0$:

$$T_i(N) \leq l + \sum_{t=1}^{\infty} \sum_{s=1}^{t-1} \sum_{s_i=l}^{t-1} \mathbb{1}\{a_{i^*} \bar{X}_{i^*,(s)} + a_{i^*} \varepsilon_{i^*,t,s} \leq a_i \bar{X}_{i,(s_i)} + a_i \varepsilon_{i,t,s_i}\}.$$

If we consider the event in the previous inequality, we have:

$$
\begin{aligned}
&a_{i^*}\bar{X}_{i^*,(s)} + a_{i^*}\varepsilon_{i^*,t,s} \leq a_i\bar{X}_{i,(s_i)} + a_i\varepsilon_{i,t,s_i} \\
&a_{i^*}\bar{X}_{i^*,(s)} - a_{i^*}\mu_{i^*} + a_{i^*}\varepsilon_{i^*,t,s} + a_{i^*}\mu_{i^*} \leq a_i\bar{X}_{i,(s_i)} - a_i\mu_i - a_i\varepsilon_{i,t,s_i} + \\
&\quad + a_i\mu_i + 2a_i\varepsilon_{i,t,s_i} \\
&a_{i^*}\bar{X}_{i^*,(s)} - a_{i^*}\mu_{i^*} + a_{i^*}\varepsilon_{i^*,t,s} - a_i\bar{X}_{i,(s_i)} + a_i\mu_i + a_i\varepsilon_{i,t,s_i} + a_{i^*}\mu_{i^*} - \\
&\quad - a_i\mu_i - 2a_i\varepsilon_{i,t,s_i} \leq 0,
\end{aligned}
$$

we have that it implies that at least one of the following inequalities is satisfied:

$$a_{i^*}\bar{X}_{i^*,(s)} \leq a_{i^*}\mu_{i^*} - a_{i^*}\varepsilon_{i^*,t,s} \tag{A.8}$$

$$a_i\bar{X}_{i,(s_i)} \geq a_i\mu_i + a_i\varepsilon_{i,t,s_i} \tag{A.9}$$

$$a_{i^*}\mu_{i^*} - a_i\mu_i < 2a_i\varepsilon_{i,t,s_i}. \tag{A.10}$$

Let us focus on the event in Equation (A.8). Thanks to Theorem 3 we are able to bound the probability of this event:

$$
\begin{aligned}
&\mathbb{P}(a_{i^*}\bar{X}_{i^*,(s)} \leq a_{i^*}\mu_{i^*} - a_{i^*}\varepsilon_{i^*,t,s}) = \mathbb{P}\left(\bar{X}_{i^*,(s)} \leq \mu_{i^*} - \varepsilon_{i^*,t,s}\right) \\
&\leq e^{-\frac{s(\varepsilon_{i^*,t,s})^2}{2\mu^*}} \leq e^{-\frac{s(\varepsilon_{i^*,t,s})^2}{2\mu_{\max}}} = e^{-4\log t} = t^{-4}.
\end{aligned}
$$

By relying on the upper tail of the Chernoff's bound, as described in Theorem 13 (cited in this appendix) we can bound the probability of the event in Equation (A.9):

$$
\begin{aligned}
&\mathbb{P}(a_i\bar{X}_{i,(s_i)} \geq a_i\mu_i + a_i\varepsilon_{i,t,s_i}) = \mathbb{P}\left(\bar{X}_{i,(s_i)} \geq \mu_i + \varepsilon_{i,t,s_i}\right) \\
&\leq \exp\left\{-\frac{s_i(\varepsilon_{i,t,s_i})^2}{2\mu_i + \frac{\varepsilon_{i,t,s_i}}{3}}\right\} \leq e^{-\frac{s_i(\varepsilon_{i,t,s_i})^2}{\frac{7}{3}\mu_{\max}}} \leq t^{-\frac{24}{7}},
\end{aligned}
$$

where we consider $\varepsilon_{i,t,s_i} \leq \mu_{\max}$ and $\mu_i \leq \mu_{\max} \leq \frac{1}{2}$. At last, if we focus on the event in Equation (A.10) and we consider $l = \left\lceil \frac{32\mu_{\max}a_i^2\log(t)}{\Delta_i^2} \right\rceil$, where $\Delta_i = a_{i^*}\mu_{i^*} - a_i\mu_i$, the event in Equation (A.3) is not possible since:

$$
\begin{aligned}
&0 \geq a_{i^*}\mu_{i^*} - a_i\mu_i - 2a_i\varepsilon_{i,t,s_i} \\
&\underbrace{\geq}_{s_i \geq l} a_{i^*}\mu_{i^*} - a_i\mu_i - 2a_i\varepsilon_{i,t,l} \geq a_{i^*}\mu_{i^*} - a_i\mu_i - a_{i^*}\mu_{i^*} - a_i\mu_i = 0.
\end{aligned}
$$

Finally we have:

$$\mathbb{E}[T_i(N)] \leq \left\lceil \frac{32\mu_{\max} a_i^2 \log(t)}{\Delta_i^2} \right\rceil + \sum_{t=1}^{\infty} \sum_{s=1}^{t-1} \sum_{s_i=l}^{t-1} (t^{-4} + t^{-\frac{24}{7}})$$
$$\leq \frac{32\mu_{\max} a_i^2 \log(N)}{\Delta_i^2} + 1 + \frac{\pi^2}{6} + \zeta\left(\frac{10}{7}\right)$$

where $\zeta(\cdot)$ is the Riemann zeta function. The total regret becomes (since $\sum_{i=1}^{K} \mathbb{E}[T_i(N)] = N$):

$$\bar{R}_N = a_{i^*}\mu_{i^*}N - \sum_{i=1}^{K} \mathbb{E}[T_i(N)]a_i\mu_i = \sum_{i=1}^{K}(a_{i^*}\mu_{i^*} - a_i\mu_i)\mathbb{E}[T_i(N)]$$
$$\leq \sum_{i|a_i \neq a_{i^*}} \frac{32\mu_{\max} a_i^2 \log(N)}{\Delta_i} + \left[1 + \frac{\pi^2}{6} + \zeta\left(\frac{10}{7}\right)\right] \sum_{i=1}^{K} \Delta_i,$$

which concludes the proof. □

A.1.4 Proof of Theorem 5

Theorem 5. *If policy UCB-LM is run over a stationary MAB setting with a monotonic set A in which each arm $a_i \in A$ has outcome $X_{i,t}$ such that $\mathbb{E}[X_{i,t}] = \mu_i \leq \mu_{\max} \leq \frac{1}{2}$ for each t, the pseudo-regret after N rounds is at most:*

$$\bar{R}_N \leq \sum_{i|a_i \neq a_{i^*}} \frac{32\mu_{\max} a_i^2 \log(N)}{\Delta_i} + \sum_{i|a_i \neq a_{i^*}} \frac{8\mu_{\max} a_i^2 \log(K)}{\Delta_i} + \\ + \left[1 + \frac{\pi^2}{6} + \zeta\left(\frac{10}{7}\right)\right] \sum_{i=1}^{K} \Delta_i,$$

where $\zeta(\cdot)$ is the Riemann zeta function.

Proof. The proof is a straightforward combination of the arguments used for the UCB1-M and UCB-L ones. Consider the round of the learning process at which a specific arm a_i has been selected for s rounds and define:

- $\bar{j}(i,t) := \bar{j}$ (with abuse of notation) as the index $j \in \{1, \dots, i\}$ minimizing the quantity $\bar{x}_{ji,t} + \sqrt{\frac{2\mu_{\max}[4\log(t)+\log(i)]}{T_{ji}(t-1)}}$, i.e., the upper bound of arm a_i;

- $\bar{j}^* := \bar{j}(i^*, t)$ as the index $j \in \{1, \dots, i^*\}$ minimizing the quantity $\bar{x}_{ji^*,t} + \sqrt{\frac{2\mu_{\max}[4\log(t)+\log(i^*)]}{T_{ji^*}(t-1)}}$, i.e., the upper bound of arm a_{i^*};

- $\bar{X}_{i,(s)}$ is the unbiased estimate of μ_i in the case we collected a total of s samples from arm a_i;

- $\bar{X}_{\bar{j}i,(s)}$ is the unbiased estimate of $\mu_{\bar{j}i,t,s} = \mathbb{E}\left[\bar{X}_{\bar{j}i,(s)}\right]$, in the case we collected a total of s samples from arm a_i (and thus we have $s' \geq s$ samples to estimate $\mu_{\bar{j}i,s}$);

- $c_{i,t,s} := \sqrt{\frac{2\mu_{\max}[4\log(t)+\log(i)]}{s}}$ as the Hoeffding bound with confidence $\frac{t^{-4}}{i}$ for $\bar{X}_{i,(s)}$ after t rounds;

- $c_{ji,t,s} := \sqrt{\frac{2\mu_{\max}[4\log(t)+\log(i)]}{s'}}$ as the Hoeffding bound with confidence $\frac{t^{-4}}{i}$ for $\bar{X}_{ji,(s)}$ after t rounds, in the case arm a_i has been pulled a total of s times and the arms $\{a_j, \dots, a_i\}$ have been chosen in total $s' > s$ times.

We have that, for each $l > 0$:

$$T_i(N) \leq l + \sum_{t=1}^{\infty} \sum_{s=1}^{t-1} \sum_{s_i=l}^{t-1} \mathbb{1}\left\{a_{i^*}\bar{X}_{\bar{j}^*i^*,(s)} + a_{i^*}c_{\bar{j}^*i^*,t,s} \leq a_i\bar{X}_{\bar{j}i,(s_i)} + a_i c_{\bar{j}i,t,s}\right\}$$

and consequently we only need to bound the probability of these three events:

$$a_{i^*}\bar{X}_{\bar{j}^*i^*,(s)} \leq a_{i^*}\mu_{i^*} - a_{i^*}c^{\bar{j}^*i^*,t,s} \tag{A.11}$$

$$a_i\bar{X}_{i,(s_i)} \geq a_i\mu_i + a_i c_{i,t,s_i} \tag{A.12}$$

$$a_{i^*}\mu_{i^*} - a_i\mu_i + a_i\bar{X}_{i,(s_i)} - a_i\bar{X}_{\bar{j}i,(s_i)} - a_i c_{\bar{j}i,t,s_i} - a_i c_{i,t,s_i} \leq 0. \tag{A.13}$$

Similarly to what has been done for Theorem 1, the probability of the event in Equation (A.11) can be bounded by t^{-4} by using the monotonicity assumption and Theorem 3, the one corresponding to the event in Equation (A.12) is bounded by $t^{\frac{24}{7}}$ by using the Chernoff theorem (Theorem 13, which considers the upper tail) and the event in Equation (A.13) is not possible if we choose $l = \left\lceil \frac{8a_i^2\mu_{\max}[4\log(t)+\log(i)]}{\Delta_i^2} \right\rceil$. Thus, by considering that $\log(t) \leq \log(N)$ and $\log(i) \leq \log(K)$, $\forall i$, we have:

$$\begin{aligned}
\bar{R}_N &= a_{i^*}\mu_{i^*}N - \sum_{i=1}^{K} \mathbb{E}[T_i(N)]a_i\mu_i \\
&= \sum_{i=1}^{K} (a_{i^*}\mu_{i^*} - a_i\mu_i)\mathbb{E}[T_i(N)] \\
&\leq \sum_{i|a_i \neq a_{i^*}} \frac{32\mu_{\max}a_i^2\log(N)}{\Delta_i} + \sum_{i|a_i \neq a_{i^*}} \frac{8\mu_{\max}a_i^2\log(K)}{\Delta_i} + \\
&\quad + \left[1 + \frac{\pi^2}{6} + \zeta\left(\frac{10}{7}\right)\right] \sum_{i=1}^{K} \Delta_i,
\end{aligned}$$

which concludes the proof. □

A.1.5 Proof of Theorem 7

Theorem 7. *If policy SW-UCB-M is run over a non-stationary MAB setting $\mathcal{S}^{(B)}$, for any $\tau \in \mathbb{N}$ and $\xi > \frac{1}{2}$, the pseudo-regret after N rounds is at most:*

$$\bar{R}_N \leq \sum_{i=1}^{K} \left[\frac{N}{\tau} \frac{4a_i^2 \xi [\log(i) + \log(\tau)]}{\Delta_i} + a_i \Upsilon_N \tau + \right.$$

$$\left. + \frac{2N}{\tau} \left\lceil \frac{\log(\tau)}{\log\left(1 + 4\sqrt{1 - \frac{1}{2\xi}}\right)} \right\rceil \right],$$

where Υ_N is the number of breakpoints before N and

$$\Delta_i := \min_{\phi \in \{1,\dots,\Upsilon_N\}} \left(a_{i^*_\phi} \mu_{i^*_\phi,\phi} - a_i \mu_{i,\phi} \right) \mathbb{1}\{i \neq i^*_\phi\} \quad \forall i \in \{1, \dots, K\},$$

denotes the minimum, over all the phases Φ_ϕ in which the arm a_i is not optimal, of the difference of the expected reward $a_{i^*_\phi} \mu_{i^*_\phi,\phi}$ of the best arm $a_{i^*_\phi}$ and the expected reward $a_i \mu_{i,\phi}$ of the arm a_i.

Proof. Consider the phases $\phi \in \{1, \dots, \Upsilon_N\}$ introduced in Section 5.1. Let us define:

$$A_{i,\phi}(\tau) = \frac{4a_i^2 \xi [\log(i) + \log(\tau)]}{\Delta_{i,\phi}^2},$$

where $\Delta_{i,\phi} = a_{i^*_\phi} \mu_{i^*_\phi,\phi} - a_i \mu_{i,\phi}$, $\forall i \in \{1, \dots, K\} \setminus \{i^*_\phi\}$.

Let us denote with $T_i(\Phi'_\phi)$ the number of times an arm a_i, with $i \in \{1, \dots, K\} \setminus \{i^*_\phi\}$, has been played when it was not the best arm during the rounds $t \in \Phi'_\phi := \{t | b_{\phi-1} + \tau \leq t < b_\phi\}$. We consider $\tau < N_\phi$, i.e., τ is smaller than the number of rounds in each phase.[1]

We can bound the number of times we are pulling an arm as:

$$T_i(N) = \sum_{\phi=1}^{\Upsilon_N} T_i(\Phi_\phi) \leq \sum_{\phi=1}^{\Upsilon_N} \tau + T_i(\Phi'_\phi)$$

where we assume that $\tau > K$.

Let us focus on a single phase Φ_ϕ. Consider the number of times a sub-

[1] We make this assumption for ease of notation. In the case $\exists \tau > N_\phi$, it is straightforward to extend the analysis.

optimal arm $a_i \neq a_{i^*_\phi}$ has been pulled, we have:

$$T_i(\Phi'_\phi) = \sum_{t \in \Phi'_\phi} \mathbb{1}\{i_t = i\}$$

$$\leq \sum_{t \in \Phi'_\phi} \mathbb{1}\{i_t = i, T_i(t-1,\tau) < A_{i,\phi}(\tau)\} + \tag{A.14}$$

$$+ \sum_{t \in \Phi'_\phi} \mathbb{1}\{i_t = i, T_i(t-1,\tau) \geq A_{i,\phi}(\tau)\} \tag{A.15}$$

where i_t is the index of the arm a_{i_t} selected at round t by policy SW-UCB-M with a window of size τ.

By using Lemma 25 in [77], we can bound the first term of Equation (A.15), we have:

$$T_i(\Phi'_\phi) \leq \left\lceil \frac{|\Phi'_\phi|}{\tau} \right\rceil A_{i,\phi}(\tau) + \sum_{t \in \Phi'_\phi} \mathbb{1}\{i_t = i, T_i(t-1,\tau) \geq A_{i,\phi}(\tau)\}$$

$$\leq \left\lceil \frac{N_\phi - \tau}{\tau} \right\rceil A_{i,\phi}(\tau) + \sum_{t \in \Phi'_\phi} \mathbb{1}\{i_t = i, T_i(t-1,\tau) \geq A_{i,\phi}(\tau)\}$$

$$\leq \frac{N_\phi}{\tau} A_{i,\phi}(\tau) + \sum_{t \in \Phi'_\phi} \mathbb{1}\{i_t = i, T_i(t-1,\tau) \geq A_{i,\phi}(\tau)\}. \tag{A.16}$$

Let us focus on the second term of the last expression. The event $i_t = i$ occurs when:

$$a_{i^*_\phi} \bar{X}_{\bar{j}^* i^*_\phi, t, \tau} + a_{i^*_\phi} \varepsilon_{i^*_\phi, t, T_{\bar{j}^* i^*}(t-1), \tau} \leq a_i \bar{X}_{\bar{j}i, t, \tau} + a_i \varepsilon_{i, t, T_{\bar{j}i}(t-1), \tau}$$
$$a_{i^*_\phi} \bar{X}_{\bar{j}^* i^*_\phi, t, \tau} - a_{i^*_\phi} \mu_{i^*_\phi, \phi} + a_{i^*_\phi} \varepsilon_{i^*_\phi, t, T_{\bar{j}^* i^*}(t-1), \tau} - a_i \bar{X}_{i, t, \tau} + a_i \mu_{i, \phi} + a_i \varepsilon_{i, t, T_i(t-1), \tau} +$$
$$+ a_{i^*_\phi} \mu_{i^*_\phi, \phi} - a_i \mu_{i, \phi} + a_i \bar{X}_{i, t, \tau} - a_i \bar{X}_{\bar{j}i, t, \tau} - a_i \varepsilon_{i, t, T_i \bar{j} i(t-1), \tau} - a_i \varepsilon_{i, t, T_i(t-1), \tau}$$

where $\varepsilon_{i,t,T_{ji}(t-1),\tau} := \sqrt{\dfrac{\xi[\log(i) + \log(\min\{t,\tau\})]}{T_{ji}(t-1)}} = \sqrt{\dfrac{\xi[\log(i) + \log(\tau)]}{T_{ji}(t-1)}}$, since $t \in \Phi'_\phi \Rightarrow t > \tau$ and it is contained in the union of the following three events:

$$a_{i^*_\phi} \bar{X}_{\bar{j}^* i^*_\phi, t, \tau} \leq a_{i^*_\phi} \mu_{i^*_\phi, \phi} - a_{i^*_\phi} \varepsilon_{i^*_\phi, t, T_{\bar{j}^* i^*}(t-1), \tau}; \tag{A.17}$$

$$a_i \bar{X}_{i, t, \tau} \geq a_i \mu_{i, \phi} + a_i \varepsilon_{i, t, T_i(t-1), \tau}; \tag{A.18}$$

$$a_{i^*_\phi} \mu_{i^*_\phi, \phi} - a_i \mu_{i, \phi} + a_i \bar{X}_{i, t, \tau} - a_i \bar{X}_{\bar{j}i, t, \tau} - a_i \varepsilon_{i, t, T_{\bar{j}i}(t-1), \tau} - a_i \varepsilon_{i, t, T_i(t-1), \tau} \leq 0. \tag{A.19}$$

Let us define $\delta = \varepsilon_{i,t,T_i(t-1),\tau}\sqrt{T_i(t-1,\tau)} = \sqrt{\xi[\log(i)+\log(\tau)]}$ and consider the probability of the event in Equation (A.18), we have:

$$\begin{aligned}
&\mathbb{P}\left(a_i\bar{X}_{i,t,\tau} \geq a_i\mu_{i,\phi} + a_i\frac{\delta}{\sqrt{T_i(t-1,\tau)}}\right)\\
&= \mathbb{P}\left(\bar{X}_{i,t,\tau} - \mu_{i,\phi} \geq \frac{\delta}{\sqrt{T_i(t-1,\tau)}}\right)\\
&\leq \mathbb{P}\left(\bar{X}_{i,t,\tau} - \mu_{i,\phi} \geq \frac{\delta}{\sqrt{T_i(t-1,\tau)}}\right)\\
&= \mathbb{P}\left(\frac{T_i(t-1,\tau)\left(\bar{X}_{i,t,\tau} - \mu_{i,\phi}\right)}{\sqrt{T_i(t-1,\tau)}} \geq \delta\right).
\end{aligned}$$

By applying Corollary 21 in [77] we have that for all $\eta > 0$:

$$\begin{aligned}
&\mathbb{P}\left(\frac{T_i(t-1,\tau)\left(\bar{X}_{i,t,\tau} - \mu_{i,\phi}\right)}{\sqrt{T_i(t-1,\tau)}} \geq \delta\right)\\
&\leq \left\lceil \frac{\log(\tau)}{\log(1+\eta)} \right\rceil \exp\left(-2\delta^2\left(1-\frac{\eta^2}{16}\right)\right)\\
&\leq \left\lceil \frac{\log(\tau)}{\log(1+\eta)} \right\rceil \exp\left(-2\xi[\log(i)+\log(\tau)]\left(1-\frac{\eta^2}{16}\right)\right)\\
&= \left\lceil \frac{\log(\tau)}{\log(1+\eta)} \right\rceil (i\tau)^{-2\xi\left(1-\frac{\eta^2}{16}\right)}
\end{aligned}$$

where we consider the events of choosing arms a_i as the sequence of previsible variables.

