POLITECNICO DI MILANO
School of Industrial and Information Engineering
Master of Science in Mathematical Engineering

Quasi-safe Bandit Algorithms for the Bid Optimization Problem in Online Advertising

Supervisor: Prof. Nicola Gatti

Co-supervisor: Dott. Matteo Castiglioni

Master thesis of:
Samuele Milanesi, 920341

Academic Year 2019-2020

A chi mi ha insegnato qualcosa.

Sommario

Nel corso dell'ultimo decennio il volume annuale degli investimenti in campagne pubblicitarie online è quintuplicato. Nel solo mercato USA durante il 2020 si stima che tale volume superi i 135 miliardi di dollari, attestandosi come principale mezzo pubblicitario. Il fondamentale vantaggio di questo mezzo è la capacità di controllo che gli inserzionisti hanno in termini sia di personalizzazione delle campagne sia di valutazione sull'impatto che esse sortiscono. La mole di dati sulle performance e l'ampio ventaglio di tipologie di annunci rendono l'ottimizzazione di queste campagne non percorribile manualmente. Lo sviluppo di algoritmi per problemi di ottimizzazione in processi decisionali iterati è ben noto nel campo del Reinforcement Learning. In questo lavoro ci focalizziamo sul framework Stochastic Bandit con l'obiettivo di proporre soluzioni che tengano in conto vincoli di business reali, quali vincoli di budget e vincoli di ritorno sull'investimento (ROI). Nell'elaborato presentiamo due algoritmi rispettivamente per il framework Stochastic Multi-Armed Bandit con vincoli di budget e Stochastic Multi-Armed Bandit con vincoli di ROI. Analizziamo le prestazioni teoriche degli algoritmi mostrando che ammettono bound sublineari per il regret. Condizione necessaria per ottenere i suddetti bound è ammettere una violazione dei vincoli entro una certa soglia di tollerabilità. Dunque, indaghiamo il rapporto tra numero atteso di violazioni intollerabili dei vincoli e tale soglia di intollerabilità, aspetto ancora inesplorato a livello teorico. Estendiamo infine gli algoritmi, e l'analisi degli stessi, al contesto combinatorio con feedback Semi-Bandit. Questo permette l'uso di tali algoritmi nello scenario, più realistico, in cui le campagne degli inserzionisti si compongano di più sotto-campagne, diversificate ad esempio per canale, target o contesto.

Abstract

Over the past decade, the annual volume of investments in online advertising campaigns has quintupled. In the US market alone, during 2020, it is estimated that this volume exceeds 135 billion dollars, making it the primary advertising medium. The fundamental advantage of this medium is the control capacity that advertisers have in terms of both personalization of campaigns and evaluation of their impact. The amount of performance data and the wide range of ad types make the optimization of these campaigns not addressable by hand. The development of algorithms for optimization in sequential decision-making problems is well known in the field of Reinforcement Learning. In this work, we focus on the Stochastic Bandit framework to propose solutions that take into account real business constraints, such as budget constraints and return on investment (ROI) constraints. In the dissertation, we present two algorithms respectively for the Stochastic Multi-Armed Bandit framework with budget constraints and Stochastic Multi-Armed Bandit with ROI constraints. We analyze the theoretical performance of the algorithms by showing sublinear bounds for the regret. A necessary condition to find sublinear bounds is to allow a violation of the constraints up to a selected threshold. Thus, we also investigate the relationship between the expected number of intolerable violations of the constraints and the threshold of intolerability, which is an aspect that results unexplored at a theoretical level. Finally, we will extend the algorithms, and their analysis, to the combinatorial context with Semi-Bandit feedback. This allows the use of these algorithms in the more realistic scenario in which advertisers' campaigns are composed by several sub-campaigns diversified, for example, by target, channel or context.

Contents

List of Algorithms

List of Frameworks

Chapter 1

Introduction

In this chapter, we first present an overview of the context and motivations of our work. Then we show the main points of our contribution. Finally, we summarize the structure of the dissertation.

1.1 Overview

Thanks to rising internet penetration rates and the ever-expanding popularity of digital platforms, digital advertising has become one of the most influential advertising mediums. Nearly 137 billion U.S. dollars were spent on digital advertising in 2020 only in the United States[1]. This figure is forecast to increase in the upcoming years[2]. Choosing digital platforms for ads brings two advantages: it allows accurate control of the advertising campaigns' parameters and permits a rigorous evaluation of the publicity strategy's performances.

By contrast, this opportunity comes with the need to choose an effective way to optimize those campaigns. Doing this is not a trivial task. The most influential advertising platforms provide a massive amount of data and a vast spectrum of parameters to be tuned. This fact leads to the necessity of learning algorithms that can calibrate those parameters incrementally, basing on the collected data.

We focus on pay-per-click advertising, in which the advertiser pays only if her ads are clicked. Advertising platforms generally make auctions for

[1] Statista, Digital advertising spending in selected countries worldwide in 2020 (in million U.S. dollars) Statista, https://www.statista.com/statistics/459632/digital-advertising-revenue-countries-digital-market-outlook/

[2] Zenith, Internet advertising spending in North America from 2000 to 2022 (in million U.S. dollars) Statista, https://www.statista.com/statistics/882027/internet-advertising-expenditure-in-north-america/

the prices-per-click of different sets of options. The advertiser makes a bid, which is the maximum amount of money she is willing to pay for a click. The goal is to adjust the bids sequentially, trying to maximize the expected revenue cumulated over time. This can be seen as a sequential decision-making problem: the advertiser sets a bid, receives the feedback about how effective her choice has been, and based on this feedback, she resets her bid, trying to improve her results. The problem of balancing the need to gather new information about which are the best bids and using that information is known in Reinforcement Learning [20] as the *exploration-exploitation dilemma.*

A framework in which this dilemma has been addressed reaching strong theoretical guarantees is the Multi-Armed Bandit framework, introduced in [21]. The name bandit refers to the informal term for a slot machine (*one-armed bandit*). When a gambler faces many slot machines at once (a *multi-armed bandit*), she repeatedly chooses where to insert the next coin. This leads to a sequential decision problem: the objective of the gambler is to find the slot machine with the best expected return, without losing too much money in the learning process. Bandit problems are sequential decision-making with limited information and naturally address the fundamental trade-off between exploration and exploitation in sequential experiments. The player must weigh the exploitation of actions that performed well in the past and the exploration of actions that could be a source of even better performances. The advertising optimization problem naturally fits this framework.

Nevertheless, there is a further business aspect to address in the advertising scenario: the advertiser has to deal with the cost of her choices. A crucial point in building a publicity campaign is balancing the need to reach high volumes of revenues and the necessity to maximize the Return-on-Investment (ROI). In [10] the authors combine theory and empirics based on Google's Ads Exchange data to show that a significant set of buyers in online advertising markets are financially constrained. Moreover, they show that this behavior can be explained if we assume that they have a minimum ROI requirement.

From a modeling perspective, this leads to a constrained formulation of the bandit problem: the decision-maker wishes to maximize the cumulative revenue while respecting constraints on cost and ROI at each procedure's iteration. In literature are studied algorithms for bid optimization that take into account budget constraint. In [7] authors propose a Multi-Armed Bandit framework in the case of a finite number of possible bids. In [22] results are extended to a continuous space of bids. These works do not consider a daily budget limitation, but they respect a cumulative cost constraint over time. More recent works study an extension of the framework in which multiple advertising campaigns are managed simultaneously. This extension is much closer to real advertising campaigns in which the advertiser can compose many sub-campaigns. The sub-campaigns differ in platforms (e.g. Google, Bing, Facebook), targets (e.g. keywords, interests, language, geographic area), and formats (e.g. text, images, video). In [17, 18] this problem is addressed with an algorithm that combines Combinatorial Multi-Armed Bandits and Gaussian Processes to perform *bid/budget* allocation each round. There is still a vast literature that focuses on bid optimization in online advertising (e.g. [11, 9, 18]), however none of the above works deals directly with the idea of ROI constraint. In the unpublished work [5], a constrained Combinatorial Bandit problem with stochastic revenues and costs is considered. In this case, both cost and ROI constraints are part of the model. Authors show a critical result: no algorithm can guarantee a sublinear total regret while ensuring that the expected number of constraint violations is sublinear in the number of rounds. This result is the starting point of the following dissertation.