Similarly, by exploiting the monotonicity property, we have that for each

$j \leq i^*_\phi$ and by defining $\delta = \varepsilon_{i,t,T_{ji^*_\phi}(t-1),\tau}\sqrt{T_{ji^*_\phi}(t-1,\tau)}$:

$$\begin{aligned}
&\mathbb{P}\left(a_{i^*_\phi}\bar{X}_{ji^*_\phi,t,\tau} \leq a_{i^*_\phi}\mu_{i^*_\phi,\phi} - a_{i^*_\phi}\varepsilon_{i^*_\phi,t,T_{ji^*}(t-1),\tau}\right)\\
&\leq \mathbb{P}\left(a_{i^*_\phi}\bar{X}_{ji^*_\phi,t,\tau} \leq a_{i^*_\phi}\mu_{ji^*_\phi,\phi} - a_{i^*_\phi}\varepsilon_{i^*_\phi,t,T_{ji^*}(t-1),\tau}\right)\\
&= \mathbb{P}\left(a_{i^*_\phi}\bar{X}_{ji^*_\phi,t,\tau} \geq a_{i^*_\phi}\mu_{ji^*_\phi,\phi} + a_{i^*_\phi}\varepsilon_{i^*_\phi,t,T_{ji^*}(t-1),\tau}\right)\\
&= \mathbb{P}\left(\bar{X}_{ji^*_\phi,t,\tau} - \mu_{ji^*_\phi,\phi} \geq \varepsilon_{i^*_\phi,t,T_{ji^*}(t-1),\tau}\right)\\
&\left\lceil\frac{\log(\tau)}{\log(1+\eta)}\right\rceil \exp\left(-2\delta^2\left(1-\frac{\eta^2}{16}\right)\right)\\
&\leq \left\lceil\frac{\log(\tau)}{\log(1+\eta)}\right\rceil \exp\left(-2\xi[\log(i)+\log(\tau)]\left(1-\frac{\eta^2}{16}\right)\right)\\
&= \left\lceil\frac{\log(\tau)}{\log(1+\eta)}\right\rceil (i\tau)^{-2\xi\left(1-\frac{\eta^2}{16}\right)}\\
&= \left\lceil\frac{\log(\tau)}{\log(1+\eta)}\right\rceil (\tau)^{-2\xi\left(1-\frac{\eta^2}{16}\right)},
\end{aligned}$$

where first equality sign is due to the symmetry of the Bernoulli distribution, the event of choosing an arm among the set $\{a_j, \dots a_{i^*_\phi}\}$ has been chosen as the sequence of previsible Bernoulli variables.

Thus, the probability of the event in Equation (A.17) can be bounded by:

$$\begin{aligned}
&\mathbb{P}\left(a_{i^*_\phi}\bar{X}_{\bar{j}^*i^*_\phi,t,\tau} \leq a_{i^*_\phi}\mu_{i^*_\phi,\phi} - a_{i^*_\phi}\varepsilon_{i^*_\phi,t,T_{\bar{j}^*i^*}(t-1),\tau}\right)\\
&= \left\lceil\frac{\log(\tau)}{\log(1+\eta)}\right\rceil (\tau)^{-2\xi\left(1-\frac{\eta^2}{16}\right)},
\end{aligned}$$

by resorting to an union bound over all $j \leq i$.

Finally, consider the event in Equation (A.19) and that $T_i(t-1,\tau) \geq A_{i,\phi}(\tau)$:

$$\begin{aligned}
0 &\geq \Delta_{i,\phi} + \underbrace{a_i\bar{X}_{i,t,\tau} - a_i\bar{X}_{\bar{j}i,t,\tau} - a_i\varepsilon_{i,t,T_{\bar{j}i}(t-1),\tau}}_{\geq -a_i\varepsilon_{i,t,T_i(t-1),\tau}} - a_i\varepsilon_{i,t,T_i(t-1),\tau}\\
&\geq \Delta_{i,\phi} - 2a_i\varepsilon_{i,t,T_i(t-1),\tau} > 0;
\end{aligned}$$

where the inequality is given from the fact that the SW-UCB-M algorithm chooses the tightest bound among the $a_i\bar{X}_{ji,t,\tau} + a_i\varepsilon_{i,t,T_{ji}(t-1),\tau}$ with $1 \leq j \leq i$. Since the last expression is a contradiction, the considered event does not occur.

By choosing $\eta = 4\sqrt{1-\frac{1}{2\xi}}$ we have $2\xi\left(1-\frac{\eta^2}{16}\right) = 1$ and we get:

$$
\begin{aligned}
\mathbb{E}[T_i(\Phi'_\phi)] &\le \frac{N_\phi}{\tau}A_{i,\phi}(\tau) + 2\sum_{t\in\Phi'_\phi}\frac{\left\lceil\frac{\log(\tau)}{\log(1+\eta)}\right\rceil}{\tau} \\
&= \frac{N_\phi}{\tau}A_{i,\phi}(\tau) + \frac{2|\Phi'_\phi|}{\tau}\frac{\log(\tau)}{\log(1+\eta)} \\
&\le \frac{N_\phi}{\tau}\frac{4a_i^2\xi[\log(i)+\log(\tau)]}{\Delta^2_{i,\phi}} + \frac{2N_\phi}{\tau}\left\lceil\frac{\log(\tau)}{\log\left(1+4\sqrt{1-\frac{1}{2\xi}}\right)}\right\rceil
\end{aligned}
$$

The total regret becomes:

$$
\begin{aligned}
\bar{R}_N &= \sum_{\phi=1}^{\Upsilon_N}\left(a_{i^*,\phi}\mu_{i^*,\phi}N_\phi - \sum_{i=1}^{K}a_i\mu_{i,\phi}\mathbb{E}[T_i(\Phi_\phi)]\right) \\
&= \sum_{\phi=1}^{\Upsilon_N}\left(\sum_{i=1}^{K}(a_{i^*,\phi}\mu_{i^*,\phi} - a_i\mu_{i,\phi})\mathbb{E}[T_i(\Phi_\phi)]\right) \\
&= \sum_{i=1}^{K}\left(\sum_{\phi=1}^{\Upsilon_N}(a_{i^*,\phi}\mu_{i^*,\phi} - a_i\mu_{i,\phi})\mathbb{E}[T_i(\Phi_\phi)]\right) \\
&\le \sum_{i=1}^{K}\left(\sum_{\phi=1}^{\Upsilon_N}\Delta_{i,\phi}\mathbb{E}[T_i(\Phi_\phi)]\right) \\
&\le \sum_{i=1}^{K}\left[\sum_{\phi=1}^{\Upsilon_N}\Delta_{i,\phi}\left(\tau + \mathbb{E}[T_i(\Phi_\phi)]\right)\right] \\
&\le \sum_{i=1}^{K}\Bigg[a_i\Upsilon_N\tau + \sum_{\phi=1}^{\Upsilon_N}\Delta_{i,\phi}\Bigg(\frac{N_\phi}{\tau}\frac{4a_i^2\xi[\log(i)+\log(\tau)]}{\Delta^2_{i,\phi}} + \\
&\qquad + \frac{2N_\phi}{\tau}\left\lceil\frac{\log(\tau)}{\log\left(1+4\sqrt{1-\frac{1}{2\xi}}\right)}\right\rceil\Bigg)\Bigg]
\end{aligned}
$$

Considering Δ_i as defined in in the theorem statement, we obtain:

$$\bar{R}_N \leq \sum_{i=1}^{K} \left[\frac{N}{\tau} \frac{4a_i^2 \xi [\log(i) + \log(\tau)]}{\Delta_i} + \right.$$

$$\left. + a_i \Upsilon_N \tau + \frac{2N}{\tau} \left\lceil \frac{\log(\tau)}{\log\left(1 + 4\sqrt{1 - \frac{1}{2\xi}}\right)} \right\rceil \right],$$

which concludes the proof. □

A.2 Bayesian Approach

A.2.1 Bayes update rule

Here we report the derivation of the update rule for the Bayesian approach, presented in Equation (6.1), and the Sequential Monte Carlo (SMC) scheme, presented in Equation (6.2).

Assume to have a monotonic set of arms $A = \{a_1, \dots, a_K\}$ as defined in Section 5.1. Given a generic random variable M_i whose distribution is the prior μ_i for all $i \in \{1, \dots, K\}$ and a policy $\mathfrak{U}(h_t)$ to select arm a_{i_t}, we have a realization $x_{i_t,t}$ of X_{i_t} at time t. If we consider the generic i-th arm, we have different Bayesian updates depending on the fact that $i_t = i$, $i_t < i$ or $i_t > i$. In fact, if $i_t = i$, we are able to update the prior directly, i.e.,

$$\mathbb{P}_t(\mu_i) := \mathbb{P}_t(\mu_i | x_{i_t,t}) \propto \mathbb{P}(x_{i_t,t} | \mu_i) \mathbb{P}_{t-1}(\mu_i),$$

where $\mathbb{P}(x_{i_t,t} | \mu_{i_t})$ the the loglikelihood of the realization $x_{i_t,t}$. If $i_t \neq i$ we have:

$$\begin{aligned}
\mathbb{P}_t(\mu_i | x_{i_t,t}) \propto \mathbb{P}(x_{i_t,t} | \mu_i) \mathbb{P}_{t-1}(\mu_i) &= \int_{\mu_j \in [0,1]} \mathbb{P}(x_{i_t,t}, \mu_j | \mu_i) \mathbb{P}_{t-1}(\mu_i) \, \mathrm{d}\mu_j \\
&= \int_{\mu_j \in [0,1]} \mathbb{P}_{t-1}(x_{i_t,t}, \mu_j, \mu_i) \, \mathrm{d}\mu_j = \\
&= \int_{\mu_j \in [0,1]} \mathbb{P}_{t-1}(x_{i_t,t}, \mu_i | \mu_j) \mathbb{P}_{t-1}(M_{i_t} = \mu_j) \, \mathrm{d}\mu_j = \\
&= \int_{\mu_j \in [0,1]} \mathbb{P}(x_{i_t,t} | \mu_j) \mathbb{P}_{t-1}(\mu_i | \mu_j) \mathbb{P}_{t-1}(M_{i_t} = \mu_j) \, \mathrm{d}\mu_j = \\
&= \int_{\mu_j \in [0,1]} \mathbb{P}(x_{i_t,t} | \mu_j) \frac{\mathbb{P}_{t-1}(\mu_j | \mu_i) \mathbb{P}_{t-1}(\mu_i)}{\mathbb{P}_{t-1}(M_{i_t} = \mu_j)} \mathbb{P}_{t-1}(M_{i_t} = \mu_j) \, \mathrm{d}\mu_j = \\
&= \mathbb{P}_{t-1}(\mu_i) \int_{\mu_j \in [0,1]} \mathbb{P}(x_{i_t,t} | \mu_j) \mathbb{P}_{t-1}(\mu_j | \mu_i) \, \mathrm{d}\mu_j \qquad \text{(A.20)}
\end{aligned}$$

where we assume conditional independence of $X_{i_t} \perp \mu_i | \mu_j$ for every $i, j \in \{1, \dots, K\}$ with $i \neq j$.

Since we have the assumption monotony on A, if $i_t > i \Rightarrow \mu_{i_t} \leq \mu_i$ thus:

$$\mathbb{P}_{t-1}(\mu_j | \mu_i) = \begin{cases} 0 & \mu_j > \mu_i \\ \dfrac{\mathbb{P}_{t-1}(M_{i_t} = \mu_j)}{\int_0^{\mu_i} \mathbb{P}_{t-1}(M_{i_t} = x) \mathrm{d}x} & \mu_j \leq \mu_i \end{cases} \qquad \text{(A.21)}$$

By substituting Equation (A.21) in Equation (A.20), we obtain:

$$\begin{aligned}\mathbb{P}_{t-1}(\mu_i|x_j) &\propto \mathbb{P}_{t-1}(\mu_i)\int_{\mu_j\in[0,1]}\mathbb{P}(x_{i_t,t}|\mu_j)\mathbb{P}_{t-1}(\mu_j|\mu_i)\,\mathrm{d}\mu_j = \\ &= \mathbb{P}_{t-1}(\mu_i)\int_0^{\mu_i}\mathbb{P}(x_{i_t,t}|\mu_j)\frac{\mathbb{P}_{t-1}(M_{i_t}=\mu_j)}{\int_0^{\mu_i}\mathbb{P}_{t-1}(M_{i_t}=x)\mathrm{d}x}\,\mathrm{d}\mu_j = \\ &= \mathbb{P}_{t-1}(\mu_i)\frac{\int_0^{\mu_i}\mathbb{P}(x_{i_t,t}|\mu_j)\mathbb{P}_{t-1}(M_{i_t}=\mu_j)\mathrm{d}\mu_j}{\int_0^{\mu_i}\mathbb{P}_{t-1}(M_{i_t}=x)\mathrm{d}x}\end{aligned}$$

In the case $i_t < i \Rightarrow \mu_{i_t} \geq \mu_i$ we have that:

$$\mathbb{P}_{t-1}(\mu_j|\mu_i) = \begin{cases} \dfrac{\mathbb{P}_{t-1}(M_{i_t}=\mu_j)}{\int_{\mu_i}^1\mathbb{P}_{t-1}(M_{i_t}=x)\mathrm{d}x} & \mu_j \geq \mu_i \\ 0 & \mu_j < \mu_i \end{cases} \tag{A.22}$$

leading to:

$$\begin{aligned}\mathbb{P}_{t-1}(\mu_i|x_j) &\propto \mathbb{P}_{t-1}(\mu_i)\int_{\mu_j\in[0,1]}\mathbb{P}_{t-1}(M_{i_t}=\mu_j)\mathbb{P}_{t-1}(\mu_j|\mu_i)\,\mathrm{d}\mu_j = \\ &= \mathbb{P}_{i,t-1}(\mu_i)\int_{\mu_i}^{1}\mathbb{P}(x_{i_t,t}|\mu_j)\frac{\mathbb{P}_{t-1}(M_{i_t}=\mu_j)}{\int_{\mu_i}^{1}\mathbb{P}_{t-1}(M_{i_t}=x)\mathrm{d}x}\,\mathrm{d}\mu_j = \\ &= \mathbb{P}_{i,t-1}(\mu_i)\frac{\int_{\mu_i}^{1}\mathbb{P}(x_{i_t,t}|\mu_j)\mathbb{P}_{t-1}(M_{i_t}=\mu_j)\,\mathrm{d}\mu_j}{\int_{\mu_i}^{1}\mathbb{P}_{t-1}(M_{i_t}=x)\mathrm{d}x}.\end{aligned}$$

Summarizing we have:

$$\mathbb{P}_t(\mu_i) \propto \begin{cases} \mathbb{P}_{t-1}(\mu_i)\mathbb{P}_{t-1}(x_{i_t,t}|\mu_i) & i_t = i \\ \mathbb{P}_{t-1}(\mu_i)\dfrac{\int_0^{\mu_i}\mathbb{P}(x_{i_t,t}|\mu_j)\mathbb{P}_{t-1}(M_{i_t}=\mu_j)\mathrm{d}\mu_j}{\int_0^{\mu_i}\mathbb{P}t-1(m_{i_t}=x)\mathrm{d}x} & i_t > i \\ \mathbb{P}_{t-1}(\mu_i)\dfrac{\int_{\mu_i}^{1}\mathbb{P}(x_{i_t,t}|\mu_j)\mathbb{P}_{t-1}(M_{i_t}=\mu_j)\,\mathrm{d}\mu_j}{\int_{\mu_i}^{1}\mathbb{P}_{t-1}(M_{i_t}=x)\mathrm{d}x} & i_t < i \end{cases} \tag{A.23}$$

This scheme does not provide a straightforward closed-form solution in its general formulation. In this work, we consider a non-parametric approximation of the prior by resorting to SMC techniques [82]. Each prior distribution is represented by a finite number $N_p \in \mathbb{N}$ of particles $P_i = \{p_{i1}, \dots p_{iN_p}\}$ with $p_{ih} \in [0,1]$ for every $h \in \{1,\dots,N_p\}$ and their corresponding weights $W_i = \{w_{i1},\dots,w_{iN_p}\}$ with $w_{ih} \in \mathbb{R}^+$ for every $h \in \{1,\dots,N_p\}$. In this case, by also considering $X_i \sim Be(\mu_i)$ which implies that $\mathbb{P}(x_{i_t,t}|\mu_j) = \mu_j^{x_{i_t,t}}(1-\mu_j)^{1-x_{i_t,t}}$, the update scheme in (A.23) becomes:

$$w_{ih} \leftarrow \begin{cases} w_{ih} p_{i_t,h}^{x_{i_t,t}}(1-p_{i_t,h})^{1-x_{i_t,t}} & i_t = i \\ w_{ih} \dfrac{\sum_{h|p_{i_t,h} \leq p_{i,h}} p_{i_t,h}^{x_{i_t,t}}(1-p_{i_t,h})^{1-x_{i_t,t}} w_{i_t,h}}{\sum_{h|p_{i_t,h} \leq p_{i,h}} w_{i_t,h}} & i_t > i \\ w_{ih} \dfrac{\sum_{h|p_{i_t,h} \geq p_{i,h}} p_{i_t,h}^{x_{i_t,t}}(1-p_{i_t,h})^{1-x_{i_t,t}} w_{i_t,h}}{\sum_{h|p_{i_t,h} \geq p_{i,h}} w_{i_t,h}} & i_t < i \end{cases} \tag{A.24}$$

A.2.2 Preliminaries for Theorem 8, Theorem 9 and Theorem 10

Before presenting the proofs of the Theorem 8, Theorem 9 and Theorem 10, we provide a technical lemma and some results which will be used in what follows.