1.2 Contributions

This thesis expands the Bandit framework to deal with daily budget and ROI constraints. First, we focus on the Stochastic Multi-Armed Bandit framework with cost feedback (CostMAB). We start from the previously mentioned impossibility result from [5] and elaborate algorithms that can achieve sublinear regret while maintaining a sublinear expected number of constraint violations assuming a threshold of tolerance in these constraints. We call an algorithm that satisfies this property about the expected number of violations *quasi-safe algorithm.* We propose two algorithms for the CostMAB setting. First, we propose an algorithm to deal with budget constraints; we show theoretical results on performances and we show that is a quasi-safe algorithm. Second, we propose a more sophisticated algorithm to deal with the case of ROI constraint. Even in this case, we show theoretical results in terms of both performances and quasi-safety. We show how to derive from these algorithms a third policy that deals with both budget and ROI constraints. The proposed algorithms are based on the well-known principle of optimism in the face of uncertainty proposed in the seminal paper [14] and largely exploited in UCB-like algorithms in bandit contexts (see [2]). Finally, we expand the framework to deal with multiple sub-campaigns proposing algorithms for Combinatorial Bandit problems with Semi-Bandit feedback. We focus on the framework of Multi-Task Stochastic Semi-Bandit with cost feedback (CostMTSSB). This framework can model the scenario in which the advertiser has to deal with M sub-campaigns to be optimized simultaneously. Again, our focus will be on proving theoretical guarantees about the quasi-safety of the proposed algorithms.

1.3 Structure of the thesis

In Chapter 2 we introduce an overview of the Multi-Armed Bandit framework, and its combinatorial extension. We present the main algorithms proposed in the literature with their theoretical guarantees. Particular importance will be given to the principle of optimism in the face of uncertainty (OFU) that will drive the construction of our algorithms.

In Chapter 3 we formalize the problem introducing two settings. The former is the Multi-Armed Bandit setting with cost feedback (CostMAB), the latter is the Multi-Task Stochastic Semi-Bandit setting with cost feedback (CostMTSSB). We introduce the concept of safe-learning and quasi-safe learning with respect to both budget and ROI constraints. Finally we show how to frame the advertising bid optimization problem into the aforementioned settings.

In Chapter 4 we focus on the CostMAB setting. We propose and analyze two algorithms. The first deals with budget constraints, the second with ROI constraints. We evaluate theoretical bound for their performances and analyze their safety guarantees.

In Chapter 5 we focus on the CostMTSSB setting. Also in this case, we propose two algorithms to deal with budget and ROI constraints respectively. We also show how these two algorithms can be assembled to obtain a policy that takes in account both budget and ROI constraints.

Finally, Chapter 6 summarizes the main results of this dissertation and details possible future developments.

Chapter 2

Literature review

In this chapter, we present an overview of the main tools we will use throughout the thesis.

In the first section, we study the Stochastic Multi-Armed Bandit (MAB) framework. We introduce the fundamental assumptions, the learning objectives, and the concept of regret. Then we present the main algorithms that have been proposed in the literature and their theoretical results. We conclude the first section with theoretical lower bounds for the regret and examples of applications.

In the second section, we extend the previous context to Combinatorial Stochastic Bandits and Semi-Bandit feedback, focusing on the subclass of Multi-Task Stochastic Semi-Bandit (MTSSB). Again, we present the fundamental assumptions. We elaborate on the difference between Bandit feedback and Semi-Bandit feedback. We present the main algorithms with results on their theoretical performance. We conclude the chapter with examples of applications. For a more in-depth analysis of the themes introduced, we refer to [19, 3, 15].

2.1 Stochastic Bandits

The bandit problem has a history now almost 100 years old. It was introduced in 1933 by Thompson in [21], where the problem was presented in the context of medical trials. Thompson addressed the problem of minimizing the health impact on trial participants by adapting treatments iteratively based on the drugs' effects.

In the 1950s, Robert Bush and Frederick Mosteller developed a stochastic model study with applications in learning [4]. They devised an experiment in which participants interfaced with a two-armed slot machine called a *two-*

armed bandit (the name comes from the American slang used to refer to slot machines). The simplest example of a Multi-Armed Bandit problem is thus presented.

Example 1. Imagine being faced with the above-mentioned two-armed bandit device. Both levers have been already sampled five times each, obtaining the following results.

Round	1	2	3	4	5	6	7	8	9	10
Right lever reward	0	-	1	-	0	-	1	-	-	0
Left lever reward	-	0	-	1	-	1	-	1	0	-

How should the strategy be adjusted, given these results? The left lever seems to be the one with the higher expected payoff, based on the data collected so far, but it may have been just a lucky event. How many more times should one test both levers to be confident enough about which one is the best?

This simple example captures the essence of the Multi-Armed Bandit problem: finding a way to balance exploration of the environment and exploiting the information gathered.

2.1.1 Problem description

A Stochastic Bandit problem is a sequential game between a player (usually referred to as *learner* or *decision-maker*) and the environment. The game lasts for $T \in \mathbb{N}$ round. At each round t the player chooses an action A_t to be played from a set of available actions $\mathcal{A}$. These actions are also called *arms* for the aforementioned historical reasons. We call Multi-Armed Bandit any bandit problem in which the cardinality of $\mathcal{A}$ is a natural number $\#\mathcal{A} = K \in \mathbb{N}$. This framework is also known as K-Armed Bandit. More general cases have been studied in the literature; in these works, $\mathcal{A}$ can have infinite or even continuous cardinality. Once the player performs the action A_t, the environment provides a reward X_t. In the case of Stochastic Bandit, this reward is stochastic. The so-called *adversary* case is also extensively studied in the literature; in this case, the environment reacts in an arbitrary (and generally hostile) way to the decision-maker's actions. Although of great interest, this case is not explored in this dissertation; in fact, in advertising is a common assumption that users would react stochastically to displayed ads, clicking or not on them, rather than adversarially.

The action A_t that the player selects in round t can only depend on the history of the game up to that moment, that is, on the sequence $H_{t-1} := \{A_1, X_1, A_2, X_2, \ldots, A_{t-1}, X_{t-1}\}$ which consist of actions performed and rewards obtained in the past. We define policy $\mathfrak{U}$ as map from the space of possible histories $\mathcal{H}$ to the set of possible actions $\mathcal{A}$. In general, this map may not be deterministic.

The learner's goal is generally to maximize the cumulative reward over the T round, $\sum_{t=1}^{T} X_t$. We will formalize this goal in the next paragraphs, introducing the concept of regret.

The fundamental problem in this context is that the environment's reactions are not only stochastic, but their probability distribution is unknown to the learner. We will shortly reduce this indecision space by introducing the concept of environment-class, namely classes of probability distributions admissible for the environment's feedback.

2.1.2 Assumptions, environment-classes, concentration inequalities

To detail the Stochastic MAB problem is necessary to detail the assumptions on which it is based. **Assumptions.**

We define an instance of a Stochastic MAB problem as a set of distributions $\Psi := \{P_a : a \in \mathcal{A}\}$. Each distribution P_a of Ψ is defined on the space of possible rewards obtainable by the decision-maker by selecting the arm $a \in \mathcal{A}$. It follows that if in round t the player selects the action A_t, the environment will sample the reward X_t from the distribution P_t. Observe that the interaction between agent and environment induces a probability measure on the sequence of actions-feedback $\{A_1, X_1, A_2, X_2, \ldots, A_T, X_T\}$. We assume that the sequence of action-feedback satisfies the following hypotheses:

- the conditional distribution of the reward X_t given the sequence of feedback actions up to round t, $\{A_1, X_1, \ldots, A_{t-1}, X_{t-1}, A_t\}$, is P_{A_t}.
- the distribution of A_t given the sequence $\{A_1, X_1, \ldots, A_{t-1}, X_{t-1}\}$ is $\pi_t(\cdot|A_1, X_1, \ldots, A_{t-1}, X_{t-1})$. The sequence $\mathfrak{U} := \{\pi_t\}_{t=1}^{T}$ is called policy and characterizes the decision-maker.

The first hypothesis summarizes the idea that the environment samples the value of X_t from P_{A_t}. The second assumption requires that the player's actions be selected on and only based on the history before round t.

This description of the problem may seem mathematically incomplete. In fact, the question arises about what is (if it exists!) the probability space in which the measures introduced above are defined. We refer to [15] for a deeper analysis of the mathematical assumptions.

We summarize below the sequential game that describes the framework.

Framework protocol 1 Stochastic Multi-Armed Bandit (MAB)

```
Input: $T$ time horizon, $K$ number arms, $\mathcal{A} = [K]$ set of arms
for: $t = 1, ..., T$
    1. Select an arm $A_t \in \mathcal{A}$
    2. Collect the reward $X_t$ sampled from $P_{A_t}$.
```

Environment-classes. The problem with building algorithms for this framework is that the instance $\Psi := \{P_a : a \in \mathcal{A}\}$ is not known to the player. In principle, a policy that performs very well on one instance Ψ could arbitrarily perform badly on a second instance Ψ'. Nevertheless, it is common to assume that the decision-maker has partial knowledge of the instance. More precisely, we define with Ξ the domain of the possible instances that can be presented to the learner. We call the set Ξ environment-class. This set is known to the decision-maker and allows to find algorithms that ensure good performance for any instance belonging to Ξ.