Here we recall the link shown in [83] (and cited in [10]) between Beta and Bernoulli distributions, usually addressed in the literature as the "Beta-Binomial trick".

Lemma 2 ([83])**.** *Let us denote with $F^{Beta}_{a,b}$ the Cumulative Distribution Function (CDF) of a beta distribution Beta(a, b) with parameters a and b and with $F^{B}_{n,\mu}$ the CDF of a random variable with binomial distribution Bi(n, μ) with parameters n and μ. It is true that:*

$$F^{Beta}_{a,b}(y) = 1 - F^{B}_{a+b-1,y}(a-1),$$

Lemma 1. *Consider a random variable B with Beta distribution Beta$(S + 1, T - S + 1)$, where $S := \sum_{s=1}^{T} X_s$ is the sum of $T \in \mathbb{N}$ Bernoulli trials $X_s \sim Be(\mu)$ with same parameter $\mu \in [0, 1]$. Consider a finite integer $\tau \in \mathbb{N}, \tau > T$, a parameter $\varepsilon > \frac{1}{2}$ and:*

$$u_T := \frac{S}{T} + \sqrt{\frac{\varepsilon \log \tau}{T}},$$
$$q_T := Q\left(1 - \frac{1}{\tau}\right),$$

where $Q(\alpha)$ is the α-quantile of the random variable B. We have that $q_T \leq u_T$.

This lemma is used in what follows to bound the number of times a Thompson sample $\theta_{i,t}$ is drawn from a high quantile of the Beta distribution by using a UCB-like bound u_T.

Proof. We here considered the inequalities provided in [74] to bound the quantile of a Beta distribution with the KL-divergence and elaborate over them. Consider the event that the considered variable $B \sim$ Beta$(S + 1, T -$

$S+1)$ is greater than the considered UCB-like bound u_T. We have:

$$\begin{aligned}
\mathbb{P}(B \geq u_T) &= 1 - F^{\text{Beta}}_{S+1,T-S+1}(u_T) && \text{(A.25)}\\
&= F^{\text{B}}_{T+1,u_T}(S) && \text{(A.26)}\\
&= F^{\text{B}}_{T+1,1-u_T}(T-S+1) = \mathbb{P}(Bi_{T+1,1-u_T} > T-S+1) && \text{(A.27)}\\
&\leq \exp\left\{-T \cdot KL\left(\frac{T-S+1}{T+1}, 1-u_T\right)\right\} && \text{(A.28)}\\
&\leq \exp\left\{-2T\left(\frac{T-S+1}{T+1} - 1 + u_T\right)^2\right\} && \text{(A.29)}\\
&= \exp\left\{-2T\left(\frac{T-S+1}{T+1} - 1 + \frac{S}{T} + \sqrt{\frac{\varepsilon \log \tau}{T}}\right)^2\right\} && \text{(A.30)}\\
&\leq \exp\left\{-2T\frac{\varepsilon \log \tau}{T}\right\} = \frac{1}{\tau^{2\varepsilon}}, && \text{(A.31)}
\end{aligned}$$

where $Bi_{n,\mu}$ is a random variable with binomial distribution Bi(n,μ) with parameters n and μ, $KL(\cdot,\cdot)$ is the Kullback-Leibler divergence, the equalities in Equation (A.25) follow from the Lemma 2, Equation (A.27) follow from the properties of the binomial distribution, Equation (A.28) follows from the Sanov inequality, Equation (A.29) follows from the Pinsker inequality.

Considering the quantile $Q\left(1-\frac{1}{\tau}\right)$, we have:

$$\mathbb{P}\left(B \geq q_T\right) = \frac{1}{\tau}.$$

Since for $\varepsilon \geq \frac{1}{2}$ we have $\frac{1}{\tau} \geq \frac{1}{\tau^{2\varepsilon}}$, it follows that $q_T \leq u_T$. □

Here we recall the lemma that has been proven independently in the appendices of [84] (Lemma 1) and [37] (Lemma 4.1).

Lemma 3 ([84, 37]). *Let $A \subset \mathbb{N}$ and $a(t) = \sum_{t'=t-\tau}^{t-1} \mathbb{1}\{t' \in A\}$, then for any positive integer τ and any $s \in \mathbb{N}$ we have:*

$$\sum_{t=1}^{N} \mathbb{E}\left[\mathbb{1}\{t \in A, a(t) \leq s\}\right] \leq s\left\lceil\frac{N}{\tau}\right\rceil.$$

A.2.3 Proof of Theorem 8

Theorem 8. *If policy SW-TS is run over an AC-MAB setting with $X_{i,t} \sim$ Be$(\mu_{i,t})$, for any $\tau \in \mathbb{N}$, the pseudo-regret after N rounds is at most:*

$$\bar{R}_N(\mathfrak{U}) \leq \sum_{i=1}^{K} \Bigg[\tau \Upsilon N^{\alpha} + \\ + \sum_{\phi=1}^{\Upsilon_N} \Delta_{i,\phi} \frac{N_\phi}{\tau} \left(\frac{56 \log \tau}{\Delta_{i,\phi}^2} + \log \tau + 4 + \frac{22}{\log \tau} + \frac{1}{\tau^{\frac{1}{2}}} \right) \Bigg],$$

where Υ and α are defined in Assumption 1 and $\Delta_{i,\phi} := \mu_{i^*,\phi} - \mu_{i,\phi}$ is the difference between the expected reward $\mu_{i^*,\phi}$ of the best arm $a_{i^*_\phi}$ and the expected reward $\mu_{i,\phi}$ of arm a_i. By defining:

$$\Delta_i := \min_{\phi \in \{1,\dots,\Upsilon_N\}} \Delta_{i,\phi} \mathbb{1}\{i \neq i^*_\phi\},$$

for all $i \in \{1,\dots,K\}$, i.e., the minimum over all the phases Φ_ϕ of the difference of the expected rewards $\Delta_{i,\phi}$, the pseudo-regret becomes:

$$\bar{R}_N(\mathfrak{U}) \leq \tau K \Upsilon N^{\alpha} + \frac{N}{\tau} \sum_{i=1}^{K} \left(\frac{56 \log \tau}{\Delta_i^2} + \log \tau + 4 + \frac{22}{\log \tau} + \frac{1}{\tau^{\frac{1}{2}}} \right).$$

Proof. In the proof, we follow the strategy presented in [74] to bound the regret of the classical Thompson Sampling algorithm. We will underline the critical points where we needed to deviate from the original proof to deal with the AC-MAB setting.

Let us define the effective phase $\Phi'_\phi := \{t \in \mathbb{N} \text{ s.t. } b_{\phi-1} + \tau \leq t < b_\phi\}$ and denote with $T_i(\Phi'_\phi) := \sum_{t \in \Phi'_\phi} \mathbb{1}\{i_t = i, i \neq i^*_\phi\}$, i.e., the number of times a suboptimal arm $a_i \neq a_{i^*_\phi}$ has been played in the effective phase Φ'_ϕ. During the generic effective phase Φ'_ϕ we are considering a stationary MAB setting. Moreover, by the definition of effective phase Φ'_ϕ we have:

$$\mathbb{E}\left[T_i(\Phi_\phi)\right] \leq \tau + \mathbb{E}\left[T_i(\Phi'_\phi)\right], \tag{A.32}$$

where we recall that $T_i(\Phi_\phi)$ is the number of times the arm a_i has been pulled during phase Φ_ϕ.

At first, we bound the expected number of times we selected a suboptimal arm in a generic effective phase Φ'_ϕ. We consider two events to bound $\mathbb{E}[T_i(\Phi'_\phi)]$: in the first one the optimal arm $a_{i^*_\phi}$ is underestimated and in the second one the optimal arm $a_{i^*_\phi}$ is not underestimated but the suboptimal arm a_i is played. Hence, we have:

$$\mathbb{E}\left[T_i(\Phi'_\phi)\right] = \sum_{t \in \Phi'_\phi} \mathbb{E}\left[\mathbb{1}\{i_t = i\}\right] \tag{A.33}$$

$$= \sum_{t \in \Phi'_\phi} \left[\mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, i_t = i\right) + \right.$$

$$\left. + \mathbb{P}\left(\theta_{i^*_\phi,t} > \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, i_t = i\right) \right] \tag{A.34}$$

$$\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}\right) +$$

$$+ \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i,t} > \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, i_t = i\right) \tag{A.35}$$

$$\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}\right) +$$

$$+ \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i,t} > \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, i_t = i, \theta_{i,t} < q_{T_{i,t,\tau}}\right) +$$

$$+ \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i,t} \geq q_{T_{i,t,\tau}}\right) \tag{A.36}$$

$$\leq \underbrace{\sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}\right)}_{R_A} +$$

$$+ \underbrace{\sum_{t \in \Phi'_\phi} \mathbb{P}\left(u_{T_{i,t,\tau}} > \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, i_t = i\right)}_{R_B} +$$

$$+ \underbrace{\sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i,t} \geq q_{T_{i,t,\tau}}\right)}_{R_C}, \tag{A.37}$$

where in bounding Equation (A.34) we used the fact that the Thompson sample $\theta_{i_t,t} = \theta_{i,t}$ chosen for the t turn is larger than the one of the optimal arm $\theta_{i^*_\phi,t}$, i.e., $\theta_{i,t} \geq \theta_{i^*_\phi,t}$, $q_{T_{i,t,\tau}}$ is the quantile of order $1 - \frac{1}{\tau}$ of the posterior Beta distribution of the expected value $\mu_{i,t}$ of the arm a_i and we used Lemma 1 to bound the second term in Equation (A.36), by considering the rewards gained by arm a_i, $T = T_{i,t,\tau}$ and $\varepsilon = 2$.

Let us focus on R_A. While in [74] the probability that the optimal arm has been pulled in the past less than t^b times (by properly defining the constant $b \in (0, 1)$) was bounded by a constant (from Proposition 1 in [74]), in this case we cannot resort to that result, since the amount of samples considered in the posterior distribution $\pi_{i,t}$ of the expected reward $\mu_{i,t}$ does not increase indefinitely over time due to the SW approach (we use at most τ samples). Thus, we bound the event that an arm is pulled less than $\bar{n}_A$ times by considering Lemma 3 with $A = \{t | i_t = i\}$, $t \in \Phi'_\phi$ and, consequently $a(t) = T_{i,t,\tau}$. We have:

$$\sum_{t \in \Phi' \phi} \mathbb{E}\left[\mathbb{1}\{i_t = i, T_{i,t,\tau} \leq \bar{n}_A\}\right] \leq \bar{n}_A \left\lceil \frac{N_\phi - \tau}{\tau} \right\rceil \leq \bar{n}_A \frac{N_\phi}{\tau}. \tag{A.38}$$

where $|\Phi'_\phi| = N_\phi - \tau \leq N_\phi$, which holds for all $i \in \{1, \dots, K\}$. Thus, by choosing $\bar{n}_A = \left\lceil \frac{22}{\log \tau} \right\rceil$, we have:

$$R_A = \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}\right) \tag{A.39}$$

$$\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, T_{i^*_\phi,t,\tau} > \bar{n}_A\right) + \\ + \sum_{t \in \Phi'_\phi} \mathbb{P}\left(T_{i^*_\phi,t,\tau} \leq \bar{n}_A\right) \tag{A.40}$$

$$\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, T_{i^*_\phi,t,\tau} > \bar{n}_A\right) + \\ + \sum_{t \in \Phi'_\phi} \mathbb{E}\left[\mathbb{1}\left\{T_{i^*_\phi,t,\tau} \leq \bar{n}_A\right\}\right] \tag{A.41}$$

$$\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, T_{i^*_\phi,t,\tau} > \bar{n}_A\right) + \bar{n}_A \frac{N_\phi}{\tau}, \tag{A.42}$$

where we considered Lemma 3 to bound Equation (A.42).

Let us define:

- $\{U_t\}_{t \in \Phi'_\phi}$ a sequence of i.i.d. uniform random variables over $\Omega = [0, 1]$;
- $S_{i,t,\tau} := \sum_{h=t-\tau+1}^{t} X_{i,h} \mathbb{1}\{i_h = i\}$ the sum of the rewards received by the arm a_i in the last τ rounds (with abuse of notation);
- $\Sigma_{i,t,\tau,s} := \sum_{s=t-\tau+1}^{t-\tau+s} X_{i,h} \mathbb{1}\{i_h = i\}$ the sum of the first s rewards over the last τ rounds of arm a_i.

Recalling that $T_{i,t,\tau} := \sum_{h=\max\{t-\tau+1,1\}}^{t} \mathbb{1}\{i_h = i\}$, we have:

$$\mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, T_{i^*_\phi,t,\tau} > \bar{n}_A\right) \tag{A.43}$$

$$= \mathbb{P}\left(U_t \leq F^{\text{Beta}}_{S_{i^*_\phi,t,\tau}+1, T_{i^*_\phi,t,\tau} - S_{i^*_\phi,t,\tau}+1}\left(\mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}\right), T_{i^*_\phi,t,\tau} > \bar{n}_A\right) \tag{A.44}$$

$$= \mathbb{P}\left(U_t \leq 1 - F^{\text{B}}_{T_{i^*_\phi,t,\tau}+1, \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}}(S_{i^*_\phi,t,\tau}), T_{i^*_\phi,t,\tau} > \bar{n}_A\right) \tag{A.45}$$

$$= \mathbb{P}\left(F^{\text{B}}_{T_{i^*_\phi,t,\tau}+1, \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}}(S_{i^*_\phi,t,\tau}) \leq U_t, T_{i^*_\phi,t,\tau} > \bar{n}_A\right) \tag{A.46}$$

$$\leq \mathbb{P}\left(\exists s \in \{\bar{n}_A, \dots, \tau\} F^{\text{B}}_{s+1, \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{s}}}(\Sigma_{i^*_\phi,t,\tau,s}) \leq U_t\right) \tag{A.47}$$

$$\leq \sum_{s=\bar{n}_A}^{\tau} \mathbb{P}\left(\Sigma_{i^*_\phi,t,\tau,s} \leq (F^{\text{B}})^{-1}_{s+1, \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{s}}}(U_t)\right), \tag{A.48}$$

where to derive Equation (A.45) we considered Lemma 2, to derive Equation (A.46) we used the fact that $U_t \sim 1 - U_t$ and to bound Equation (A.48) we considered a union bound.

Note that:

$$(F^{\text{B}})^{-1}_{s+1, \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{s}}}(U_t) \sim \text{Bi}\left(s + 1, \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{s}}\right) \tag{A.49}$$

and is independent from $\Sigma_{i^*_\phi,t,\tau,s} \sim \text{Bi}(s, \mu_{i^*_\phi,t})$. Similarly to what has been considered in [10], we define, for a chosen s, two i.i.d. sequences of Bernoulli random variables $\{X_{1,l}\}_{l=1}^{s}$ and $\{X_{2,l}\}_{l=1}^{s}$ of size s and $s+1$, respectively:

$$X_{1,l} \sim \text{Be}\left(\mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{s}}\right), \tag{A.50}$$

$$X_{2,l} \sim \text{Be}\left(\mu_{i^*_\phi,t}\right), \tag{A.51}$$

whose summations correspond to the r.h.s. and l.h.s. of the inequality present in the probability in Equation (A.48), respectively. Let $\{Z_l\}_{l=1}^{s}$ be another i.i.d. sequence of random variables, with $Z_l := X_{2,l} - X_{1,l}$ and $\mathbb{E}[Z_l] = \sqrt{\frac{6 \log \tau}{s}}$.[2] We get:

$$\mathbb{P}\left(\Sigma_{i^*_\phi,t,\tau,s} \leq (F^{\mathrm{B}})^{-1}_{s+1,\mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{s}}}(U_t)\right) \tag{A.52}$$

$$= \mathbb{P}\left(\sum_{l=1}^{s} X_{2,l} \leq \sum_{l=1}^{s+1} X_{1,l}\right) \tag{A.53}$$

$$= \mathbb{P}\left(\sum_{l=1}^{s} Z_l \leq X_{1,s+1}\right) \tag{A.54}$$

$$\leq \mathbb{P}\left(\sum_{l=1}^{s} Z_l \leq 1\right) \tag{A.55}$$

$$= \mathbb{P}\left(\sum_{l=1}^{s}\left(Z_l - \sqrt{\frac{6 \log \tau}{s}}\right) \leq -\sum_{l=1}^{s}\sqrt{\frac{6 \log \tau}{s}} + 1\right) \tag{A.56}$$

$$= \mathbb{P}\left(\sum_{l=1}^{s}\left(Z_l - \sqrt{\frac{6 \log \tau}{s}}\right) \leq -\left(\sqrt{6 s \log \tau} - 1\right)\right) \tag{A.57}$$

$$\leq \mathbb{P}\left(\sum_{l=1}^{s}\left(Z_l - \sqrt{\frac{6 \log \tau}{s}}\right) \leq -\sqrt{5 s \log \tau}\right), \tag{A.58}$$

where to bound Equation (A.58) we used the fact that $s > \bar{n}_A \Rightarrow \sqrt{6 s \log \tau} - 1 > \sqrt{5 s \log \tau}$. We apply the Hoeffding's inequality [73] to the bounded martingale difference sequence $\{Z_l\}_{l=1}^{s}$ (having support of measure 2) and