An environment-class can be defined as

$$\Xi := \{\Psi = \{P_a : \ a \in \mathcal{A}\} : \ P_a \in \mathcal{M}_a \forall a \in \mathcal{A}\}$$

where $\mathcal{M}_a$ is a set of possible distributions associated with the reward of arm $a \in \mathcal{A}$.

Example 2. Some examples of environment-class in the Stochastic MAB framework are:

- Bernoulli environment: $\Xi_{\mathcal{B}} := \{\{\mathcal{B}(p_a)\}_{a\in\mathcal{A}} : \ p_a \in [0,1] \ \forall a \in \mathcal{A}\}\}$, where $\mathcal{B}(p)$ is the Bernoulli distribution of mean p.
- Uniform environment: $\Xi_{\mathcal{U}} := \{\{\mathcal{U}(l_a, u_a)\}_{a\in\mathcal{A}} : \ l_a, \ u_a \in \mathbb{R} \ \forall a \in \mathcal{A}\}$, where $\mathcal{U}(l,u)$ is the uniform distribution on the real interval $[l,u]$.
- σ-sub-Gaussian environment: $\Xi_{SG(\sigma)} := \{\{P_a\}_{a\in\mathcal{A}} : \ P_a$ is a σ-sub-Gaussian distribution $\forall a \in \mathcal{A}\}$. Recall that a real random variable X has σ-sub-Gaussian distribution if $\forall \ \lambda \in \mathbb{R}, \ \ \mathbb{E}\left[e^{\lambda X}\right] \leq \exp(\lambda^2\sigma^2/2)$.
- Bounded support environment: $\Xi_{BS(l,u)} := \{\{P_a\}_{a\in\mathcal{A}} : \ P_a$ has support contained in $[l,u] \ \forall a \in \mathcal{A}\}$. Recall that a real random variable X has distribution with support in $[l,u]$ if holds that $\mathbb{P}\left(X \in [l,u]^c)\right) = 0$.

In the rest of this thesis, we will focus on the class-environment $\Xi_{BS(0,1)}$. All the results we will obtain are trivially extendable to the class-environment case $\Xi_{BS(l,u)}$ for any real interval $[l, u]$ via a rescaling.

Concentration inequalities. Choosing specific class-environments allows us to obtain general bounds on the distribution of rewards. These bounds will be the main tool to build algorithms capable of ensuring good theoretical performance on any instance of the environment-class that the learner may be presented with. We focus on sub-Gaussian distributions, of which limited support distributions are a subclass. Concentration inequalities are inequalities that limit how much the sample mean of a sequence of independent and identically distributed random variables (IIDs) $\{X_i\}_{i=1}^n$ can deviate from its mean value $\mu = \mathbb{E}[X_i]$.

Theorem 2.1.1. *Let $n \in \mathbb{N}$. Let $\{X_i\}_{i=1}^n$ be a set of IID random variable of expected value $\mu := \mathbb{E}[X_i]$, such that $X_i - \mu$ is σ-sub-gaussian. Fixing $\epsilon \geq 0$,*

$$\mathbb{P}(\hat{\mu} \geq \mu + \epsilon) \leq \exp\left(\frac{n\epsilon^2}{2\sigma^2}\right) \qquad \mathbb{P}(\hat{\mu} \leq \mu - \epsilon) \leq \exp\left(\frac{n\epsilon^2}{2\sigma^2}\right) \tag{2.1}$$

Where $\hat{\mu} := \frac{1}{n}\sum_{i=1}^n X_i$ the sample mean of the random sample.

See [15] for the proof. Observe that if a random variable X has zero mean and bounded support in $[l, u]$, then it is $(u - l)/2$-sub-Gaussian. From this observation, combined with the previous theorem, we obtain the following lemma, to which we will make constant reliance throughout the dissertation.

Lemma 2.1.1 (Chernoff-Hoeffding Bound)**.** *Let $\{X_i\}_{i=1}^n$ be a random sample of IID random variables such that $\mathbb{P}(X_i \in [0, 1]^c) = 0$. Fix $\epsilon \geq 0$. Then,*

$$\mathbb{P}(\mu - \hat{\mu} \geq \epsilon) \leq e^{-2\epsilon n^2} \qquad \mathbb{P}(\hat{\mu} - \mu \geq \epsilon) \leq e^{-2\epsilon n^2}. \tag{2.2}$$

Where $\hat{\mu} := \frac{1}{n}\sum_{i=1}^n X_i$ the sample mean of the random sample and $\mu := \mathbb{E}[X]$.

Note that many other concentration inequalities have been proposed in the literature and are exploited to obtain different algorithms for the Multi-Armed Bandit problem. However, in the rest of the thesis, we will rely only on Chernoff-Hoeffding Bound for our results.

2.1.3 Optimality and regret

Once we have introduced the concept of class-environment and the concentration inequality tool, we would like to find a way to verify that a policy ensures good performance on all possible instances of the class-environment. To do this, we need to define a performance measure formally.

As we mentioned, the learner's goal is to maximize the cumulative reward over the time horizon T. We introduce the following notation to indicate the mean of reward distributions. Given an instance $\Psi := \{P_a : a \in \mathcal{A}\}$ and an arm $a \in \mathcal{A}$:

$$\mu_a(\Psi) := \int_{\mathbb{R}} xP_a(dx). \tag{2.3}$$

We denote by $\mu_\star(\Psi) := \max_{a\in\mathcal{A}} \mu_a(\Psi)$, the largest of the averages of the arm rewards.

Remark. Where clear from the context, we will drop the explicit dependence from Ψ to enlighten the notation.

The idea in order to evaluate the performances of a policy is to measure how much the choices imposed by the policy deviate from the optimal choices. The measure we introduce for this evaluation is called regret.

Definition 2.1.1 (Regret)**.** *Given an instance Ψ of the Stochastic Bandit problem and a policy $\mathfrak{U}$, we define the regret of $\mathfrak{U}$ at round $n \in [T]$:*

$$\mathcal{R}_n(\mathfrak{U}, \Psi) := n\mu_\star - \sum_{t=1}^{n} \mathbb{E}\left[X_t\right] \tag{2.4}$$

Where the expectation is with respect to the probability measure induced by the interaction between policy $\mathfrak{U}$ and the environment.

Remark. Where clear from the context, we will drop the explicit dependence from Ψ and $\mathfrak{U}$ to enlighten the notation.

Note that regret of $\mathfrak{U}$ measures the cumulative difference of the expected reward between an optimal policy, i.e. a policy capable of selecting arms with a mean reward equal to $\mu_\star$, and $\mathfrak{U}$.

It is worth remarking that the measure is in expectation with respect to the probability induced by the interaction between policy and environment. This is justified by the fact that rewards are drawn independently at each round. So, for the asymptotic evaluation of the performance, using the mean is legitimated by the law of large numbers. This approach in studying the theoretical performance of algorithms in the Bandits setting was pioneered by the seminal paper [14].

Remark 1. Defining a measure of deviation from optimality is not trivial, and there is no one-size-fits-all solution. There are several definitions of regret in the literature that capture different aspects of the problem. The definition of regret that we have given removes randomness from the measurement. It is possible to define stochastic regret measures in which this randomness is not eliminated. We cite two other formulations:

- $R_n := n\mu_\star - \sum_{t=1}^{n} X_t$, is called random regret.
- $\hat{R}_n := n\mu_\star - \sum_{t=1}^{n} \mu_{A_t}$, is called pseudo-regret.

Random regret is the closest measure to the concept of cumulative stochastic reward. Minimizing R_n is exactly equivalent to maximizing $\sum_{t=1}^{n} X_t$. On the other hand, it suffers from stochasticity due to noise $X_t - \mu_{A_t}$.

The pseudo-regret filters out this noise and coincides with the expectation of random regret conditional to A_t. Since it is a conditional expectation, it is stochastic.

A natural question arises: what bounds for regret can be considered synonymous with policy's good performance under analysis?

First, consider the class environment $\Xi_{BS(0,1)}$. It's easy to state that the worst possible regret of any policy on any instance over T round, is T. Indeed, at each round $t \in [T]$, $\mu_\star - \mathbb{E}[X_t] \leq 1$ *a.s.* for any instance and for any policy.

Thus, a minimum requirement is that the policy ensures sublinear regret for every instance in the environment class, i.e.

$$\forall \Psi \in \Xi, \qquad \lim_{T \to 0} \frac{\mathcal{R}_T(\mathfrak{U}, \Psi)}{T} = 0. \tag{2.5}$$

Observe this implies that the policy is able to learn which is the optimal choice, eventually. Namely, if we consider $T \to +\infty$, $\exists n \in \mathbb{N}$ such that $\forall t \geq n,\ A_t = a_\star$.

In practice, we can hope to satisfy more stringent conditions. We will see how many algorithms manage to guarantee sublinear regret over a finite time horizon, i.e.:

$$\forall T \in \mathbb{N},\ \forall \Psi \in \Xi, \qquad \mathcal{R}_T(\mathfrak{U}, \Psi) \leq C(\Psi) f(T), \tag{2.6}$$

where $C : \Xi \to \mathbb{R}^+$ is a positive function of the instance, and $f : \mathbb{N} \to \mathbb{R}^+$ is a sublinear function of the number of rounds.

Remark 2. Usually, when we look for regret bounds of the form of Inequality (2.6), we can encounter two types of bounds. The first is called *instance-dependent* regret bound. In this case, the function C is dependent on the instance Ψ and, in particular on the set of distribution $\{P_a\}_{a \in \mathcal{A}}$. The second case is known as *instance independent* regret bound. In this case, fixed the size of the problem K and the time horizon T, the regret bound holds in the same exact form for every instance of the environment-class Ξ, namely the value of C is the same for every instance in the environment-class. In the thesis, we will focus on instance independent bound.

Regret decomposition. We now introduce another viewpoint about policy regret. We define for each arm $a \in \mathcal{A}$ the gap from optimality, or instant regret, as

$$\Delta_a = \mu_\star - \mu_a \tag{2.7}$$

Therefore, the following Lemma applies:

Lemma 2.1.2 (Regret Decomposition). *Let $\mathfrak{U}$ be a policy over the environment class Ξ. The regret on time horizon $T \in \mathbb{N}$ of the policy can be decomposed as*

$$\mathcal{R}_T = \sum_{a \in \mathcal{A}^+} \Delta_a \mathbb{E}\left[N_T(a)\right] \tag{2.8}$$

Where we denote with $\mathcal{A}^+ := \{a \in \mathcal{A} : \mu_a < \mu_\star\}$ the set of sub-optimal arms and with $N_T(a)$ the random variable that counts the number of times arm a is pulled during the whole horizon T.

See [15] for the proof.

This Lemma provides both a different insight into regret and a practical tool used extensively throughout the thesis. Intuitively, we describe regret by summing the times a suboptimal arm is prescribed to be played, weighted by the relative optimality gap.

Regret lower bounds. On the other hand, one should wonder *what is the best regret that a policy can ensure?* This is a much harder question. The following theorem by [14] provides an asymptotic instance dependent lower bound:

Theorem 2.1.2 (MAB lower bound, Lai&Robbins, 1985). *Given an instance $\Psi = \{P_a : a \in \mathcal{A}\}$ of the Stochastic MAB problem, any policy $\mathfrak{U}$ satisfies:*

$$\lim_{T \to +\infty} \frac{\mathcal{R}_T}{\log(T)} \geq \sum_{a \in \mathcal{A}^+} \frac{\Delta_a}{\mathit{KL}\left(P_a, P_\star\right)} \tag{2.9}$$

Where $\mathit{KL}\left(P_a, P_\star\right)$ is the Kullback-Leiber[1] divergence between the distribution of arm a and the distribution of the optimal arm, and $\mathcal{A}^+ := \{a \in \mathcal{A} : \mu_a < \mu_\star\}$ is the set of sub-optimal arms.

[1] See [16] for a formal introduction.

This powerful result tells us that we can not find an algorithm that ensures a regret asymptotically lower than a logarithm function of T on every instance of the Stochastic MAB problem.

However, since we focus on finite time analysis and instance independent bounds, the following theorem by [8] give us a more practical result to understand what we can hope to achieve.

Theorem 2.1.3 (MAB instance independent lower bound)**.** *Consider the environment-class $\Xi_{BS(0,1)}$. Fix a time horizon T. For any policy $\mathfrak{U}$ there exist a problem instance $\Psi \in \Xi_{BS(0,1)}$ such that*

$$\mathcal{R}_T \geq \frac{1}{20} \min \left\{ \sqrt{KT}, T \right\} \tag{2.10}$$

Where $K \in \{2, 3, \dots\}$ is the cardinality of $\mathcal{A}$.

As we will shortly see, we are able to find algorithms that reach, at least as order of magnitude, the theoretical bounds.

2.1.4 Bandit algorithms

Having the tools to evaluate the algorithms' performance, we present some algorithms known from the literature. The results presented apply to our interest's environment-class $\Xi_{BS(0,1)}$. Among these particularly important is the UCB1 algorithm, based on the principle of Optimism in the Face of Uncertainty: the same principle inspires the algorithms that we will propose in the thesis.

Explore-first The idea beyond this algorithm is to separate the exploration and exploitation parts. The algorithm is given as input a natural number n, representing the algorithm's number of times it must sample each arm. So the algorithm explores for nK rounds, chooses the best arm with respect to the collected data, and plays the selected arm for the remaining $T - nK$ rounds. We denote by $\hat{\mu}_a(t)$ the average reward obtained by the arm a in round t.

$$\hat{\mu}_a(t) = \frac{1}{N_a(t)} \sum_{j=1}^{t} X_j \mathbb{1} \left\{ A_t = a \right\} \tag{2.11}$$

Algorithm 1 Explore-first

```
Input: T, $\mathcal{A} = [K]$, $n$
for $t$=1,...,$nK$ do
    Play action:
        $A_t = (t \mod K) + 1$
Set $\bar{a} := \text{argmax}_{a \in \mathcal{A}} \; \hat{\mu}_a(nK)$
for $t$=$nK$+1,...,$T$ do
    Play action $\bar{a}$
```

For this simple algorithm, we manage to obtain an instance independent regret bound, sublinear in the number of rounds T.

Theorem 2.1.4 (Explore-first instance independent regret upper bound)**.** *Given any instance of the K-Armed Stochastic Bandit problem, running Algorithm (1) ensures to obtain a total regret over the T rounds bounded by*

$$\mathcal{R}_T \leq T^{2/3} \mathcal{O} \left(K \log(T) \right)^{1/3} \tag{2.12}$$

Choosing $n = (T/K)^{2/3} \log(T)^{1/3}$.

We refer to [19] for the proof. Note that, even if Algorithm (1) ensures sublinear regret, it is far from the order of magnitude of the lower bound introduced with Theorem (2.1.3).

Optimism in the Face of Uncertainty and UCB1. To motivate the idea of Optimism in the Face of Uncertainty (OFU), we need to take a step back to the exploration-exploitation dilemma. Any Reinforcement Learning algorithm has to explore the set of possible choices while taking advantage of the knowledge collected. But, how to balance these two aspects? The idea of OFU is: despite our lack of knowledge about which are the best actions, we will construct an *optimistic guess* on their expected payoff. Then we choose the action a with the highest guess among all the possible arms. If the resulting realization of the reward is bad, then the value of our future optimistic guesses on the reward of a will quickly decrease, and we will be compelled to switch to a different action. On the other hand, if we pick well, we will be able to exploit that action and incur little regret. In this way, we balance exploration and exploitation.

We want now to translate the idea of OFU into a policy available to the decision-maker. The main point is *how to construct the optimistic guess.* To do this, we will rely on confidence intervals: we construct a confidence interval on the mean reward for every arm. The optimistic guess will be the upper bound of this interval. Each time an arm is played, and a new sample of its reward is observed, we update the confidence interval according to the incoming information. To construct the confidence interval on mean reward, we will rely on the previously introduced Chernoff-Hoeffding's Bound, Lemma (2.1.1). If we have a random sample $\{X_t\}_{t=1}^n$ and a fixed probability level $p \in [0, 1]$, then applying Lemma (2.1.1)

$$\mathbb{P}\left(|\mu - \hat{\mu}| \geq \sqrt{\frac{2\log(1/p)}{n}}\right) \leq 2p \tag{2.13}$$

where $\hat{\mu} := \frac{1}{n}\sum_{i=1}^n X_i$ and $\mu = \mathbb{E}[X_i]$. Now, the main idea is to exploit this fact on the independently sampled rewards and let p decrease as a function of number of rounds. Based on this idea we define for every $t \in [T]$, $a \in \mathcal{A}$:

$$\texttt{UCB}_a(t-1) := \begin{cases} +\infty & \text{if } N_a(t-1) = 0 \\ \hat{\mu}_a(t-1) + \sqrt{\frac{2\log(t)}{N_a(t-1)}} & \text{otherwise} \end{cases} \tag{2.14}$$

Where $\hat{\mu}_a(t)$ is defined in Equation (2.11) and $N_a(t)$ is the number of times a has been sampled up to round t. This quantity is called upper-confidence bound of arm a at round t.