[2] We here assume that $\mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{s}} \geq 0$, i.e., that the sequence $\{X_{1,l}\}_{l=1}^{s}$ is well defined. In the case this condition does not hold we have $R_A = 0$, since the event that the Thompson sample $\theta_{i^*_t,t} < 0$ has zero probability.

we get:

$$\sum_{s=\bar{n}_A}^{\tau} \mathbb{P}\left(\Sigma_{i^*_\phi,t,\tau,s} \leq (F^{\mathrm{B}})^{-1}_{s+1,\mu_{i^*_\phi,t}-\sqrt{\frac{6\log\tau}{s}}}(U_t)\right) \tag{A.59}$$

$$\leq \sum_{s=\bar{n}_A}^{\tau} \exp\left(-\frac{2s(\sqrt{5s\log\tau})^2}{\sum_{h=1}^{s} 2^2}\right) \tag{A.60}$$

$$= \sum_{s=\bar{n}_A}^{\tau} \exp\left(-\frac{(\sqrt{5s\log\tau})^2}{2s}\right) \tag{A.61}$$

$$= \sum_{s=\bar{n}_A}^{\tau} e^{-\frac{5}{2}\log\tau} \tag{A.62}$$

$$\leq \sum_{s=1}^{\tau} \frac{1}{\tau^{\frac{5}{2}}} = \frac{1}{\tau^{\frac{3}{2}}}. \tag{A.63}$$

Finally, we get:

$$R_A = \sum_{t\in\Phi'_\phi} \mathbb{P}\left(\theta_{i^*_\phi,t} \leq \mu_{i^*_\phi,t} - \sqrt{\frac{6\log\tau}{T_{i^*_\phi,t,\tau}}}\right) \tag{A.64}$$

$$\leq \bar{n}_A \frac{N_\phi}{\tau} + \sum_{t\in\Phi'_\phi} \frac{1}{\tau^{\frac{3}{2}}} \tag{A.65}$$

$$\leq \frac{N_\phi}{\tau}\left(\frac{22}{\log\tau}+1\right) + \frac{N_\phi}{\tau^{\frac{3}{2}}} \tag{A.66}$$

$$= \frac{22N_\phi}{\tau\log\tau} + \frac{N_\phi}{\tau} + \frac{N_\phi}{\tau^{\frac{3}{2}}}. \tag{A.67}$$

Let us focus on R_B. Define $\hat{\mu}_{i,t,\tau} := \frac{\sum_{s=t-\tau+1}^{t} X_{i,s}\mathbb{1}\{i_s=i\}}{T_{i,t,\tau}}$, i.e., the estimator of the expected value $\mu_{i,\phi}$ of the rewards of the arm a_i computed over the last τ rounds and choose $\bar{n}_{B*} = \left\lceil \frac{24\log\tau}{\Delta^2_{i,\phi}} \right\rceil$ and $\bar{n}_B = \left\lceil \frac{32\log\tau}{\Delta^2_{i,\phi}} \right\rceil$, where we recall that $\Delta_{i,\phi} := \mu_{i^*_\phi,t} - \mu_{i,t}$ with $t \in \Phi'_\phi$. This choice implies that if $T_{i^*_\phi,t,\tau} > \bar{n}_{B*}$ and $T_{i,t,\tau} > \bar{n}_B$:

$$-\left(2\sqrt{\frac{2\log\tau}{T_{i,t,\tau}}} + \sqrt{\frac{6\log\tau}{T_{i^*_\phi,t,\tau}}}\right) > -\Delta_{i,\phi} \tag{A.68}$$

and thus:

$$
\begin{aligned}
R_B &\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(u_{T_{i,t,\tau}} > \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, i_t = i \right) && \text{(A.69)}\\
&= \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} + \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, i_t = i \right) && \text{(A.70)}\\
&\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} + \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, T_{i^*_\phi,t,\tau} > \bar{n}_{B*}, T_{i,t,\tau} > \bar{n}_B \right) +\\
&\quad + \sum_{t \in \Phi'_\phi} \mathbb{P}\left(T_{i^*_\phi,t,\tau} \leq \bar{n}_{B*} \right) + \sum_{t \in \Phi'_\phi} \mathbb{P}\left(T_{i,t,\tau} \leq \bar{n}_B \right) && \text{(A.71)}\\
&\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} + \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i^*_\phi,t} - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, T_{i^*_\phi,t,\tau} > \bar{n}_{B*}, T_{i,t,\tau} > \bar{n}_B \right) +\\
&\quad + \bar{n}_{B*} \frac{N_\phi}{\tau} + \bar{n}_B \frac{N_\phi}{\tau} && \text{(A.72)}\\
&\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t} + \underbrace{\mu_{i^*_\phi,t} - \mu_{i,t}}_{=\Delta_{i,\phi}} \underbrace{- \left(2\sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} + \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}} \right)}_{>-\Delta_{i,\phi}} \right) +\\
&\quad + \frac{N_\phi}{\tau} \left(\frac{56 \log \tau}{\Delta^2_{i,\phi}} + 2 \right) && \text{(A.73)}\\
&\leq \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t} \right) + \frac{N_\phi}{\tau} \frac{56 \log \tau}{\Delta^2_{i,\phi}} + \frac{2 N_\phi}{\tau}, && \text{(A.74)}
\end{aligned}
$$

where Equation (A.70) is thanks to the definition of $u_{T_{i,t,\tau}}$.

By considering Corollary 21 in [52] we have for all $\eta > 0$:

$$
\begin{aligned}
&\sum_{t \in \Phi'_\phi} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t} \right)\\
&\qquad \leq \sum_{t \in \Phi'_\phi} \frac{\log \tau}{\log(1+\eta)} \exp\left(-12 \log \tau \left(1 - \frac{\eta^2}{16} \right) \right) && \text{(A.75)}
\end{aligned}
$$

and by considering $\eta = 4\sqrt{1 - \frac{1}{12}}$

$$
\sum_{t \in \Phi'_\phi} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t} \right) \leq \sum_{t \in \Phi'_\phi} \frac{\log \tau}{\tau} \leq \frac{N_\phi \log \tau}{\tau}. \tag{A.76}
$$

Summarizing we have:

$$R_B \leq \frac{N_\phi}{\tau} \frac{56 \log \tau}{\Delta_{i,\phi}^2} + \frac{2N_\phi}{\tau} + \frac{N_\phi \log \tau}{\tau}. \tag{A.77}$$

Let us focus on R_C. The R_C term is upper bounded by:

$$R_C = \sum_{t \in \Phi'_\phi} \mathbb{P}\left(\theta_{i,t} \geq q_{T_{i,t},\tau}\right) = \sum_{t \in \Phi'_\phi} \frac{1}{\tau} \leq \frac{N_\phi}{\tau}. \tag{A.78}$$

Pseudo-regret. Since $\sum_{\phi=1}^{\Upsilon_N} N_\phi = N$ and recalling that if $t \in \Phi_\phi \supset \Phi'_\phi$ we have $\mu_{i,t} = \mu_{i,\phi}$, the total regret over all the phases becomes:

$$\bar{R}_N(\mathfrak{U}) = \mathbb{E}\left[\sum_{t=1}^{N} (\mu_{i^*,t} - \mu_{i_t,t})\right] = \sum_{\phi=1}^{\Upsilon_N} \mu_{i^*,\phi} N_\phi - \mathbb{E}\left[\sum_{t=1}^{N} \mu_{i_t,t}\right] \tag{A.79}$$

$$= \sum_{\phi=1}^{\Upsilon_N} \left(\mu_{i^*,\phi} N_\phi - \mathbb{E}\left[\sum_{t \in \Phi_\phi} \mu_{i_t,t}\right]\right) = \sum_{\phi=1}^{\Upsilon_N} \left(\mu_{i^*,\phi} N_\phi - \sum_{i=1}^{K} \mu_{i,\phi} \mathbb{E}[T_i(\Phi_\phi)]\right) \tag{A.80}$$

$$= \sum_{\phi=1}^{\Upsilon_N} \left(\sum_{i=1}^{K} (\mu_{i^*,\phi} - \mu_{i,\phi}) \mathbb{E}[T_i(\Phi_\phi)]\right) = \sum_{i=1}^{K} \left(\sum_{\phi=1}^{\Upsilon_N} (\mu_{i^*,\phi} - \mu_{i,\phi}) \mathbb{E}[T_i(\Phi_\phi)]\right) \tag{A.81}$$

$$= \sum_{i=1}^{K} \left(\sum_{\phi=1}^{\Upsilon_N} \Delta_{i,\phi} \mathbb{E}[T_i(\Phi_\phi)]\right) \leq \sum_{i=1}^{K} \left[\sum_{\phi=1}^{\Upsilon_N} \Delta_{i,\phi} \left(\tau + \mathbb{E}[T_i(\Phi'_\phi)]\right)\right] \tag{A.82}$$

$$\leq \sum_{i=1}^{K} \left[\tau \Upsilon_N + \sum_{\phi=1}^{\Upsilon_N} \Delta_{i,\phi} \left(R_A + R_B + R_C\right)\right] \tag{A.83}$$

$$\leq \sum_{i=1}^{K} \left[\tau \Upsilon N^\alpha + \sum_{\phi=1}^{\Upsilon_N} \Delta_{i,\phi} \left(\frac{22 N_\phi}{\tau \log \tau} + \frac{N_\phi}{\tau} + \frac{N_\phi}{\tau^{\frac{3}{2}}} + \frac{N_\phi}{\tau} \frac{56 \log \tau}{\Delta_{i,\phi}^2} + \frac{2N_\phi}{\tau} + \frac{N_\phi \log \tau}{\tau} + \frac{N_\phi}{\tau}\right)\right] \tag{A.84}$$

$$\leq \sum_{i=1}^{K} \left[\tau \Upsilon N^\alpha + \sum_{\phi=1}^{\Upsilon_N} \Delta_{i,\phi} \left(\frac{22 N_\phi}{\tau \log \tau} + \frac{N_\phi}{\tau^{\frac{3}{2}}} + \frac{N_\phi}{\tau} \frac{56 \log \tau}{\Delta_{i,\phi}^2} + \frac{N_\phi \log \tau}{\tau} + \frac{4N_\phi}{\tau}\right)\right] \tag{A.85}$$

$$\leq \sum_{i=1}^{K} \left[\tau \Upsilon N^\alpha + \sum_{\phi=1}^{\Upsilon_N} \Delta_{i,\phi} \frac{N_\phi}{\tau} \left(\frac{56 \log \tau}{\Delta_{i,\phi}^2} + \log \tau + 4 + \frac{22}{\log \tau} + \frac{1}{\tau^{\frac{1}{2}}}\right)\right] \tag{A.86}$$

where Υ_N is the number of breakpoints before N.

By defining:

$$\Delta_i := \min_{\phi \in \{1,\dots,\Upsilon_N\}} \Delta_{i,\phi} \mathbb{1}\{i \neq i^*_\phi\} \quad \forall i \in \{1, \dots, K\}, \tag{A.87}$$

i.e., the minimum over all the phases Φ_ϕ in which the arm a_i is not optimal of the difference of the expected reward $\mu_{i^*_\phi,\phi}$ of the best arm $a_{i^*_\phi}$ and the expected reward $\mu_{i,\phi}$ of arm a_i, the regret becomes:

$$\bar{R}_N(\mathfrak{U}) \leq \tau K \Upsilon N^\alpha + \frac{N}{\tau} \sum_{i=1}^{K} \left(\frac{56 \log \tau}{\Delta_i^2} + \log \tau + 4 + \frac{22}{\log \tau} + \frac{1}{\tau^{\frac{1}{2}}} \right), \tag{A.88}$$

which concludes the proof. □

A.2.4 Proof of Theorem 9

Theorem 9. *If policy SW-TS is run over a SC-MAB setting with $X_{i,t} \sim Be(\mu_{i,t})$, Lipschitz constant $\sigma > 0$ and there exists $\Delta_0 \in (0,1)$ as in Assumption 3, for any $\tau \in \mathbb{N}$ s.t. $2\sigma\tau < \Delta \leq 3\sigma\tau \leq \Delta_0$, the expected pseudo-regret after N rounds is at most:*

$$\begin{aligned}\bar{R}_N(\mathfrak{U}) \leq & \left(3\sigma\mathcal{F}N^{\beta} + 1\right)\tau \\ & + \frac{NK}{\tau}\left[\frac{56\log\tau}{(\Delta - 2\sigma\tau)^2} + \log\tau + 4 + \frac{22}{\log\tau} + \frac{1}{\tau^{\frac{1}{2}}}\right].\end{aligned}$$

Proof. Let us consider:

- $\Phi_{\Delta,N} := \{t \in \{1,\dots,N\}$ s.t. $\exists i \neq j, |\mu_{i,t} - \mu_{j,t}| < \Delta\}$, i.e., the set of the rounds in which there exist two arms with expected values differing less than Δ;
- $\Phi_{\Delta^C,N} := \{\tau,\dots,N\} \setminus \Phi_{\Delta,N}$, i.e., the set of the rounds $t \geq \tau$, in which the expected rewards of the arms are well separated ($|\mu_{i,t} - \mu_{j,t}| > \Delta, \forall i \neq j\}$);
- $T_i(\Phi_{\Delta,N}) := \sum_{t \in \Phi_{\Delta,N}} \mathbb{1}\{i_t = i, i \neq i_t^*\}$, i.e., the amount of rounds the arm a_i has been played when it was not the optimal one during rounds $t \in \Phi_{\Delta,N}$;
- $T_i(\Phi_{\Delta^C,N}) := \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{1}\{i_t = i, i \neq i_t^*\}$, i.e., the amount of rounds the arm a_i has been played when it was not the optimal one during rounds $t \in \Phi_{\Delta^C,N}$.

By considering Δ s.t. $2\sigma\tau \leq \Delta \leq 3\sigma\tau$, we have that:

$$\bar{R}_N(\mathfrak{U}) = \mathbb{E}\left[\sum_{t=1}^{N}\left(\mu_{i_t^*,t} - \mu_{i_t,t}\right)\right] \tag{A.89}$$

$$\leq \sum_{t=1}^{N} \mathbb{E}\left[\mathbb{1}\{i_t = i, i \neq i_t^*\}\right] = \sum_{i=1}^{K}\sum_{t=1}^{N} \mathbb{E}\left[\mathbb{1}\{i_t = i, i \neq i_t^*\}\right] \tag{A.90}$$

$$\leq \tau + \sum_{i=1}^{K} \mathbb{E}[T_i(\Phi_{\Delta,N})] + \sum_{i=1}^{K} \mathbb{E}[T_i(\Phi_{\Delta^C,N})]. \tag{A.91}$$

While the second term in Equation (A.91) $\mathbb{E}[T_i(\Phi_{\Delta,N})]$ is bounded by Assumption 3, we need to bound with a more complex procedure the third one $\mathbb{E}[T_i(\Phi_{\Delta^C,N})]$. Similarly to what has been considered in Theorem 8, we follow the line delineated in [74], where we have the further technical difficulty that the reward distributions are varying at every round.[3] We consider two events: in the first one the optimal arm $a_{i_t^*}$ is underestimated; in the second one the optimal arm $a_{i_t^*}$ is not underestimated, but the suboptimal arm a_i is played. Hence, we have:

$$\mathbb{E}\left[T_i(\Phi_{\Delta^C,N})\right] \tag{A.92}$$

$$\leq \sum_{t\in\Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i_t^*,t} \leq \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}\right) +$$

$$+ \sum_{t\in\Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i,t} > \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}, i_t = i\right) \tag{A.93}$$

$$\leq \sum_{t\in\Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i_t^*,t} \leq \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}\right) +$$

$$+ \sum_{t\in\Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i,t} > \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}, i_t = i, \theta_{i,t} \leq q_{T_{i,t,\tau}}\right) +$$

$$+ \sum_{t\in\Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i,t} \geq q_{T_{i,t,\tau}}\right) \tag{A.94}$$

$$\leq \underbrace{\sum_{t\in\Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i_t^*,t} \leq \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}\right) +}_{R_A}$$

$$+ \underbrace{\sum_{t\in\Phi_{\Delta^C,N}} \mathbb{P}\left(u_{T_{i,t,\tau}} > \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}\right)}_{R_B}$$

$$+ \underbrace{\sum_{t\in\Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i,t} \geq q_{T_{i,t,\tau}}\right)}_{R_C}, \tag{A.95}$$

where to bound the expression in Equation (A.95) we considered Lemma 1 over the rewards of the arm a_i, $T = T_{i,t,\tau}$ and $\varepsilon = 2$.

Let us focus on R_A. By considering Lemma 3 and defining $\bar{n}_A = \left\lceil \frac{22}{\log\tau} \right\rceil$,

[3] For sake of concision we will omit those derivations which are equal to those considered in Theorem 8.

we have:

$$\begin{aligned}
R_A &= \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i_t^*,t} \le \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}\right) \\
&\le \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i_t^*,t} \le \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}, T_{i_t^*,t,\tau} > \bar{n}_A\right) + \\
&\quad + \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(T_{i_t^*,t,\tau} \le \bar{n}_A\right) \\
&\le \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i_t^*,t} \le \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}, T_{i_t^*,t,\tau} > \bar{n}_A\right) + \bar{n}_A \left\lceil \frac{N_\Delta}{\tau} \right\rceil \\
&\le \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i_t^*,t} \le \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}, T_{i_t^*,t,\tau} > \bar{n}_A\right) + \bar{n}_A \frac{N}{\tau}
\end{aligned}$$

where $N_\Delta := |\Phi_{\Delta^C,N}| \le N - \tau$ and $|\cdot|$ denotes the cardinality operator.

While in the proof of Theorem 8 the expected values of the considered rewards were constant over the considered τ rounds, in this case we don not have such an assumption and, thus, we need to define a set of auxiliary variables whose mean is constant over the last τ rounds and whose value is smaller than the one of the optimal arm. Over this newly defined variables we can use the Lemma 2 to transform the Beta distribution into a Binomial one.

Let us define:

- $\{U_t\}_{t \in \Phi_{\Delta^C,N}}$ as a sequence of i.i.d. uniform random variables over $\Omega = [0, 1]$;
- $S_{i,t,\tau} := \sum_{s=t-\tau+1}^{t} \mathbb{1}\{i_s = i\} X_{i,s}$, i.e., the amount of successes of arm a_i at round t in the previous τ rounds (with abuse of notation);
- $\tilde{X}_{i,s} := X_{i,s} + \mu_{i,t} - \mu_{i,s} - \sigma\tau, \forall s \in \{t - \tau + 1, t\}$, i.e., a set of auxiliary variables having $\tilde{X}_{i,s} \le X_{i,s}$ (since $|\mu_{i,t} - \mu_{i,s}| \le \sigma\tau$) and $\underline{\mu}_{i,t} := \mathbb{E}[\tilde{X}_{i,s}] = \mu_{i,t} - \sigma\tau$;
- $\underline{S}_{i,t,\tau} := \sum_{s=t-\tau+1}^{t} \mathbb{1}\{i_s = i\} \tilde{X}_{i,s}$, the amount of successes of an arm $\underline{a}_i$ having rewards $\tilde{X}_{i,s}$ at round s in the rounds $\{t - \tau + 1, \dots, t\}$;
- $\Sigma_{i,t,\tau,s} := \sum_{h=t-\tau+1}^{t-\tau+s} \mathbb{1}\{i_h = i\} \tilde{X}_{i,h}$, i.e., the sum of the random variables $\tilde{X}_{i,t-\tau+1}, \dots, \tilde{X}_{i,t-\tau+s}$.

Note that if we consider an arm $\underline{a}_{i_t^*}$ having expected value $\underline{\mu}_{i_t^*,t}$, it would still be the optimal one, since we are in the rounds $t \in \Phi_{\Delta^C,N}$. Hence, we have:

$$\mathbb{P}\left(\theta_{i_t^*,t} \leq \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}, T_{i_t^*,t,\tau} > \bar{n}_A\right) \tag{A.96}$$

$$= \mathbb{P}\left(U_t \leq F^{\text{Beta}}_{S_{i_t^*,t,\tau}+1, T_{i_t^*,t,\tau}-S_{i_t^*,t,\tau}+1}\left(\mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}\right),\right.$$
$$\left., T_{i_t^*,t,\tau} > \bar{n}_A\right) \tag{A.97}$$

$$= \mathbb{P}\left(U_t \leq 1 - F^{\text{B}}_{T_{i_t^*,t,\tau}+1, \mu_{i_t^*,t}-\sigma\tau-\sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}}(S_{i_t^*,t,\tau}), T_{i_t^*,t,\tau} > \bar{n}_A\right) \tag{A.98}$$

$$= \mathbb{P}\left(F^{\text{B}}_{T_{i_t^*,t,\tau}+1, \mu_{i_t^*,t}-\sigma\tau-\sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}}(S_{i_t^*,t,\tau}) \leq U_t, T_{i_t^*,t,\tau} > \bar{n}_A\right) \tag{A.99}$$

$$\leq \mathbb{P}\left(F^{\text{B}}_{T_{i_t^*,t,\tau}+1, \underline{\mu}_{i_t^*,t}-\sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}}(\underline{S}_{i_t^*,t,\tau}) \leq U_t, T_{i_t^*,t,\tau} > \bar{n}_A\right) \tag{A.100}$$

$$\leq \mathbb{P}\left(\exists s \in \{\bar{n}_A, \dots, \tau\} \text{ s.t. } F^{\text{B}}_{s+1, \underline{\mu}_{i_t^*,t}-\sqrt{\frac{6\log\tau}{s}}}(\Sigma_{i_t^*,t,\tau,s}) \leq U_t\right) \tag{A.101}$$

$$= \sum_{s=\bar{n}_A}^{\tau} \mathbb{P}\left(\Sigma_{i_t^*,t,\tau,s} \leq (F^{\text{B}})^{-1}_{s+1, \underline{\mu}_{i_t^*,t}-\sqrt{\frac{6\log\tau}{s}}}(U_t)\right), \tag{A.102}$$

where to derive Equation (A.98) we considered Lemma 2, Equation (A.99) follows from $U_t \sim 1 - U_t$ and we bound Equation (A.100) by the fact that $S_{i,t,\tau} \geq \underline{S}_{i,t,\tau}, \forall i$, which follows from the definition of $\underline{S}_{i,t,\tau}$.

Note that:

$$(F^{\text{B}})^{-1}_{s+1, \underline{\mu}_{i_t^*,t}-\sqrt{\frac{6\log\tau}{s}}}(U_t) \sim \text{Bi}\left(s+1, \underline{\mu}_{i_t^*,t} - \sqrt{\frac{6\log\tau}{s}}\right) \tag{A.103}$$

and is independent from $\Sigma_{i_t^*,t,\tau,s} \sim \text{Bi}(s, \underline{\mu}_{i_t^*,t})$. Consider, for a chosen s, two i.i.d. sequences of random variables $\{X_{1,l}\}_{l=1}^{s}$ and $\{X_{2,l}\}_{l=1}^{s}$ of size s

and $s+1$, respectively:

$$X_{1,l} \sim \mathrm{Be}\left(\underline{\mu}_{i_t^*,t} - \sqrt{\frac{6 \log \tau}{s}}\right), \tag{A.104}$$

$$X_{2,l} \sim \mathrm{D}\left(\underline{\mu}_{i_t^*,t}\right), \tag{A.105}$$

whose summations correspond to the r.h.s. and l.h.s. of the inequality present in the probability in Equation (A.102), respectively. In equation (A.104) we denoted with $\mathrm{Be}(\mu)$ a Bernoulli distribution with mean μ and in Equation (A.105) we denoted with D a discrete distribution defined over $\Omega = \{1 + \mu_{i_t^*,t} - \mu_{i_t^*,s} - \sigma\tau, \mu_{i_t^*,t} - \mu_{i_t^*,s} - \sigma\tau\}$ and expected value equal to $\underline{\mu}_{i_t^*,t}$. Let $\{Z_l\}_{l=1}^s$ be another i.i.d. sequence of random variables, with $Z_l := X_{2,l} - X_{1,l}$, having support of measure 2 and $\mathbb{E}[Z_l] = \sqrt{\frac{6 \log \tau}{s}}$.[4] We get:

$$\mathbb{P}\left(\Sigma_{i_t^*,t,\tau,s} \leq (F^{\mathrm{B}})^{-1}_{s+1,\underline{\mu}_{i_t^*,t}-\sqrt{\frac{6\log\tau}{s}}}(U_t)\right) \tag{A.106}$$

$$= \mathbb{P}\left(\sum_{l=1}^{s} X_{2,l} \leq \sum_{l=1}^{s+1} X_{1,l}\right) = \mathbb{P}\left(\sum_{l=1}^{s} Z_l \leq X_{1,s+1}\right) \tag{A.107}$$

$$\leq \mathbb{P}\left(\sum_{l=1}^{s} Z_l \leq 1\right) \tag{A.108}$$

$$= \mathbb{P}\left(\sum_{l=1}^{s}\left(Z_l - \sqrt{\frac{6\log\tau}{s}}\right) \leq -\sum_{l=1}^{s}\sqrt{\frac{6\log\tau}{s}} + 1\right) \tag{A.109}$$

$$= \mathbb{P}\left(\sum_{l=1}^{s}\left(Z_l - \sqrt{\frac{6\log\tau}{s}}\right) \leq -\left(\sqrt{6s\log\tau} - 1\right)\right) \tag{A.110}$$

$$\leq \mathbb{P}\left(\sum_{l=1}^{s}\left(Z_l - \sqrt{\frac{6\log\tau}{s}}\right) \leq -\sqrt{5s\log\tau}\right), \tag{A.111}$$

where we used the fact that $s > \bar{n}_A \Rightarrow \sqrt{6s\log\tau} - 1 > \sqrt{5s\log\tau}$. We apply the Hoeffding's inequality to the bounded martingale difference sequence

[4] Similarly to what has been done in Theorem 8, we here consider only the case in which the sequence $\{X_{1,l}\}_{l=1}^s$ is well defined.

$\{Z_l\}_{l=1}^{s}$ and we get:

$$\sum_{s=\bar{n}_A}^{\tau} \mathbb{P}\left(\Sigma_{i_t^*,t,\tau,s} \leq (F^{\mathrm{B}})^{-1}_{s+1,\underline{\mu}_{i_t^*,t}-\sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}}(U_t)\right) \tag{A.112}$$

$$\leq \sum_{s=\bar{n}_A}^{\tau} \exp\left(-2\frac{(\sqrt{5s\log\tau})^2}{4s}\right) = \sum_{s=\bar{n}_A}^{\tau} e^{-\frac{5}{2}\log\tau} \leq \sum_{s=1}^{\tau} \frac{1}{\tau^{\frac{5}{2}}} = \frac{1}{\tau^{\frac{3}{2}}}. \tag{A.113}$$

Finally, we get:

$$\begin{aligned}
R_A &= \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i_t^*,t} \leq \mu_{i_t^*,t} - \sigma\tau - \sqrt{\frac{6\log\tau}{T_{i_t^*,t,\tau}}}\right) \\
&\leq \bar{n}_A \frac{N}{\tau} + \sum_{t \in \Phi_{\Delta^C,N}} \frac{1}{\tau^{\frac{3}{2}}} \\
&\leq \frac{22N}{\tau\log\tau} + \frac{N}{\tau} + \frac{N}{\tau^{\frac{3}{2}}}.
\end{aligned} \tag{A.114}$$

Let us focus on R_B. Let us define $\hat{\mu}_{i,t,\tau} := \frac{\sum_{s=t-\tau+1}^{t} X_{i,s}\mathbb{1}\{i_s=i\}}{T_{i,t,\tau}}$, i.e., the estimator of the expected value of the rewards of the arm a_i computed over the last τ rounds and $\mu_{i,t,\tau} := \frac{\sum_{s=t-\tau+1}^{t} \mu_{i,s}\mathbb{1}\{i_s=i\}}{T_{i,t,\tau}}$, the expected value of the rewards of the arm a_i computed over the last τ rounds. Note that $-\mu_{i,t,\tau} \geq -\mu_{i,t} - \sigma\tau$ due to Assumption 2.

We can rewrite term R_B and apply Lemma 3 with $\bar{n}_{B*} = \frac{24\log\tau}{(\Delta-2\sigma\tau)^2}$ and

$\bar{n}_B = \frac{32 \log \tau}{(\Delta - 2\sigma\tau)^2}$:

$$R_B = \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(u_{T_{i,t,\tau}} > \mu_{i^*_t,t} - \sigma\tau - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, i_t = i \right) \tag{A.115}$$

$$= \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} + \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i^*_t,t} - \sigma\tau - \sqrt{\frac{6 \log \tau}{T_{i^*_t,t,\tau}}}, i_t = i \right) \tag{A.116}$$

$$\begin{aligned} \leq & \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} + \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i^*_t,t} - \sigma\tau - \sqrt{\frac{6 \log \tau}{T_{i^*_\phi,t,\tau}}}, T_{i^*_t,t,\tau} > \bar{n}_{B*}, T_{i,t,\tau} > \bar{n}_B \right) + \\ & + \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(T_{i^*_t,t,\tau} \leq \bar{n}_{B*} \right) + \sum_{t \in N_\Delta} \mathbb{P}\left(T_{i,t,\tau} \leq \bar{n}_B \right) \end{aligned} \tag{A.117}$$

$$\begin{aligned} \leq & \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} + \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i^*_\phi,t} - \sigma\tau - \sqrt{\frac{6 \log \tau}{T_{i^*_t,t,\tau}}}, T_{i^*_t,t,\tau} > \bar{n}_{B*}, T_{i,t,\tau} > \bar{n}_B \right) + \\ & + \bar{n}_{B*} \left\lceil \frac{N_\Delta}{\tau} \right\rceil + \bar{n}_B \left\lceil \frac{N_\Delta}{\tau} \right\rceil \end{aligned} \tag{A.118}$$

$$\begin{aligned} = & \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t,\tau} + \mu_{i^*_t,t} - \mu_{i,t,\tau} - \sigma\tau - 2\sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} - \sqrt{\frac{6 \log \tau}{T_{i^*_t,t,\tau}}} \right) + \\ & + \frac{N}{\tau} \left[\frac{56 \log \tau}{(\Delta - 2\sigma\tau)^2} + 2 \right] \end{aligned} \tag{A.119}$$

$$\begin{aligned} \leq & \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t,\tau} + \mu_{i^*_t,t} - \mu_{i,t} - \sigma\tau - \sigma\tau - 2\sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} - \sqrt{\frac{6 \log \tau}{T_{i^*_t,t,\tau}}} \right) + \\ & + \frac{N}{\tau} \frac{56 \log \tau}{(\Delta - 2\sigma\tau)^2} + 2\frac{N}{\tau} \end{aligned} \tag{A.120}$$

$$\begin{aligned} \leq & \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t,\tau} + \Delta_{i,t} - 2\sigma\tau \underbrace{- \left(2\sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} + \sqrt{\frac{6 \log \tau}{T_{i^*_t,t,\tau}}} \right)}_{\geq -(\Delta - 2\sigma\tau)} \right) + \\ & + \frac{N}{\tau} \frac{56 \log \tau}{(\Delta - 2\sigma\tau)^2} + 2\frac{N}{\tau} \end{aligned} \tag{A.121}$$

$$\leq \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t,\tau} \right) + \frac{N}{\tau} \frac{56 \log \tau}{(\Delta - 2\sigma\tau)^2} + 2\frac{N}{\tau}, \tag{A.122}$$

where to bound Equation (A.121) we considered that $\Delta_{i,t} > \Delta \forall i$, $\forall t \in \Phi_{\Delta^C,N}$.

By considering Corollary 21 in [52] we have for all $\eta > 0$:

$$\sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t,\tau} \right) \\ \leq \sum_{t \in \Phi_{\Delta^C,N}} \frac{\log \tau}{\log(1+\eta)} \exp\left(-12 \log \tau \left(1 - \frac{\eta^2}{16} \right) \right)$$

thus by considering $\eta = 4\sqrt{1 - \frac{1}{12}}$ we have:

$$\sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\hat{\mu}_{i,t,\tau} - \sqrt{\frac{2 \log \tau}{T_{i,t,\tau}}} > \mu_{i,t,\tau} \right) \\ \leq \sum_{t \in \Phi_{\Delta^C,N}} \frac{\log \tau}{\tau} = \frac{N_\Delta \log \tau}{\tau} \leq \frac{N \log \tau}{\tau}.$$

Thus, summarizing:

$$R_B \leq \frac{N}{\tau} \frac{56 \log \tau}{(\Delta - 2\sigma\tau)^2} + 2\frac{N}{\tau} + \frac{N \log \tau}{\tau}. \tag{A.123}$$

Let us focus on R_C The R_C term is upper bounded by:

$$R_C = \sum_{t \in \Phi_{\Delta^C,N}} \mathbb{P}\left(\theta_{i,t} \geq q_{T_{i,t,\tau}} \right) = \sum_{t \in \Phi_{\Delta^C,N}} \frac{1}{\tau} \leq \frac{N}{\tau}.$$

Pseudo-regret Summing all the derived bounds the pseudo-regret becomes:

$$\bar{R}_N(\mathfrak{U}) = \mathbb{E}\left[\sum_{t=1}^{N} \left(\mu_{i_t^*,t} - \mu_{i_t,t} \right) \right] \tag{A.124}$$

$$\leq \tau + \sum_{i=1}^{K} \left(\mathbb{E}[T_i(\Phi_{\Delta,N})] + \mathbb{E}[T_i(\Phi_{\Delta^C,N})] \right) \tag{A.125}$$

$$\leq \tau + |\Phi_{\Delta,N}| + K\left(R_A + R_B + R_C \right) \tag{A.126}$$

$$= \tau + \mathcal{F} \Delta N^\beta + \tag{A.127}$$

$$+ K \left(\frac{22N}{\tau \log \tau} + \frac{N}{\tau} + \frac{N}{\tau^{\frac{3}{2}}} + \frac{N}{\tau} \frac{56 \log \tau}{(\Delta - 2\sigma\tau)^2} + 2\frac{N}{\tau} + \frac{N \log \tau}{\tau} + \frac{N}{\tau} \right) \tag{A.128}$$

$$\leq \left(3\sigma \mathcal{F} N^\beta + 1 \right) \tau + \frac{NK}{\tau} \left[\frac{56 \log \tau}{(\Delta - 2\sigma\tau)^2} + \log \tau + 4 + \frac{22}{\log \tau} + \frac{1}{\tau^{\frac{1}{2}}} \right], \tag{A.129}$$

where we considered that $\sum_{i=1}^{K} \mathbb{E}[T_i(\Phi_{\Delta,N})] \leq |\Phi_{\Delta,N}| \leq \mathcal{F}\Delta N^{\beta}$ by definition, which concludes the proof. □

A.2.5 Proof of Theorem 10

Theorem 10. *If policy SW-TS is run over an ASC-MAB setting with $X_{i,t} \sim Be(\mu_{i,t})$, Lipschitz constant $\sigma > 0$ as in Assumption 4 and there exists $\Delta_0 \in (0,1)$ as in Assumption 3, for any $\tau \in \mathbb{N}$ s.t. $2\sigma\tau < \Delta \leq 3\sigma\tau \leq \Delta_0$, the expected pseudo-regret after N rounds is at most:*

$$\bar{R}_N(\mathfrak{U}) \leq \left(3\sigma\mathcal{F}N^{\beta} + \Upsilon N^{\alpha}\right)\tau \\ + \frac{NK}{\tau}\left[\frac{56\log\tau}{(\Delta - 2\sigma\tau)^2} + \log\tau + 4 + \frac{22}{\log\tau} + \frac{1}{\tau^{\frac{1}{2}}}\right],$$

where Υ and α are defined in Assumption 1.

Proof. Let us consider:

- $\Phi_{\Delta,N} := \{t \in \{1,\dots,N\}$ s.t. $\exists i \neq j, |\mu_{i,t} - \mu_{j,t}| < \Delta\}$, i.e., the set of the rounds in which there exist two arms with expected values differing less than Δ;

- $\Phi_{\Delta^C,N} := \{\tau,\dots,N\} \setminus \Phi_{\Delta,N}$, i.e., the set of the rounds $t \geq \tau$, in which the expected rewards of the arms are well separated ($|\mu_{i,t} - \mu_{j,t}| > \Delta, \forall i \neq j\}$);

- $\Phi_{\Delta^C,\phi} := \{b_{\phi-1},\dots,b_{\phi}\} \setminus \Phi_{\Delta,N}$, i.e., the set of the rounds of phase Φ_ϕ, in which the expected rewards of the arms are well separated;

- $\Phi'_{\Delta^C,\phi} := \{b_{\phi-1}+\tau,\dots,b_{\phi}\} \setminus \Phi_{\Delta,N}$, i.e., the set of the rounds of phase Φ_ϕ discarding the first τ ones, in which the expected rewards of the arms are well separated;

- $T_i(\Phi) := \sum_{t\in\Phi} \mathbb{1}\{i_t = i, i \neq i^*_t\}$, i.e., the amount of rounds the arm a_i has been played when it was not the optimal one during rounds $t \in \Phi$.