Given the Definition (2.14), we are ready to formulate the algorithm called UCB1.

Algorithm 2 UCB1

```
Input: $\mathcal{A} = [K]$ arms set, $T \geq K$ time horizon
for $t$= 1,...,$T$ do
    Choose action $A_t = \text{argmax}_{a \in \mathcal{A}} \texttt{UCB}_a(t-1)$
    Observe reward $X_t$ and update upper confidence bounds.
```

Observe that the upper confidence bound (2.14) shrinks with the number of times the arm is sampled. This accounts for the principle of being optimistic in the face of uncertainty.

We now present a performance analysis of the UCB1 algorithm. In [2], regret bounds are presented in the case of finite time horizon analysis. The following theorem provides an instance-dependent regret bound.

Theorem 2.1.5 (Instance dependent regret bound for UCB1)**.** *Fix a time horizon $T \in \mathbb{N}$. Given any instance Ψ of the K-Armed Stochastic Bandit problem in the class-environment $\Xi_{BS(0,1)}$, running Algorithm (2) on Ψ implies a regret that is bounded by:*

$$\mathcal{R}_T \leq \left[8\log(T)\sum_{a \in \mathcal{A}^+}\frac{1}{\Delta_a}\right] + \left(1 + \frac{\pi^2}{3}\right)\left(\sum_{a \in \mathcal{A}^+}\Delta_a\right) \tag{2.15}$$

where $\mathcal{A}^+ := \{a \in \mathcal{A} : \Delta_a > 0\}$, and Δ_a is the optimality gap of arm a.

We observe how, despite the simplicity of the UCB1 algorithm, we are able to obtain an instance-dependent upperbound that reaches a logarithmic order of magnitude in the number of rounds T. This order of magnitude coincides with that of the theoretical lower bound presented in Theorem (2.1.2). Improvements have been proposed in the literature to lower bound multiplicative constants. Then, we provide an instance independent regret bound for the UCB1 algorithm, derivable from the previous theorem.

Lemma 2.1.3 (Instance independent regret bound for UCB1)**.** *Fix a time horizon $T \in \mathbb{N}$. Given any instance Ψ of the K-Armed Stochastic Bandit problem in the class-environment $\Xi_{BS(0,1)}$, running Algorithm (2) on Ψ implies a regret that is bounded by:*

$$\mathcal{R}_T \leq 5\sqrt{KT\log(T)} + 8K \tag{2.16}$$

In this case, we observe that the theoretical performance limit provided by the Theorem (2.1.3) is reached, as an order of magnitude, by the Algorithm (2) up to a factor $\sqrt{\log(T)}$. These results legitimize the use of techniques that mimic the OFU principle in the algorithms that we will propose in the next chapters.

Bayesian approach. Finally, we mention the existence of algorithms that are based on a Bayesian approach. Among these, the best known is Thompson Sampling, introduced in [21]. Giving bounds on the regret of algorithms like Thompson Sampling is technical and out of the dissertation's objectives. However, in many empirical applications, it is worth mentioning that the use of a Bayesian algorithm like Thompson Sampling can outperform frequentist algorithms like UCB1. We refer to [6] for some experimental comparisons between the two types of algorithm.

2.1.5 Applications

The Stochastic MAB framework is widely applied to all contexts where the exploration-exploitation dilemma plays a crucial role. We present a non-exhaustive list of application examples known in the literature. We will give special attention to the case of advertising, presented last in this list. We delegate further discussion to [15].

Clinical trials. Testing and refining a clinical treatment implies being able to balance the effectiveness of the treatment on the patients involved and collecting information that can improve the performance for future patients. From a modeling point of view, we can put the problem in the MAB context as follows. Each new patient is considered a new round of the sequential decision problem. In each round, the decision-maker must administer a treatment from an available $\mathcal{A}$ set of treatments. We assume that the patient's reaction to the treatment is numerically quantifiable in the range $[0, 1]$. Each treatment is thus an arm of the Multi-Armed Stochastic Bandit problem, and the patient's response to the treatment is the learner's reward.

Dynamic pricing. Many online retailers (e.g. airlines, online learning platforms) use the dynamic pricing technique. This consists of selecting the price of their products in real-time, trying to collect information about the demand for their products in order to maximize profits in the long run. We insert as follows the problem in the Bandit context. Fix a product to be priced. Each new user on the product page is considered a new round of the sequential decision problem. The retailer can choose in an $\mathcal{A}$ which price to show to the customer. The customer may or may not buy the product: the reward coincides with the customer's expense. Considering a discrete and finite range of possible prices for the product, the problem can be framed in the Multi-Armed Bandit context with class-environment $\Xi_{BS(l,u)}$ where l and u are respectively the minimum and maximum applicable prices for the product.

Advertising. The problem addressed in this dissertation is that of advertising campaign selection. In its simplest form, the problem can be reduced as follows to the Multi-Armed Bandit framework. The advertiser is presented with a set of possible campaigns $\mathcal{A}$ that differ, for example, in target, content or price-per-click. In each round, the decision-maker has to select one of the campaigns. We assume that each ad campaign is associated with a stochastic profit of fixed mean and bounded support distribution. We thus obtain a Multi-Armed Bandit problem in the class-environment $\Xi_B S$. It is important to underline that this formulation neglects many aspects of the real problem. First, we assume that the decision-maker has no budget constraints at each round. This means that the algorithm will search for solutions with the highest expected profit even in the face of very high risk-exposure due to costs. The second problem was presented in [10]: one of the major business constraints found in campaign allocation is to keep the Return on Investment above a given threshold at each round. We will address these issues by extending the Multi-Armed Bandit framework in later chapters. Finally, there is a problem with Bandit feedback. Usually, an advertiser is not tasked with selecting a single ad campaign each round: an ad campaign is composed of many sub-campaigns. The combination of these must be optimally allocated. We will address this problem by introducing the framework of Multi-Task Stochastic Semi-Bandit.

2.2 Multi-Task Stochastic Semi-Bandit

This section aims to present the extension of the MAB framework known as Multi-Task Stochastic Semi-Bandit. This will be the starting point for extending the MAB problem's results with cost and ROI constraints to the case where multiple sub-campaigns are present in the advertising problem. We will constructively introduce this framework. First, we will present the Multi-Task Bandit framework as a particular case of the more general Combinatorial Multi-Armed Bandit (CMAB) framework. As done for the MAB case, we will introduce the framework by describing it as a sequential game between the decision-maker and the environment. Second, we introduce the idea of Semi-Bandit feedback, emphasizing its difference from Bandit feedback. Then, we examine the algorithm CombUCB to deal with Combinatorial Bandits, providing related results regarding its theoretical performances. We conclude the section with two examples of practical applications.

2.2.1 Combinatorial MAB and Multi-Task Bandits

Combinatorial MAB. As we did for the MAB case, we describe the combinatorial Multi-Armed Bandit problem as a sequential game between a player (i.e., learner or decision-maker) and the environment. The game consists of $T \in \mathbb{N}$ rounds. In each round, the player selects an element, called a superarm, from the set

$$\mathcal{S} \subset \left\{ \mathbf{a} \in \{0,1\}^d : \sum_{i=1}^{d} a_i \leq L \right\} \tag{2.17}$$

Namely a superarm is a binary vector of length d with at most L entries equal to 1. As a result of his choice, the decision-maker receives a stochastic reward X_t, which is linear in the selected superarm, i.e.:

$$X_t = \langle \mathbf{A}_t, \boldsymbol{\mu}_{\mathbf{A}_t} + \boldsymbol{\eta}_t \rangle \tag{2.18}$$

The unknown vector $\boldsymbol{\mu}_{\mathbf{A}_t} \in \mathbb{R}^d$ is the vector of expected rewards relative to the superarm $\mathbf{A}_t$. The vector $\boldsymbol{\eta}_t$ is the source of the reward's randomness. It is a white noise that we assume to have unknown distribution of support bounded in $[0,1]^d$. The learner's goal is to maximize the cumulative reward over the time horizon T.

Framework protocol 2 Combinatorial Multi-Armed Bandit (CMAB)

```
Input: $T$ time horizon, $d$ number arms, $\mathcal{S} \subset \{0,1\}^d$ set of superarms
for: $t = 1, ..., T$
    1. Select a superarm $\mathbf{A}_t \in \mathcal{S}$
    2. Collect the reward $X_t = \langle \mathbf{A}_t, \boldsymbol{\mu}_{\mathbf{A}_t} + \boldsymbol{\eta}_t \rangle$
```

We observe that the problem can be seen as a particular case of the K-Armed Bandit problem. It is sufficient to consider every possible superarm of the combinatorial problem as an arm of a K-Armed Bandit problem. The class-environment of this problem is given by the assumptions made on the distribution of white noise $\{\boldsymbol{\eta}_t\}_{t=1}^T$. However, doing this, the number of arms K grows exponentially in d.