By considering Δ s.t. $2\sigma\tau \leq \Delta \leq 3\sigma\tau$, we have that:

$$\begin{aligned}
\bar{R}_N(\mathfrak{U}) &= \mathbb{E}\left[\sum_{t=1}^{N}\left(\mu_{i_t^*,t} - \mu_{i_t,t}\right)\right] && \text{(A.130)}\\
&\leq \sum_{t=1}^{N}\mathbb{E}\left[\mathbb{1}\{i_t = i, i \neq i_t^*\}\right] = \sum_{i=1}^{K}\sum_{t=1}^{N}\mathbb{E}\left[\mathbb{1}\{i_t = i, i \neq i_t^*\}\right] && \text{(A.131)}\\
&= \sum_{i=1}^{K}\mathbb{E}[T_i(\Phi_{\Delta,N})] + \sum_{i=1}^{K}\mathbb{E}[T_i(\Phi_{\Delta^C,N})] && \text{(A.132)}\\
&= \sum_{i=1}^{K}\mathbb{E}[T_i(\Phi_{\Delta,N})] + \sum_{\phi=1}^{\Upsilon_N}\sum_{i=1}^{K}\mathbb{E}[T_i(\Phi_{\Delta^C,\phi})] && \text{(A.133)}\\
&= \sum_{i=1}^{K}\mathbb{E}[T_i(\Phi_{\Delta,N})] + \sum_{\phi=1}^{\Upsilon_N}\left(\tau + \sum_{i=1}^{K}\mathbb{E}[T_i(\Phi'_{\Delta^C,\phi})]\right) && \text{(A.134)}\\
&= \sum_{i=1}^{K}\mathbb{E}[T_i(\Phi_{\Delta,N})] + \tau\Upsilon_N + \sum_{\phi=1}^{\Upsilon_N}\sum_{i=1}^{K}\mathbb{E}[T_i(\Phi'_{\Delta^C,\phi})]. && \text{(A.135)}
\end{aligned}$$

The first term in Equation (A.135) $\mathbb{E}[T_i(\Phi_{\Delta,N})]$ is bounded by Assumption 3, while each element $\mathbb{E}[T_i(\Phi'_{\Delta^C,\phi})]$ of the summation in the third term of Equation (A.135) can be bounded as has been done for $\mathbb{E}[T_i(\Phi_{\Delta^C,N})]$ in the proof of Theorem 9, by considering a specific time horizon of length $N_\phi - \tau$ for phase Φ_ϕ. This is due to the fact that when we are considering rounds belonging to a single phase, we are effectively considering a SC-MAB setting. Formally, we have:

$$\begin{aligned}
\mathbb{E}[T_i(\Phi'_{\Delta^C,\phi})] &\leq \frac{N_\phi - \tau}{\tau}\left[\frac{56\log\tau}{(\Delta - 2\sigma\tau)^2} + \log\tau + 4 + \frac{22}{\log\tau} + \frac{1}{\tau^{\frac{1}{2}}}\right]\\
&\leq \frac{N_\phi}{\tau}\left[\frac{56\log\tau}{(\Delta - 2\sigma\tau)^2} + \log\tau + 4 + \frac{22}{\log\tau} + \frac{1}{\tau^{\frac{1}{2}}}\right]
\end{aligned}$$

Finally we have:

$$\begin{aligned}
\bar{R}_N(\mathfrak{U}) &\leq \sum_{i=1}^{K} \mathbb{E}[T_i(\Phi_{\Delta,N})] + \tau\Upsilon_N + \sum_{\phi=1}^{\Upsilon_N}\sum_{i=1}^{K} \mathbb{E}[T_i(\Phi'_{\Delta^C,\phi})] \qquad \text{(A.136)}\\
&\leq \mathcal{F}\Delta N^{\beta} + \tau\Upsilon N^{\alpha} + \\
&\quad + \sum_{\phi=1}^{\Upsilon_N}\sum_{i=1}^{K} \frac{N_\phi}{\tau}\left[\frac{56\log\tau}{(\Delta-2\sigma\tau)^2} + \log\tau + 4 + \frac{22}{\log\tau} + \frac{1}{\tau^{\frac{1}{2}}}\right] \qquad \text{(A.137)}\\
&\leq \tau 3\sigma\mathcal{F}N^{\beta} + \tau\Upsilon N^{\alpha} + \\
&\quad + \frac{NK}{\tau}\left[\frac{56\log\tau}{(\Delta-2\sigma\tau)^2} + \log\tau + 4 + \frac{22}{\log\tau} + \frac{1}{\tau^{\frac{1}{2}}}\right] \qquad \text{(A.138)}\\
&= \left(3\sigma\mathcal{F}N^{\beta} + \Upsilon N^{\alpha}\right)\tau + \\
&\quad + \frac{NK}{\tau}\left[\frac{56\log\tau}{(\Delta-2\sigma\tau)^2} + \log\tau + 4 + \frac{22}{\log\tau} + \frac{1}{\tau^{\frac{1}{2}}}\right], \qquad \text{(A.139)}
\end{aligned}$$

where we consider Assumption 1 and Assumption 3 in Equation (A.137) and we consider that $\Delta \leq 3\sigma\tau$ and that $\sum_{\phi=1}^{\Upsilon_N} N_\phi = N$ in Equation (A.138), which concludes the proof.

□

A.2.6 Proof of Assumption 3

Here we prove that the Assumption 3 on Δ in the SC-MAB experimental setting is satisfied. We want to show that the cardinality of the rounds in which at least one pair $i, j \in \{1, \dots, K\}, \quad i \neq j$ the following holds:

$$|\mu_{i,t} - \mu_{j,t}| < \Delta$$

is bounded by $\mathcal{F}\Delta$, where $\mathcal{F}$ is a constant w.r.t. the time horizon N. Let us consider $\Delta_0 = \frac{1}{3}$ which implies $\Delta \leq \frac{1}{3}$.

The evolution of the expected values of the arms over time in the SC-MAB we analysed in the experimental section is the following:

$$\mu_{i,t} = \frac{K-1}{K} - \frac{\left|1 + \frac{1}{2}(K-1)(1+\sin(t\sigma)) - i\right|}{K}$$

If we are in $\Phi_{\Delta,N}$ there exists a couple of index i and j s.t. $i \neq j$, we have:

$$\begin{aligned}
&|\mu_{i,t} - \mu_{j,t}| \\
&= \left| \frac{K-1}{K} - \frac{\left|1 + \frac{1}{2}(K-1)(1+\sin(t\sigma)) - i\right|}{K} - \frac{K-1}{K} + \frac{\left|1 + \frac{1}{2}(K-1)(1+\sin(t\sigma)) - j\right|}{K} \right| = \\
&= \left| - \frac{\left|1 + \frac{1}{2}(K-1)(1+\sin(t\sigma)) - i\right|}{K} + \frac{\left|1 + \frac{1}{2}(K-1)(1+\sin(t\sigma)) - j\right|}{K} \right| = \\
&= \frac{1}{K} \left| \left|1 + \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) - j\right| - \left|1 + \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) - i\right| \right| \leq \Delta,
\end{aligned}$$

thus we have:

$$-K\Delta < \left|1 + \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) - j\right| - \left|1 + \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) - i\right| < K\Delta.$$

In what follows, we divide the analysis in two cases.

Case 1**: t s.t.** $(1+\frac{1}{2}(K-1)(1+\sin(t\sigma))-j)(1+\frac{1}{2}(K-1)(1+\sin(t\sigma))-i) > 0$ Let us consider the case in which both $1 + \frac{1}{2}(K-1)(1+\sin(t\sigma) > 0$ and $1 + \frac{1}{2}(K-1)(1+\sin(t\sigma)) - i > 0$. The same holds in the case both the terms are negative and by inverting the roles of i and j. In the former case, the inequality becomes:

$$\begin{aligned}
&-K\Delta < 1 + \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) - j - 1 - \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) + i < K\Delta \\
&-K\Delta < i - j < K\Delta.
\end{aligned}$$

If $i > j$ the inequality $-K\Delta < i - j$ is always satisfied, while we need to examine whether $i - j < K\Delta$ or not. The most strict case is when the two arms being similar are $i = K$ and $j = 1$, thus:

$$\begin{aligned} &i - j < K\Delta \\ &K - 1 < K\Delta \\ &\Delta > \frac{K-1}{K}, \end{aligned}$$

which is always false if $K > 2$ since $\Delta_0 = \frac{1}{3}$. Thus the set of the rounds in this case is empty.

If $i < j$ the inequality $i - j < K\Delta$ is always satisfied, while we need to verify $-K\Delta < i - j$. The worst case is when $i = 1$ and $j = K$, thus:

$$\begin{aligned} &- K\Delta < i - j \\ &- K\Delta < 1 - K \\ &\Delta > \frac{K-1}{K} \end{aligned}$$

thus the same reasoning made for the previous case holds and we have an empty set of rounds.

Case 2**: t s.t.** $(1+\frac{1}{2}(K-1)(1+\sin(t\sigma))-j)(1+\frac{1}{2}(K-1)(1+\sin(t\sigma))-i) < 0$
Even in this case we analyse the case in which $1+\frac{1}{2}(K-1)(1+\sin(t\sigma))-j < 0$ and $1 + \frac{1}{2}(K-1)(1+\sin(t\sigma)) - i < 0$, being the opposite one analogous. In this case, the inequality becomes:

$$\begin{aligned} &- K\Delta < -1 - \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) + j + 1 + \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) - i < K\Delta \\ &- K\Delta < j - i < K\Delta \end{aligned}$$

whose analysis is the same as in Case 2 by substituting the role of i and j indexes.

Case 2**: t s.t.** $1+\frac{1}{2}(K-1)(1+\sin(t\sigma))-j < 0$ **and** $1+\frac{1}{2}(K-1)(1+\sin(t\sigma))-i < 0$

$$\begin{aligned} &- K\Delta < 1 + \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) - j + 1 + \tfrac{1}{2}(K-1)(1+\sin(t\sigma)) - i < K\Delta \\ &- K\Delta < 2 + (K-1)(1+\sin(t\sigma)) - j - i < K\Delta \\ &- K\Delta - 2 + j + i < (K-1)(1+\sin(t\sigma)) < K\Delta - 2 + j + i \\ &\frac{-K\Delta - 2 + j + i}{K-1} - 1 < \sin(t\sigma) < \frac{K\Delta - 2 + j + i}{K-1} - 1 \\ &\frac{1}{\sigma}\arcsin\left(\frac{-K\Delta - 2 + j + i}{K-1} - 1\right) < t < \frac{1}{\sigma}\arcsin\left(\frac{K\Delta - 2 + j + i}{K-1} - 1\right) \end{aligned}$$

We are interested in the number of rounds for which the inequalities hold, i.e.,:

$$\begin{aligned}|t| &= \left|\{t : \frac{1}{\sigma}\arcsin\left(\frac{-K\Delta-2+j+i}{K-1}-1\right) < t < \frac{1}{\sigma}\arcsin\left(\frac{K\Delta-2+j+i}{K-1}-1\right)\}\right| \\ &= \frac{1}{\sigma}\arcsin\underbrace{\left(\frac{K\Delta-2+j+i}{K-1}-1\right)}_{A} - \frac{1}{\sigma}\arcsin\underbrace{\left(\frac{-K\Delta-2+j+i}{K-1}-1\right)}_{B}.\end{aligned}$$

By relying on the following inequalities:

$$\begin{aligned}\arcsin(x) &< 2x \qquad x > 0,\\ \arcsin(x) &> 2x \qquad x < 0,\end{aligned}$$

we have that if $A < 0$ and $B > 0$, we can write:

$$|t| \leq \frac{1}{\sigma}\left(\frac{K\Delta-2+j+i}{K-1}-1\right) - \frac{1}{\sigma}\left(\frac{-K\Delta-2+j+i}{K-1}-1\right) = \frac{2K\Delta}{\sigma(K-1)},$$

thus Assumption 3 is satisfied with $\mathcal{F} := \frac{2K}{\sigma(K-1)}$

Finally we have to show that $A < 0$ and $B > 0$. Let us start with $A < 0$. The value minimizing A for the indexes are $i = 1$ and $j = 2$, consequently, we have:

$$\begin{aligned}\frac{K\Delta-2+j+i}{K-1}-1 &= \frac{K\Delta-2+2+1-K+1}{K-1}\\ &= \frac{K\Delta-K+2}{K-1} = \frac{(\Delta-1)K+2}{K-1} < 0\\ \Delta &< \frac{K-2}{K}\end{aligned}$$

which is satisfied since $\Delta_0 \leq \frac{1}{3}$.

Let us consider $B > 0$. Even in this case the choice of $i = 1$ and $j = 2$ is the one providing the most restrictive conditions. We have:

$$\begin{aligned}\frac{-K\Delta-2+j+i}{K-1}-1 &= \frac{-K\Delta-2+2+1-K+1}{K-1}\\ &= \frac{-K\Delta-K+2}{K-1} = \frac{-(\Delta+1)K+2}{K-1} > 0\end{aligned}$$

which is the same condition as in the $A > 0$ derivations. This concludes the proof.

A.2.7 Proof of Theorem 12

Theorem 12. *Given a UMAB setting $G = (A, E)$, the expected pseudo-regret of the UTS algorithm satisfies, for every $\varepsilon > 0$:*

$$\bar{R}_N(\mathit{UTS}) \leq (1+\varepsilon) \sum_{i \in \mathcal{N}(i^*)} \frac{\mu^* - \mu_i}{KL(\mu_i, \mu^*)} [\log(N) + \log\log(N)] + \tilde{C},$$

where $\tilde{C} > 0$ is a constant depending on ε, the number of arms K and the expected rewards $\{\mu_1, \dots, \mu_K\}$.

Proof. At first, the regret of the UTS algorithm $\bar{R}_N(\text{UTS})$ can be rewritten by dividing the N rounds in two sets: those rounds in which the best arm a^* is the leader , i.e., $l(t) = i^*$, and those in which the leader is another arm, i.e., $l(t) \neq i^*$:

$$\begin{aligned}
\bar{R}_N(\text{UTS}) &= \sum_{i \neq i^*} (\mu^* - \mu_i) \mathbb{E}[T_{i,N}] \\
&= \sum_{i \neq i^*} (\mu^* - \mu_i) \mathbb{E}\left[\sum_{t=1}^{N} \mathbf{1}\{i_t = i\}\right] \\
&= \underbrace{\sum_{i \neq i^*} (\mu^* - \mu_i) \mathbb{E}\left[\sum_{t=1}^{N} \mathbf{1}\{l(t) = i^* \wedge i_t = i\}\right]}_{\mathcal{R}_1} + \\
&+ \underbrace{\sum_{i \neq i^*} (\mu^* - \mu_i) \mathbb{E}\left[\sum_{t=1}^{N} \mathbf{1}\{l(t) \neq i^* \wedge i_t = i\}\right]}_{\mathcal{R}_2}
\end{aligned}$$

Let us focus on $\mathcal{R}_1$. When i^* is the leader, the proposed algorithm behaves like Thompson Sampling restricted to the optimal arm and its neighborhood $\mathcal{N}^+(i^*)$, and the regret upper bound is the one presented in Theorem 1 in [10] for TS algorithm, i.e., for every $\varepsilon > 0$:

$$\mathcal{R}_1 \leq (1+\varepsilon) \sum_{i \in \mathcal{N}(i^*)} \frac{\mu^* - \mu_i}{KL(\mu_i, \mu^*)} [\log(N) + \log\log(N)] + C_1, \quad \text{(A.140)}$$

where C_1 is an appropriate constant depending on ε and on the expected rewards μ_i of arms in $\mathcal{N}^+(i^*)$.

Now let us consider $\mathcal{R}_2$, we have:

$$\begin{aligned}\mathcal{R}_2 &= \sum_{i \neq i^*} \underbrace{(\mu^* - \mu_i)}_{\leq 1} \mathbb{E}\left[\sum_{t=1}^{N} \mathbf{1}\{l(t) \neq i^* \wedge i_t = i\}\right] \\ &\leq \sum_{i \neq i^*} \mathbb{E}\left[L_{i,N}\right].\end{aligned}$$

Here we want to upper bound the number of times a_i has been the leader $L_{i,N}$ with $\hat{L}_{i,N}$ defined as the number of rounds spent with a_i as leader in the case only its neighborhood is considered during the whole time horizon N. This is clearly an upper bound over $L_{i,N}$, since there is nonzero probability that the UTS algorithms moves in another neighborhood. From now on in the proof the analysis is carried on an algorithm working only on a unique neighborhood $\mathcal{N}(i)$.

$$\begin{aligned}\mathcal{R}_2 \leq \sum_{i \neq i^*} \mathbb{E}\left[L_{i,N}\right] \leq \sum_{i \neq i^*} \mathbb{E}\left[\hat{L}_{i,N}\right] &= \sum_{i \neq i^*} \sum_{t=1}^{N} \mathbb{E}\left[\mathbf{1}\{l(t) = i\}\right] \\ &= \sum_{i \neq i^*} \sum_{t=1}^{N} \mathbb{E}\left[\mathbf{1}\{\hat{\mu}_{i,t} = \max_{a_j \in \mathcal{N}(i)} \hat{\mu}_{j,t}\}\right],\end{aligned}$$

where, with abuse of notation, $l(t)$ is the leader at round t in this new problem where only $\mathcal{N}(i)$ is considered.

When $i \neq i^*$ is the leader, a_i is not the optimal arm. Thus, since we are in a unimodal setting, it exists an optimal arm $a_{i'} \in \mathcal{N}(i), i' \neq i$ s.t. $\mu_{i'} = \max_{i | a_i \in \mathcal{N}(i)} \mu_i$. Nonetheless, since a_i is the leader, its empirical mean is the maximum in its neighborhood and, in particular, $\hat{\mu}_{i,t} \geq \hat{\mu}_{i'}$. Thus, we have:

$$
\begin{aligned}
\mathcal{R}_2 &\leq \sum_{i \neq i^*} \sum_{t=1}^{N} \mathbb{E}\left[\mathbf{1}\{\hat{\mu}_{i,t} = \max_{a_j \in \mathcal{N}(i)} \hat{\mu}_{j,t}\}\right] \\
&\leq \sum_{i \neq i^*} \sum_{t=1}^{N} \mathbb{E}\left[\mathbf{1}\{\hat{\mu}_{i,t} \geq \hat{\mu}_{i',t}\}\right] \\
&= \sum_{i \neq i^*} \sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i,t} \geq \hat{\mu}_{i',t}\right) \\
&= \sum_{i \neq i^*} \sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i,t} - \mu_i - \frac{\Delta_i}{2} - \hat{\mu}_{i',t} + \mu_{i'} - \frac{\Delta_i}{2} \geq 0\right) \\
&\leq \sum_{i \neq i^*} \left[\underbrace{\sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i,t} - \mu_i - \frac{\Delta_i}{2} \geq 0\right)}_{\mathcal{R}_{i1}} + \underbrace{\sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i',t} - \mu_{i'} + \frac{\Delta_i}{2} \leq 0\right)}_{\mathcal{R}_{i2}}\right],
\end{aligned}
$$

where $\Delta_i = \max_{i' | a_i \in \mathcal{N}(i)} \mu_{i'} - \mu_i$ denotes the expected loss incurred in choosing arm a_i instead of its best adjacent one $a_{i'}$.

Let us focus on $\mathcal{R}_{i1}$:

$$
\begin{aligned}
\mathcal{R}_{i1} &= \sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right) \\
&= \sum_{t=1}^{N} \sum_{h=1}^{t} \mathbb{P}\left(T_{i,t} = h \wedge \hat{\mu}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right) \\
&= \sum_{t=1}^{N} \sum_{h=1}^{t} \mathbb{P}\left(T_{i,t} = h \mid \hat{\mu}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right) \mathbb{P}\left(\hat{\mu}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right) \\
&\leq \sum_{t=1}^{N} \sum_{h=1}^{t} \mathbb{P}\left(T_{i,t} = h \mid \hat{\mu}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right) e^{-\frac{h\Delta_i^2}{2}}
\end{aligned}
$$

Where the last inequality is due to the Hoeffding inequality [73]. By relying on the fact that $\sum_{h=x+1}^{\infty} e^{-kh} \leq \frac{1}{k} e^{-kx}$ and by considering $x = \frac{t}{|\mathcal{N}^+(i)|}$ we have:

$$\mathcal{R}_{i1} \leq \sum_{t=1}^{N} \left(\underbrace{\sum_{h=1}^{\frac{t}{|\mathcal{N}^{+}(i)|}} \mathbb{P}\left(T_{i,t} = h \mid \hat{\mu}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right)}_{=0} e^{\frac{h\Delta_i^2}{2}} + \frac{2}{\Delta_i^2} e^{-\frac{\frac{t}{|\mathcal{N}^{+}(i)|}\Delta_i^2}{2}} \right)$$
$$= \sum_{t=1}^{N} \frac{2}{\Delta_i^2} e^{\frac{-\frac{t}{|\mathcal{N}^{+}(i)|}\Delta_i^2}{2}} \leq C_2$$

where $\mathbb{P}\left(T_{i,t} = h \mid \hat{\mu}_{i,t} \geq \mu_i + \frac{\Delta_i}{2}\right) = 0$ for $h \leq \frac{t}{|\mathcal{N}^{+}(i)|}$ is due to the fact that the leader is chosen at least $\frac{t}{|\mathcal{N}^{+}(i)|}$ over t rounds and C_2 is a constant.

Let us focus on $\mathcal{R}_{i2}$ and the following proposition provided in [10]:

Proposition 1. *If we use a TS policy over a set of finite arms $\{a_i\}$ where $a_{i'}$ is the optimal one, there exist constants $b \in (0,1)$ and $C_b \leq \infty$ s.t.:*

$$\sum_{t=1}^{\infty} \mathbb{E}\left[\mathbf{1}\{T_{i',t} \leq t^b\}\right] \leq C_b. \tag{A.141}$$

Similarly to what has been derived for $\mathcal{R}_{i1}$ we have:

$$\begin{aligned}
\mathcal{R}_{i2} &= \sum_{t=1}^{N} \mathbb{P}\left(\hat{\mu}_{i',s} \leq \mu_{i'} - \frac{\Delta_i}{2}\right) \\
&= \sum_{t=1}^{N} \sum_{h=1}^{t} \mathbb{P}\left(T_{i',t} = h \wedge \hat{\mu}_{i',s} \leq \mu_{i'} - \frac{\Delta_i}{2}\right) \\
&= \sum_{t=1}^{N} \sum_{h=1}^{t^b} \mathbb{P}\left(T_{i',t} = h \wedge \hat{\mu}_{i',s} \leq \mu_{i'} - \frac{\Delta_i}{2}\right) + \\
&\quad + \sum_{t=1}^{N} \sum_{h=t^b+1}^{t} \underbrace{\mathbb{P}\left(T_{i',t} = h \mid \hat{\mu}_{i',s} \leq \mu_{i'} - \frac{\Delta_i}{2}\right)}_{\leq 1} \mathbb{P}\left(\hat{\mu}_{i',s} \leq \mu_{i'} - \frac{\Delta_i}{2}\right) \\
&\leq \sum_{t=1}^{\infty} \mathbb{E}\left[\mathbf{1}\{T_{i',t} \leq t^b\}\right] + \sum_{t=1}^{N} \sum_{h=t^b+1}^{t} \mathbb{P}\left(\hat{\mu}_{i',s} \leq \mu_{i'} - \frac{\Delta_i}{2}\right) \\
&\leq C_b + \sum_{t=1}^{N} \sum_{h=t^b+1}^{t} e^{-\frac{t\Delta_i^2}{2}} \\
&\leq C_b + \sum_{t=1}^{N} \frac{2}{\Delta_i^2} e^{-\frac{t^b\Delta_i^2}{2}} \leq C_3
\end{aligned}$$

since we are using TS in among arms in $\mathcal{N}(i)$ and the last inequality holds for all $b \in (0,1)$.

By considering the three partial results on $\mathcal{R}_1, \mathcal{R}_{i1}, \mathcal{R}_{i2}$ we have:

$$\begin{aligned}
&\bar{R}_N(\text{UTS}) \leq \mathcal{R}_1 + \sum_{i \neq i^*} (\mathcal{R}_{i1} + \mathcal{R}_{i2}) \\
&= (1+\varepsilon) \sum_{i \in \mathcal{N}(i^*)} (\mu^* - \mu_i) \frac{\log(N) + \log\log(N)}{KL(\mu_i, \mu^*)} + C_1 + (K-1)(C_2 + C_3)
\end{aligned}$$

considering $\tilde{C} = C_1 + (K-1)(C_2 + C_3)$ concludes the proof. □

APPENDIX $\mathcal{B}$

Non-Stationary Environment: Additional Material

B.1 Additional Sliding Window Algorithms

In this section, we report the algorithm used in the experimental analysis of the non-stationary case. While the algorithm SW-UCB has been proposed in [77] and is used here as baseline, the other presented algorithms are the straightforward application of the developed bounds in the sliding windows paradigm.

We recall that the expected value of the outcome μ_i over the last $\min\{\tau, t\}$ rounds is:

$$\bar{X}_{i,t,\tau} = \frac{1}{T_i(t-1,\tau)} \sum_{s=T_i(\max\{t-\tau,1\})}^{T_i(t-1)} X_{i,s},$$

where $T_i(t,\tau) = T_i(t) - T_i(\max\{t-\tau+1,1\})$ is the number of rounds the arm a_i has been selected in the last $\min\{\tau, t\}$ ones and its realization is:

$$\bar{x}_{i,t,\tau} = \frac{1}{T_i(t-1,\tau)} \sum_{s=T_i(\max\{t-\tau,1\})}^{T_i(t-1)} x_{i,s}.$$

Moreover, we recall that $\bar{X}_{ji,t,\tau}$ is the following convex linear combination of the sample means $\bar{X}_j, \dots, \bar{X}_i$:

$$\bar{X}_{ji,t,\tau} = \frac{1}{T_{ji}(t-1,\tau)} \sum_{k=j}^{i} \sum_{s=T_k(\max\{t-\tau,1\})}^{T_k(t-1)} X_{k,s},$$

where $T_{ji}(t,\tau) = \sum_{k=j}^{i} T_k(t-1) - T_k(\max\{t-\tau,1\})$ is the number of rounds one of the arms in $\{a_j, \dots, a_i\}$ has been selected in the last $\min\{\tau, t\}$ ones and the realization of $\bar{X}_{ji,t,\tau}$ is denoted as follows:

$$\bar{x}_{ji,t,\tau} = \frac{1}{T_{ji}(t-1,\tau)} \sum_{k=j}^{i} \sum_{s=T_k(\max\{t-\tau,1\})}^{T_k(t-1)} x_{k,s}.$$

At last, the variances $\bar{V}_{i,t,\tau}$ and $\bar{V}_{ji,t,\tau}$ of the two aforementioned random variables $\bar{X}_{i,t,\tau}$ and $\bar{X}_{ji,t,\tau}$ is:

$$\bar{V}_{i,t,\tau} = \frac{\sum\limits_{s=T_i(\max\{t-\tau,1\})}^{T_i(t-1)} \left(X_{i,s} - \bar{X}_{i,t,\tau}\right)^2}{T_i(t,\tau)}$$

$$\bar{V}_{ji,t,\tau} = \frac{\sum\limits_{s=T_i(\max\{t-\tau,1\})}^{T_i(t-1)} \left(X_{k,s} - \bar{X}_{ji,t,\tau}\right)^2}{T_i(t,\tau)},$$

respectively, and their realizations $\bar{v}_{i,t,\tau}$ and $\bar{v}_{ji,t,\tau}$:

$$\bar{v}_{i,t,\tau} = \frac{\sum\limits_{s=T_i(\max\{t-\tau,1\})}^{T_i(t-1)} \left(x_{i,s} - \bar{x}_{i,t,\tau}\right)^2}{T_{ji}(t-1,\tau)}$$

$$\bar{v}_{ji,t,\tau} = \frac{\sum\limits_{s=T_i(\max\{t-\tau,1\})}^{T_i(t-1)} \left(x_{k,s} - \bar{x}_{ji,t,\tau}\right)^2}{T_{ji}(t-1,\tau)},$$

respectively.

In what follows, the algorithms derived from the bound in [77] consider a parameter $\xi > 0$. For ease of comparison with [77], in the experimental section we set it to $\xi = 0.6$.

Algorithm 12: SW-UCB

```
Initialization
Input: $\xi$
for $t \in \{1, \dots, K\}$ do
    Play arm $a_t$ and observe $x_{t,1}$
Loop
for $t \in \{K+1, \dots, N\}$ do
    for $i \in \{1, \dots, K\}$ do
        Compute:
            $u_{i,t}^{(\text{SW-UCB})} = \bar{x}_{i,t,\tau} + \sqrt{\frac{\xi \log(\min\{t, \tau\})}{T_i(t-1, \tau)}}$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} a_i u_{i,t}^{(\text{SW-UCB})}$ and observe $x_{i_t, T_{i_t}(t)}$
```

Algorithm 13: SW-UCB1-M

```
Initialization
for $t \in \{1, \dots, K\}$ do
    Play arm $a_t$ and observe $x_{t,1}$
Loop
for $t \in \{K+1, \dots, N\}$ do
    for $i \in \{1, \dots, K\}$ do
        Compute:
            $u_{i,t}^{(\text{SW-UCB1-M})} = \min_{j \in \{1,\dots,i\}} \left\{ \bar{x}_{ji,t,\tau} + \sqrt{\frac{4 \log(\min\{t, \tau\}) + \log(i)}{2T_{ji}(t-1, \tau)}} \right\}$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} a_i u_{i,t}^{(\text{SW-UCB1-M})}$and observe $x_{i_t, T_{i_t}(t)}$
```

Algorithm 14: SW-UCB-L

```
Initialization
Input: $\mu_{\max}$
for $t \in \{1, \dots, K\}$ do
    Play arm $a_t$ and observe $x_{t,1}$
Loop
for $t \in \{K+1, \dots, N\}$ do
    for $i \in \{1, \dots, K\}$ do
        Compute:
            $u_{i,t}^{(\text{SW-UCB-L})} = \bar{x}_{i,t,\tau} + \sqrt{\frac{8\mu_{\max} \log(\min\{t, \tau\})}{T_i(t-1, \tau)}}$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} a_i u_{i,t}^{(\text{SW-UCB-L})}$ and observe $x_{i_t, T_{i_t}(t)}$
```

Algorithm 15: SW-UCB-LM

```
Initialization
Input: $\mu_{\max}$
for $t \in \{1, \dots, K\}$ do
    Play arm $a_t$ and observe $x_{t,1}$
Loop
for $t \in \{K+1, \dots, N\}$ do
    for $i \in \{1, \dots, K\}$ do
        Compute:
        $$u_{i,t}^{\text{(SW-UCB-LM)}} = \min_{j \in \{1,\dots,i\}} \left\{ \bar{x}_{ji,t,\tau} + \sqrt{\frac{2\mu_{\max}[\log(\min\{t,\tau\}) + \log(i)]}{T_{ji}(t-1,\tau)}} \right\}$$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} a_i u_{i,t}^{\text{(SW-UCB-LM)}}$ and observe $x_{i_t,T_{i_t}(t)}$
```

Algorithm 16: SW-UCBV

```
Initialization
Input: $\xi, c$
for $t \in \{1, \dots, K\}$ do
    Play arm $a_t$ and observe $x_{t,1}$
Loop
for $t \in \{K+1, \dots, N\}$ do
    for $i \in \{1, \dots, K\}$ do
        Compute:
        $$u_{i,t}^{\text{(SW-UCBV)}} = \bar{x}_{i,t,\tau} + \sqrt{\frac{2\bar{v}_{i,t,\tau}\xi \log(\min\{t,\tau\})}{T_i(t-1,\tau)}} + \\ + \frac{3c\xi \log(\min\{t,\tau\})}{T_i(t-1,\tau)}$$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\dots,K\}} a_i u_{i,t}^{\text{(SW-UCBV)}}$ and observe $x_{i_t,T_{i_t}(t)}$
```

Algorithm 17: SW-UCBV-M

```
**Initialization**
**Input:** $\xi, c$
**for** $t \in \{1, \ldots, K\}$ **do**
    Play arm $a_t$ and observe $x_{t,1}$
**Loop**
**for** $t \in \{K+1, \ldots, N\}$ **do**
    **for** $i \in \{1, \ldots, K\}$ **do**
        Compute:
        $$u_{i,t}^{\text{(SW-UCBV-M)}} = \min_{j \in \{1,\ldots,i\}} \left\{ \sqrt{\frac{2\bar{v}_{ji,t,\tau}[\xi \log(\min\{t,\tau\}) + \log(i)]}{T_{ji}(t-1,\tau)}} + \right. \\ \left. + \frac{3c[\xi \log(\min\{t,\tau\}) + \log(i)]}{T_{ji}(t-1,\tau)} + \bar{x}_{ji,t,\tau} \right\}$$
    Play arm $a_{i_t}$ such that $i_t = \arg\max_{i \in \{1,\ldots,K\}} a_i u_{i,t}^{\text{(SW-UCBV-M)}}$ and observe $x_{i_t, T_{i_t}(t)}$
```

B.2 Sensitivity Analysis

Here we present the results of the sensitivity analysis for parameter α of Theorem 8 in the AC-MAB and for parameter β of Theorem 9 in the SC-MAB.

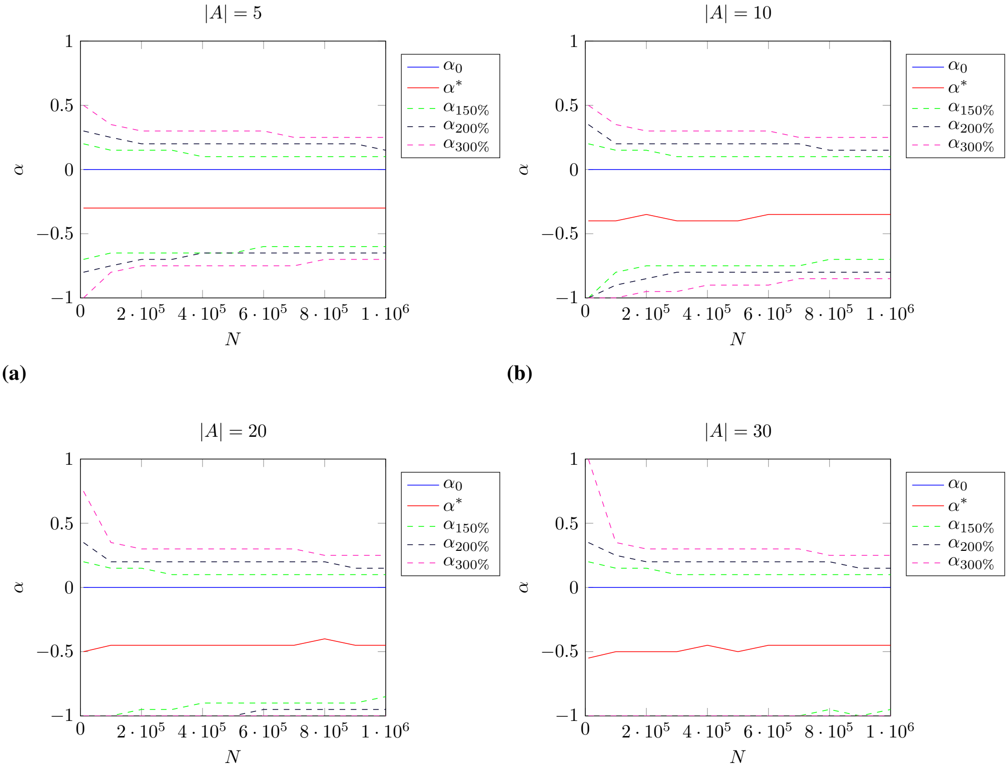
B.2.1 AC-MAB Setting

Experimental Setting We compare the performance of SW-TS using different sliding windows $\tau = N^{\frac{1-\alpha}{2}}$ with $\alpha = \{-1, -0.95, \dots, 0.95, 1\}$. We consider a time horizon $N \in \{10^4, 10^5, 2 \cdot 10^5, 3 \cdot 10^5, \dots, 9 \cdot 10^5, 10^6\}$ and a number of arms $K \in \{5, 10, 20, 30\}$. We split the time horizon N into four phases of equal length. During each phase, we select randomly the expected value $\mu_{i,\phi}$ for each arm i. After each breakpoint, we randomly change the expected value $\mu_{i,\phi}$ of each arm a_i, making sure that there is never the same optimal arm in two different phases, i.e., $a_{i^*_\phi} \neq a_{i^*_{\phi'}}, \forall \phi, \phi'$ with $\phi \neq \phi'$. We generate 10 configurations for each combination of N and K as described above and we provide the results averaged over the configurations and over 100 independent trials for each of them.