Multi-Task Bandits. To study the advertising problem, we focus on a special case of CMAB known in the literature as Multi-Task Bandits (see [15]). We also describe this as a sequential game over $T \in \mathbb{N}$ rounds. We imagine playing $M \in \mathbb{N}$ different problems K-Armed Stochastic Bandit games simultaneously. We call each of the M problems *task*. In each round, the decision-maker selects a superarm $\mathbf{A}_t \in [K]^M$. The m^{th} component of the superarm $\mathbf{A}_t$indicates which of the K actions is selected in the m^{th} task. After making a choice, the player receives a reward:

$$X_t = \sum_{m=1}^{M} X_t^{(m)} \tag{2.19}$$

where we denote by $X_t^{(m)}$ the partial reward of the m^{th} task. We emphasize that in the case of Bandit feedback, the player does not observe directly $X_t^{(m)}$, but only receives the value of the total reward of the M tasks.

Framework protocol 3 Multi-Task Stochastic Bandit (MTSB)

```
Input: $T$ time horizon, $K$ number arms for each task, $M$ number of tasks,
        $\mathcal{S} = [K]^M$ set of superarms
for: $t = 1, ..., T$
    1. Select a superarm $\mathbf{A}_t \in \mathcal{S}$
    2. Collect the reward $X_t = \sum_{m=1}^{M} X_t^{(m)}$ where $X_t^{(m)}$ is the unknown
       partial reward of the $m^{th}$ task.
```

It is worth underlining that the Multi-Task Bandit case can be framed as a special case of the CMAB problem. Taking in fact $d = MK$ it is sufficient to define the set in Formula (2.17) as

$$\mathcal{S} := \left\{ \mathbf{a} \in \{0,1\}^d : \sum_{i=1}^{K} a_{i+jK} = 1 \, \text{for all} \; 0 \leq j < M \right\} \tag{2.20}$$

Namely, we partition the d components into M disjoint sets of cardinality K and impose that the player must choose one and only one element from each set. This observation legitimizes proposing CMAB algorithms to solve Multi-Task Bandit problems.

2.2.2 Semi-Bandit feedback

We introduce in the Multi-Task Bandit problem the idea of Semi-Bandit feedback, We describe the sequential game associated with the Multi-Task Stochastic Semi-Bandit (MTSSB) setting. At each round, a superarm $\mathbf{A}_t$ is selected. The m^{th} component of $\mathbf{A}_t$ is the selected arm of the m^{th} task. For each task, the associated reward $X_t^{(m)}$ is revealed. The reward collected in round t is the sum of the individual rewards of each task.

Framework protocol 4 Multi-Task Stochastic Semi-Bandit (MTSSB)

```
Input: $T$ time horizon, $K$ number arms for each task, $M$ number of tasks,
       $\mathcal{S} = [K]^M$ set of superarms
for: $t = 1, ..., T$
  1. Select a superarm $\mathbf{A}_t \in \mathcal{S}$
  2. Observe a partial reward $X_t^{(m)}$ for every task $m \in [M]$
  3. Collect the reward $X_t = \sum_{m=1}^{M} X_t^{(m)}$
```

The Semi-Bandit feedback arises in the revelation not only of superarm's reward but of all its components. Note that, in this case, it is no longer possible to trace the problem back to a single MAB: the Bandit feedback is characterized by the fact that the learner does not receive any information other than the total reward of the just-completed round.

2.2.3 Assumptions, regret, correlation

We now extend the concepts and assumptions introduced in the MAB framework to the Multi-Task Stochastic Semi-Bandit (MTSSB) case.

Notation.

- $X_t^{(m)}$, *partial reward of the* m^{th} *task at round* t.
- $\boldsymbol{X}_t := \left[X_t^{(1)}, X_t^{(2)}, \ldots, X_t^{(M)}\right]$, *vector of partial rewards at round* t.

- $X_t := \sum_{m=1}^{M} X_t^{(m)}$, *reward at round* t.
- $\mu_{\boldsymbol{a}}$, *of distribution* $P_{\boldsymbol{a}}$
- $A_t^{(m)}$, m^{th} *component of superarm* $\boldsymbol{A}_t$.

Instance of a MTSSB problem. We call instance of a Multi-Task Stochastic Semi-Bandit problem a set of distributions $\Psi = \{P_{\mathbf{a}} : \mathbf{a} \in \mathcal{S}\}$. We assume that the distribution of the vector of partial rewards $\mathbf{X}_t$, conditional on the sequence of actions-feedbacks $\{\mathbf{A}_1, \mathbf{X}_1, \ldots, \mathbf{A}_{t-1}, \mathbf{X}_{t-1}, \mathbf{A}_t\}$, is $P_{\mathbf{A}_t}$.

Environment-class. We denote the environment-class of a MTSSB problem a collection of instances

$$\Xi := \{\Psi = \{P_{\mathbf{a}} : \ \mathbf{a} \in \mathcal{S}\} : \ P_{\mathbf{a}} \in \mathcal{M}_{\mathbf{a}}, \, \forall \mathbf{a} \in \mathcal{S}\}$$

where $\mathcal{M}_{\mathbf{a}}$ is a set of possible distributions of the vector of partial rewards associated with the superarm $\mathbf{a} \in \mathcal{S}$.

The examples of environment-classes presented in the previous section for the MAB problem extend naturally to the multivariate case. In particular, in the course of the thesis, we focus on the environment-class $\Xi_{BS([0,1]^M)}$, whose distributions have limited support contained in the hypercube $[0,1]^M$.

Policy. The distribution of $\mathbf{A}_t$ given the sequence $\{\mathbf{A}_1, \mathbf{X}_1, \ldots, \mathbf{A}_{t-1}, \mathbf{X}_{t-1}\}$ is $\pi_t(\cdot|\mathbf{A}_1, \mathbf{X}_1, \ldots, \mathbf{A}_{t-1}, \mathbf{X}_{t-1})$. The sequence $\mathfrak{U} := \{\pi_t\}_{t=1}^{T}$ is called policy and characterizes the decision-maker.

Regret. Given an instance Ψ of the MTSSB problem and a policy $\mathfrak{U}$, we define the regret of $\mathfrak{U}$ at round $n \in [T]$:

$$\mathcal{R}_T(\mathfrak{U}, \Psi) := n\mu_\star - \sum_{t=1}^{n} \mathbb{E}\left[X_t\right] \tag{2.21}$$

Where $X_t := \sum_{m=1}^{M} X_t^{(m)}$ and $\mu_\star := \operatorname{argmax}_{\mathbf{a}\in\mathcal{S}} \mu_{\mathbf{a}}$. The expectation with respect to the probability measure induced by the interaction between $\mathfrak{U}$ and environment.

We emphasize how, even if we have information about the partial rewards $X_t^{(m)}$, the regret is evaluated on the total reward of each round X_t.

Correlation. An important assumption to discuss is the one about correlation between the tasks in the problem. For each $m \in [M]$, $\mathbf{a} \in \mathcal{S}$, we denote by $P_{\mathbf{a}}^{(m)}$ the marginal distribution of the m^{th} component of the vector of partial rewards associated with the superarm $\mathbf{a} \in \mathcal{S}$, which has distribution $P_{\mathbf{a}}$.

It is well known from probability theory that from the knowledge of the joint distribution $P_{\mathbf{a}}$, one can obtain the marginal distribution of the components $P_{\mathbf{a}}^{(m)}$. However, it is not possible in general from $\left\{P_{\mathbf{a}}^{(m)}\right\}_{m=1}^{M}$ to trace back to $P_{\mathbf{a}}$. Nevertheless, when we assume independence between the components, the joint measure coincides with the product measure. So if we assume independence between the M tasks of the problem, we could characterize each instance as $\Psi = \{P_i^{(m)} : i \in [K], m \in [M]\}$, i.e., indicating the marginal distributions of the components of each possible superarm.

Still, because of the way the MTSSB framework is defined, assuming independence between superarm components is equivalent to considering M separate K-Armed Bandit problems. In this case, the best that can be done is to use in parallel M times a policy for the MAB problem, each taking into into account only the partial reward $X_t^{(m)}$. Independence between components, in fact, implies that the optimal superarm is the one whose components are optimal arms for the individual tasks considered separately.

However, if we add constraints to the problem, this statement no is no longer verified. We will see how in a constrained context, e.g., with budget or ROI constraints, a superarm that is optimal in terms of reward can have components that turn out to be suboptimal when considering individual unconstrained tasks separately.