Results In Figure B.1, with α^* we reported, for each of the possible values of N, the α with which SW-TS achieved the best performance in terms of $\bar{R}_N(\mathfrak{U})$. In order to understand how the regret gets worse as α gets far from the optimal α^*, we plot as $\alpha_{150\%}$, $\alpha_{200\%}$ and $\alpha_{300\%}$ the α for which correspond, respectively, a 150%, 200% and 300% increase w.r.t. the regret achieved with α^*. In the experimental analysis of Section 6.4.1, we use a sliding window $\tau \propto \sqrt{N}$ which corresponds to $\alpha = 0$, reported in the figures as α_0. It can be observed that using $\alpha = 0$ corresponds to an increase lower than the 150% w.r.t the regret achieved with α^*. It can be noted that α^* always corresponds to a negative value, suggesting that a sliding window τ longer than the one obtained with α_0 is preferable.

In Figure B.2, we report the results in terms of $\hat{R}_N(\mathfrak{U}) = \bar{R}_N(\mathfrak{U})/N$ for all the values of α as the time horizon N varies with the number of arms K fixed. The lowest point of the lines corresponds to α^* for the considered time horizon. As it is possible to see from the figures, no matter the time horizon, the lowest regret is always achieved with almost the same value of α. It can also be observed that $\hat{R}_N(\mathfrak{U})$ grows faster moving from α^* toward lower values of α. Conversely, the growth moving toward higher values of α is initially smoother. Such behavior suggest that an underestimation of the optimal sliding window is safer than an overestimation.

In Figure B.3, we report the results in terms of $\hat{R}_N(\mathfrak{U}) = \bar{R}_N(\mathfrak{U})/N$ for all the values of α as the number of arms K varies with the time horizon N fixed. Again, the lowest point of the lines corresponds to α^* for the considered number of arms. It can be observed that the higher the number of arms K the lower the value of α^*. Intuitively, this behavior is due to the fact that when SW-TS has more arms to play, it also needs more samples for each arm to understand which is the optimal one, thus a longer sliding window τ is preferable.



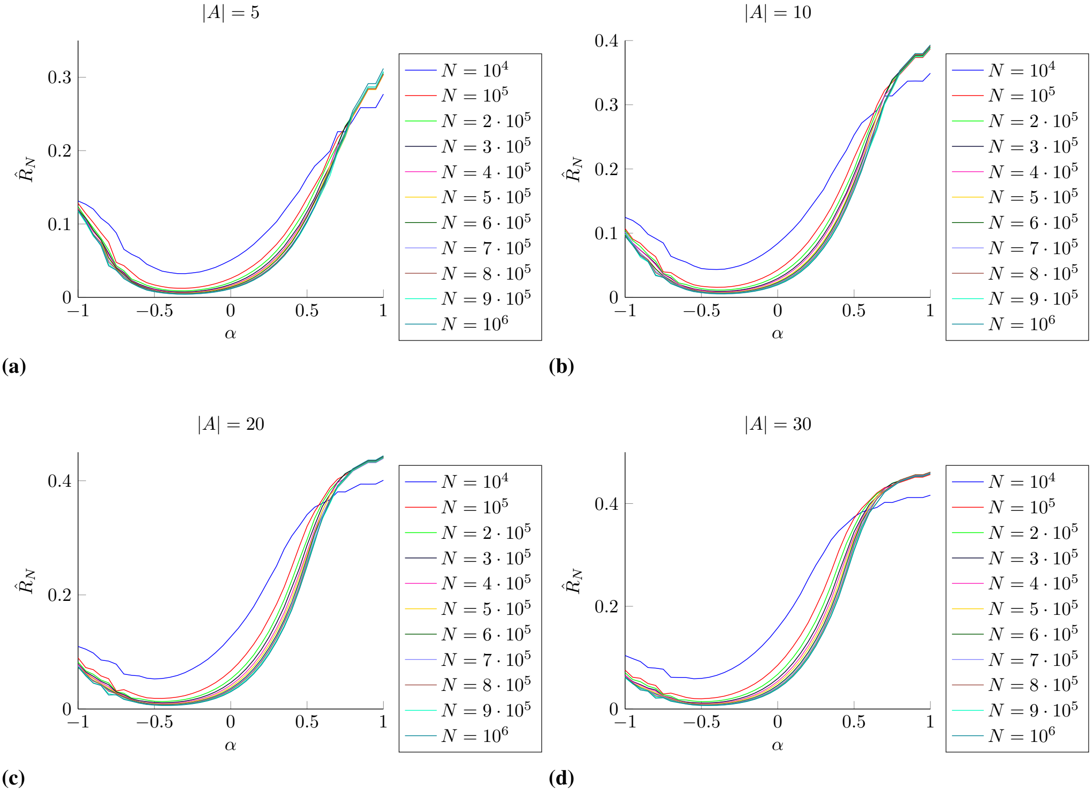
(a)

(b)

(c)

(d)

Figure B.1: *AC-MAB*



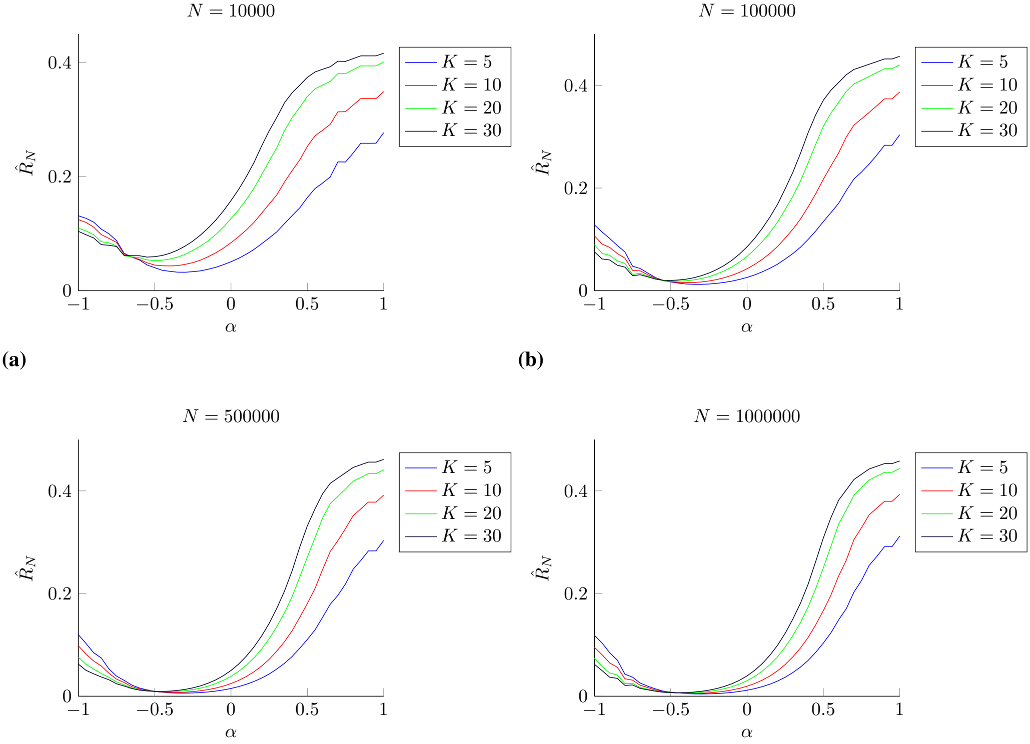
(a)

(b)

(c)

(d)

Figure B.2: *AC-MAB: Results in terms of $\hat{R}_N(\mathfrak{U}) = \bar{R}_N(\mathfrak{U})/N$ for all the values of α as the time horizon N varies with the number of arms K fixed.*



(a)

(b)

(c)

(d)

Figure B.3: *AC-MAB: Results in terms of* $\hat{R}_N(\mathfrak{U}) = \bar{R}_N(\mathfrak{U})/N$ *for all the values of* α *as the number of arms* K *varies with the time horizon* N *fixed.*

B.2.2 SC-MAB Setting

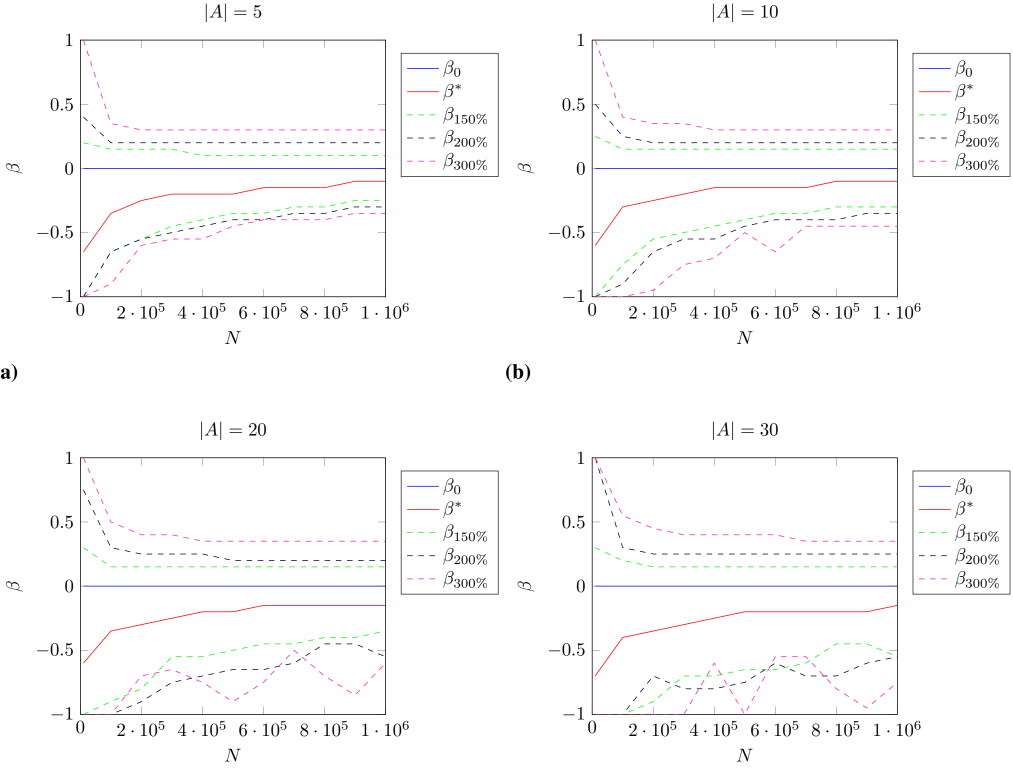
Experimental Setting We compare the performance of SW-TS using different sliding windows $\tau = N^{\frac{1-\beta}{2}}$ with $\beta = \{-1, -0.95, \dots, 0.95, 1\}$. We consider a time horizon $N \in \{10^4, 10^5, 10^6\}$ and a number of arms $K \in \{5, 10, 20, 30\}$. The expected value $\mu_{i,t}$ of arm a_i changes according to the following function:

$$\mu_{i,t} = \frac{K-1}{K} - \frac{|w(t) - i|}{K}, \quad w(t) = 1 + \frac{(K-1)(1+\sin(t\sigma))}{2}.$$

Results In Figure B.4, with β^* we report, for each of the possible values of N, the β with which SW-TS achieved the best performance in terms of $\bar{R}_N(\mathfrak{U})$. In order to understand how the regret gets worse as β gets far from the optimal β^*, we plot as $\beta_{150\%}$, $\beta_{200\%}$ and $\beta_{300\%}$ the β for which correspond, respectively, a 150%, 200% and 300% increase w.r.t. the regret achieved with β^*. In the experimental analysis of Section 6.4.2, we use a sliding window $\tau \propto \sqrt{N}$ which corresponds to $\beta = 0$, reported in the figures as β_0. It can be observed that using $\beta = 0$ corresponds to an increase of the regret lower than the 150% w.r.t the regret achieved with β^*. It can be noted that β^* always corresponds to a negative value, suggesting that a sliding window τ longer than the one obtained with τ_0 is preferable.

In Figure B.5, we report the results in terms of $\hat{R}_N(\mathfrak{U}) = \bar{R}_N(\mathfrak{U})/N$ for all the values of β as the time horizon N varies with the number of arms K fixed. The lowest point of the lines corresponds to β^* for the considered time horizon. As it is possible to see from the figures, no matter the time horizon, the lowest regret is always achieved with almost the same value of β. It can also be observed that, $\hat{R}_N(\mathfrak{U})$ grows faster moving from β^* toward higher values of β. Conversely, the growth moving toward higher values of β is smoother. Such behavior suggest that using a sliding window τ slightly longer than the one of the optimal β^* is preferable w.r.t. a slightly smaller one.

In Figure B.6, we report the results in terms of $\hat{R}_N(\mathfrak{U}) = \bar{R}_N(\mathfrak{U})/N$ for all the values of β as the number of arms K varies with the time horizon N fixed. The lowest point of the lines corresponds to β^* for the considered number of arms. It can be observed that the higher the number of arms K the lower the value of β^*. Intuitively, this behavior is due to the fact that if SW-TS has more arms to play, it also needs more samples for each arm to understand which is the optimal one, thus a longer sliding window τ is preferable.



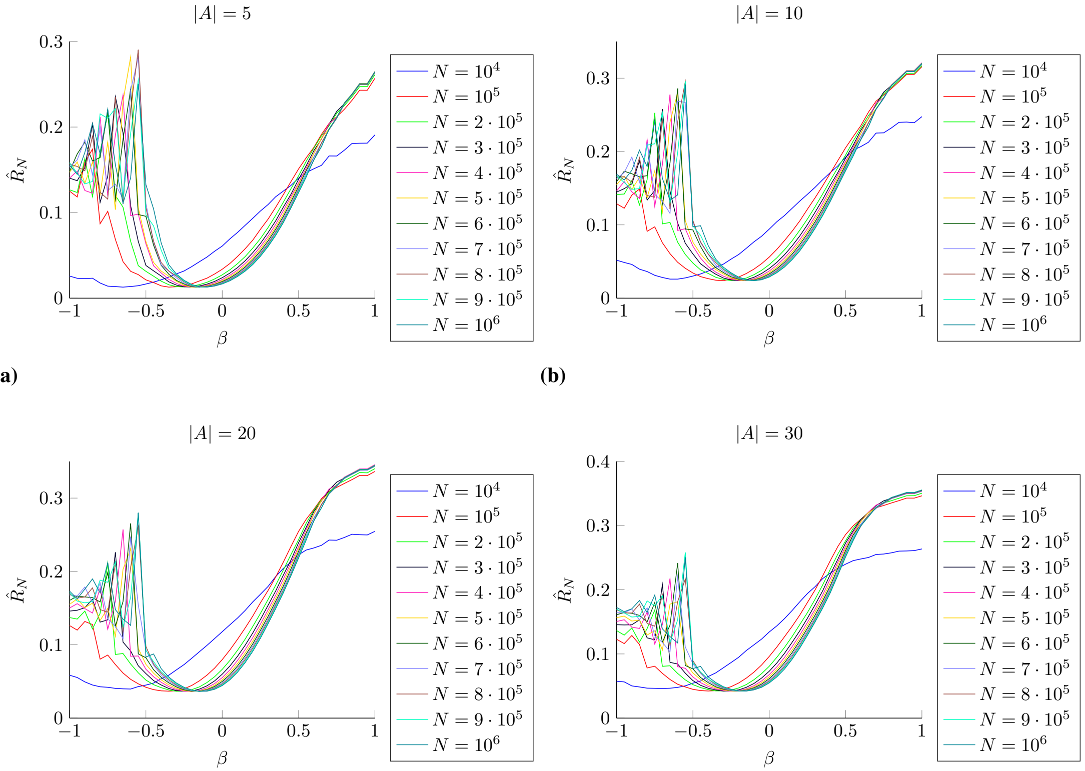
(a)

(b)

(c)

(d)

Figure B.4: *SC-MAB*



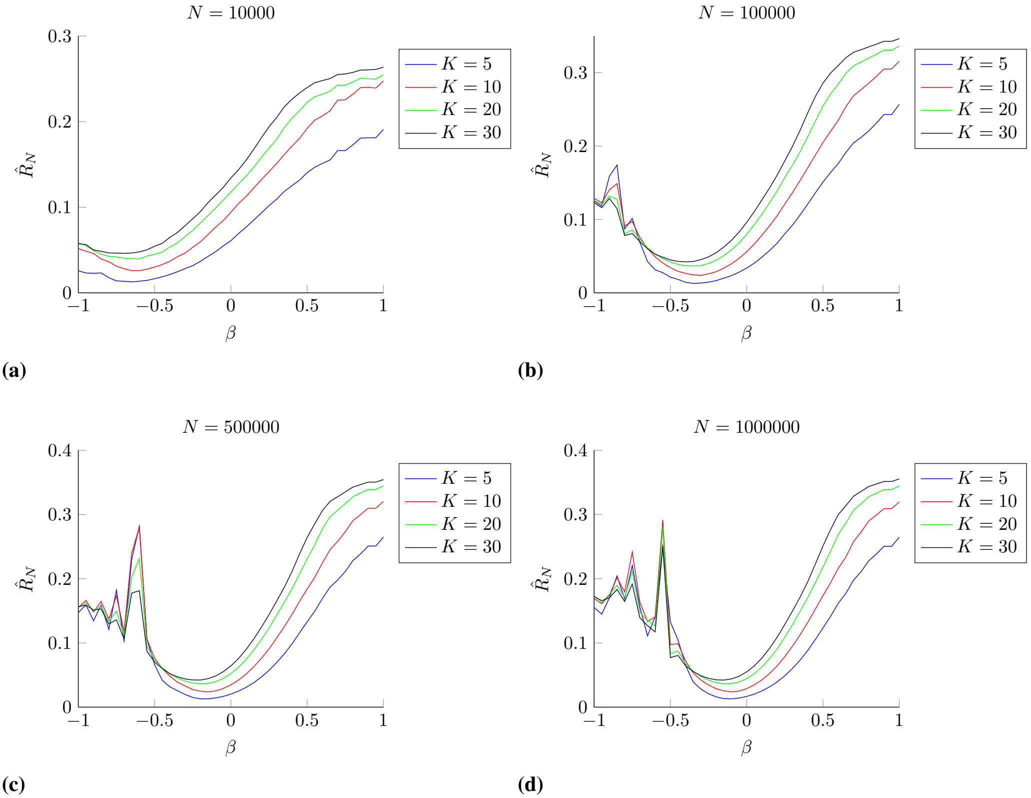
(a)

(b)

(c)

(d)

Figure B.5: *SC-MAB: Results in terms of $\hat{R}_N(\mathfrak{U}) = \bar{R}_N(\mathfrak{U})/N$ for all the values of β as the time horizon N varies with the number of arms K fixed.*



(a)

(b)

(c)

(d)

Figure B.6: *SC-MAB: Results in terms of $\hat{R}_N(\mathfrak{U}) = \bar{R}_N(\mathfrak{U})/N$ for all the values of β as the number of arms K varies with the time horizon N fixed.*