2.2.4 Combinatorial UCB algorithm

Several algorithms for the CMAB problem with Semi-Bandit feedback, of which MTSSB is a particular case, have been explored in the literature. We propose below an algorithm inspired by UCB1, studied in the paper [13] and known as CombUCB. We refer to the notation used for the CMAB framework recalling that we can apply it to the MTSSB framework as a particular case.

The algorithm follows the upper confidence bound idea introduced with the UCB1 algorithm. For each of the arms $i \in [d]$, we construct an upper confidence bound $\texttt{UCB}_i$ defined as follows.

$$\texttt{UCB}_i(t-1) := \begin{cases} +\infty & \text{if } N_i(t-1) = 0 \\ \hat{\mu}_i(t-1) + \sqrt{\frac{2\log(t)}{N_i(t-1)}} & \text{otherwise} \end{cases} \tag{2.22}$$

Where $N_i(t-1)$ is the random variable that counts the number of times arm i has played an active role in one of the arms selected up to round

$t-1$, namely $N_i(t-1) := \#\{s \in [t-1] : A_s^{(i)} = 1\}$. We denote with $\hat{\mu}_i(t-1) := \frac{1}{N_i(t-1)} \sum_{s=1}^{t-1} X_s^{(i)} A_s^{(i)}$ the sample mean of the partial reward i^{th} based on the samples collected up to round $t-1$.

The algorithm's idea is to select the superarm $\mathbf{A}_t$ optimistically. From the set of superarms $\mathcal{S}$, we select the one whose sum of `UCB`s associated to active components is maximal.

Algorithm 3 CombUCB

```
Input: $d$ number of arms, $\mathcal{S} \subset \{0,1\}^d$ set of superarms, $T$ time horizon
for $t$= 1,...,$T$ do
    Choose superarm $\mathbf{A}_t = \text{argmax}_{\mathbf{a}\in\mathcal{S}} \sum_{j=1}^{d} a^{(j)} \texttt{UCB}_{a^{(j)}}(t-1)$
    Observe partial rewards $X_t^{(j)}$ for each $j \in \{j \in [d] : \mathbf{A}_t^{(j)} = 1\}$ and
      update upper confidence bounds of each arm $i \in [d]$.
```

The algorithm turns out to be a simple extension of UCB1, but, like UCB1, it manages to guarantee excellent theoretical bounds on total regret.

Theorem 2.2.1 (Instance dependent regret bound for CombUCB)**.** *Fix a time horizon $T \in \mathbb{N}$, a number of arms $d \in \mathbb{N}$ and a superarm set $\mathcal{S} = \{\boldsymbol{a} \in \{0,1\}^d : \|\boldsymbol{a}\|_1 \leq L\}$ for some $L \in [d]$. For any instance of the stochastic Combinatorial Semi-Bandit problem, the regret of Algorithm (3) is bounded as:*

$$\mathcal{R}_T \leq \sum_{i:\Delta_i>0} d\frac{534}{\Delta_{i,min}} \log(T) + \left(\frac{\pi^2}{3}+1\right) Ld$$

Where $\Delta_{i,min}$ is minimum gap between a suboptimal superarm that contains arm i and the optimal superarm.

We refer to [13] for the proof and further discussion. In addition, [13] provides the following instance independent bound for regret:

Theorem 2.2.2 (Instance independent regret bound for CombUCB)**.** *Fix a time horizon $T \in \mathbb{N}$, a number of arms $d \in \mathbb{N}$ and a superarm set $\mathcal{S} = \{\boldsymbol{a} \in \{0,1\}^d : \|\boldsymbol{a}\|_1 \leq L\}$ for some $L \in [d]$. For any instance of the stochastic combinatorial Semi-Bandit problem, the regret of Algorithm (3) is bounded as:*

$$\mathcal{R}_T \leq 47\sqrt{T\log(T)Ld} + \left(\frac{\pi^2}{3}+1\right) Ld$$

We will use an extension of this algorithm, with constraints added, for the advertising problem with multiple sub-campaigns.

2.2.5 Applications

The Multi-Armed Combinatorial Bandit problem is reflected in several application examples of sequential decisions. We present two of them below, placing particular attention to the advertising case that we will extend in the thesis.

Social influence maximization. In a social influence maximization problem [12], is given a directed graph $G = (V, E)$, with a finite set of vertices V and set of edges $E \subset V \times V$. Each edge (i, j) is associated with an unknown *probability of influence propagation.* At the beginning of each round, a subset $S \subset V$ is selected and the nodes in S are *activated.* The subset S is called *seed* of round t. Starting from the seed, starts a propagation process, which lasts n iterations. At each iteration an active node i has a probability $p_{i,j}$ of activating an inactive node j connected to i via the edge (i, j). At the end of the propagation process, the round reward is equal to the number of active nodes on the graph. The problem is to find the set S, formed by at most L nodes, that maximizes the expected reward. The problem can be put in the CMAB framework: we do not know the probability of influence propagation of each edge, and we want to learn them by iteratively setting a subset of the vertices as seed. At the same time, we try to maximize the cumulative reward in T rounds. We can denote each edge as an arm of the CMAB problem. A superarm coincides with a set of edges exiting from at most L nodes.

Advertising problem. The Multi-Task Stochastic Semi-Bandit framework allows us to model the advertising problem in the case where the advertiser has to choose at each round an advertising campaign consisting of multiple subcampaigns. We denote by M the number of subcampaigns. In each of the M sub-campaigns, the advertiser must choose between K ads that differ, for example, in the target audience, keywords, or price-per-click. A stochastic profit is associated with each of the ads. The learner selects at each round one and only one ad for each of the M sub-campaigns, observes the profit derived from each and collects the total profit, namely the sum of each subcampaign's profit. The learner's goal is to maximize the cumulative profit over the T rounds of the sequential game. Although this is an extension of the MAB case model, it is not without limitations. The main and already mentioned ones are the lack of constraints on cost and ROI. In Chapter (5), we will propose algorithms that take in account these constraints.

Chapter 3

Problem formulation

This chapter formally introduces the constrained problems that allow us to model the advertising problem with daily requirements on budget and ROI constraints. First, we present the Multi-Armed Bandit framework with cost feedback (CostMAB), which extends the MAB problem by introducing a feedback on the cost separate from feedback on the reward. Then, we define in the CostMAB framework the idea of safe policy. Selecting algorithms in the safe class aims to ensure that constraints are violated, in expectation, a sublinear number of times. Unfortunately, the impossibility theorem presented in [5] states that if an algorithm belongs to the safe class, it cannot ensure sublinear bounds on the regret.

This tradeoff theorem motivates the entire dissertation: to overcome the limitation of safe policies, we introduce the concept of quasi-safe policies. These are algorithms that admit sublinear bounds for regret while maintaining sublinear the expected number of times the constraints are violated no more than a certain tolerance threshold.

We will focus on budget and ROI constraints, presenting them separately to have a more understandable analysis of the algorithms in the following chapters.

We extend the problem formulation to the Multi-Task Stochastic Semi-Bandit framework with cost feedback (CostMTSSB). Again, we will define the idea of quasi-safe learning, both for daily budget constraints and daily ROI constraints.

Finally, we show how we can frame the advertising bid optimization problem into the CostMAB framework –in the case of a single advertising campaign– and into the CostMTSSB framework –in the case of multiple advertising sub-campaigns–.

3.1 Cost Multi-Armed Bandit (CostMAB)

We extend the stochastic K-Armed Bandit framework by introducing cost feedback on selected arms. We describe the problem as a sequential game between the decision-maker and the environment, played over $T \in \mathbb{N}$ rounds. At each round $t \in [T]$, the player selects an arm A_t from the set $\mathcal{A} = [K]$. The environment then returns as feedback the pair (X_t, Y_t). X_t represents the reward obtained by the decision-maker, Y_t the cost incurred by the player. Both are stochastic quantities extracted from distributions not known to the learner.

Basic assumptions. We introduce the basics assumptions of the model.

We define instance of a stochastic cost MAB problem, a set

$$\Psi := \{(P_a, Q_a) : a \in \mathcal{A}\}$$

Where, for any arm $a \in \mathcal{A}$, P_a represents the distribution of arm's reward, while Q_a represents the distribution of arm's cost. Namely, if in round t the player selects the action A_t, the environment will sample the reward X_t from the distribution P_{A_t} and the cost Y_t from the distribution Q_{A_t}. Observe that the interaction between agent and environment induces a probability measure on the sequence of actions-feedback $\{A_1, (X_1, Y_1), \dots, A_T, (X_T, Y_T)\}$. We assume that the sequence of action-feedback satisfies the following hypotheses:

- the conditional distribution of the reward X_t given the sequence of feedback actions up to round t, $\{A_1, (X_1, Y_1), \dots, (X_{t-1}, Y_{t-1}), A_t\}$, is P_{A_t}.
- the conditional distribution of the reward Y_t given the sequence of feedback actions up to round t, $\{A_1, (X_1, Y_1), \dots, (X_{t-1}, Y_{t-1}), A_t\}$, is Q_{A_t}.
- the distribution of A_t given the sequence $\{A_1, (X_1, Y_1), \dots, A_{t-1}, (X_{t-1}, Y_{t-1})\}$ is $\pi_t(\cdot | A_1, (X_1, Y_1), \dots, A_{t-1}, (X_{t-1}, Y_{t-1}))$.
  The sequence $\mathfrak{U} := \{\pi_t\}_{t=1}^{T}$ is called policy and characterizes the decision-maker.

The first two hypothesis summarize the idea that the environment samples the value of X_t from P_{A_t} and the value of Y_t from Q_{A_t}. The last assumption requires that the player's actions to be selected on and only based on the history before round t.

Framework protocol 5 Cost Multi-Armed Bandit (CostMAB)

```
Input: $T$ time horizon, $K$ number arms, $\mathcal{A} = [K]$ set of arms
for: $t = 1, ..., T$
  1. Select an arm $A_t \in \mathcal{A}$
  2. Collect the reward $X_t$ sampled from $P_{A_t}$.
  3. Suffer a cost $Y_t$ sampled from $Q_{A_t}$
```

Notation. *Fix $a \in \mathcal{A}$, we define the following quantities:*

- *μ_a, mean of the distribution P_a.*
- *ν_a, mean of the distribution Q_a.*

Throughout this thesis, we focus on the environment-class $\Xi_{BS(0,1)}$, i.e., the class of instances for which the reward and cost distributions have limited support in the real interval $[0, 1]$.

3.1.1 Safe learning: daily budget constraints

To find performant algorithms that account for daily budget constraints, we need to define two elements. The first is an optimal solution against which evaluate a proposed algorithm, leveraging the idea of regret. The second element is a class of algorithms considered *safe*, among which we will look for those with good performances in terms of regret.

Optimal constrained solution. Let $b \in (0, 1]$ the daily budget required by the problem. We say that the sequential game defined by Protocol (5) is subject to budget constraints if the policy $\mathfrak{U}_\star$ considered optimal prescribes at each round $t \in T$ the arm $a_\star \in \mathcal{A}$ solution of the optimization problem:

$$\underset{a \in \mathcal{A}}{\operatorname{argmax}}\ \mu_a \tag{3.1}$$

$$\text{s.t. } \nu_a \leq b \tag{3.2}$$

Namely, it selects the arm with the highest expected reward among those whose expected cost is lower than the budget.

Regret. Once defined the optimal constrained policy, we can define the regret to measure the performances in terms of cumulative reward.

Definition 3.1.1. *Let $\Psi := \{(P_a, Q_a) : a \in \mathcal{A}\}$ be an instance of the budget constrained CostMAB problem. Given the policy $\mathfrak{U}$, we call regret of $\mathfrak{U}$ over the time horizon T the quantity:*

$$\mathcal{R}_{\mathfrak{U}}(T) = T\mu_\star - \sum_{t=1}^{T} \mathbb{E}\left[X_t\right] \tag{3.3}$$

Where $\mu_\star$ is the mean reward of the arm prescribed by the optimal policy $\mathfrak{U}_\star$ defined in Equation (3.1), and the expectation is with respect to the probability measure induced by the interaction between policy $\mathfrak{U}$ and the environment.

Observe that this definition of regret is not substantially different from the one of MAB framework. The only difference, in which the budget constraint appears, is the definition of the optimal policy. This fact implies pros and cons. On the one hand, this suggests that we should try to apply techniques similar to those proposed in the MAB literature to find algorithms with theoretical bounds on the regret. On the other hand, minimizing the regret is clearly not enough to find a satisfying solution to the problem: a policy can in general achieve an arbitrarily small regret by violating the constraints many times. Violating the constraint makes it possible to outperform the reward of optimal policy in terms of regret.

On the other side, the stochasticity of costs makes impossible to ensure a priori that a policy will never violate Constraint (3.2). Thus, it arises the need of shrinking the class of possible algorithms, adding some conditions that account for the objective of not violating Constraint (3.2) too often.

Budget safe policies. We now formalize the concept of not violating the budget constraint too often. The metric to address this problem is the expected number of constraint violations.

Definition 3.1.2 (Expected number of budget constraint violations)**.** *Let Ψ be an instance CostMAB problem with daily budget b. Fix policy $\mathfrak{U}$ and a time horizon T. We define the expected number of constraint violations of $\mathfrak{U}$ over T rounds:*

$$\mathbb{E}\left[J_T\right] := \mathbb{E}\left[\#\{t \in [T] : \nu_{A_t} > b\}\right] \tag{3.4}$$

Where A_t is the action selected at round t by policy $\mathfrak{U}$, and ν_a is the mean of the distribution Q_a.

Leveraging this definition we can now specify the class of *η-safe policy*

Definition 3.1.3 (Budget η-safe policy). *Fix $\eta \in (0,1)$. The policy $\mathfrak{U}$ is said to be η-safe if*

$$\mathbb{E}\left[J_T\right] \leq \eta T \tag{3.5}$$

Namely, the expected number of violations of Constraint (3.2) is upper-bounded by ηT.

Note that any η-safe policy ensures that, with probability at least $1-\eta$, the policy does not violate the constraint of the optimization problem. A satisfying policy for the problem would combine the request of sublinear growth of the regret and a requirement of η-safety.

Unfortunately, in the next section we will illustrate the result by [5] that proves the impossibility of building such a policy. So we'll need to relax our requests.

3.1.2 Safe learning: daily ROI constraint

Similar to what we have done in the case of daily budget constraints, we define a constrained optimization problem the that allows us to take into account daily demands on the Return on Investment. Then, we define a class of ROI safe algorithms.

Remark 3. We recall shortly the definition and the role played by the ROI index in Economics. The Return on Investment (ROI) is a performance measure used to evaluate the efficiency or profitability of an investment. ROI tries to directly measure the amount of return on a particular investment, relative to the investment's cost. For an investment of return R and cost C we define the ROI index as:

$$\texttt{ROI} := \frac{R}{C} \tag{3.6}$$

As explained in Chapter (1), in [10] authors show how the control of this index significantly impacts how advertisers select their campaigns.

While in the case of budget constraints, the goal was to keep costs below a daily budget b, in the case of ROI constraint, we would like to be able to keep at each round the reward/cost ratio above a threshold $\lambda \geq 1$.

Optimal constrained solution. We say that the CostMAB sequential game defined by the Protocol (5) is subject to ROI constraints if the policy $\mathfrak{U}_\star$ considered optimal prescribes each round to play the arm $a_\star \in \mathcal{A}$ solution of the constrained optimization problem:

$$\operatorname*{argmax}_{a\in\mathcal{A}} \mu_a \tag{3.7}$$

$$\text{s.t. } \frac{\mu_a}{\nu_a} \geq \lambda \tag{3.8}$$

Note that, unlike the budget case, we consider the threshold λ fixed for all rounds.

Regret. Once defined the optimal constrained policy, we can define the regret to measure the performances in terms of cumulative reward.

Definition 3.1.4. *Let $\Psi := \{(P_a, Q_a) : a \in \mathcal{A}\}$ be an instance of the CostMAB problem with daily ROI requirement $\lambda \geq 1$. Given the policy $\mathfrak{U}$, we call regret of $\mathfrak{U}$ over the time horizon T the quantity:*

$$\mathcal{R}_{\mathfrak{U}}(T) = T\mu_\star - \sum_{t=1}^{T} \mathbb{E}\left[X_t\right] \tag{3.9}$$

Where $\mu_\star$ is the mean reward of the optimal arm, and the expectation is with respect to the probability measure induced by the interaction between policy $\mathfrak{U}$ and the environment.

The definition is substantially unchanged from the case with budget constraints. Again, what is important is to define a class of safe algorithms.

ROI safe policies. Also in the ROI constrained problem, the measure of safety is the expected number of constraint violations.

Definition 3.1.5 (Expected number of ROI constraint violations)**.** *Let Ψ be an instance ROI CostMAB problem, with threshold $\lambda > 1$. Fix a policy $\mathfrak{U}$ and a time horizon T. We define the expected number of constraint violations of $\mathfrak{U}$ over T rounds:*

$$\mathbb{E}\left[J_T\right] := \mathbb{E}\left[\#\left\{t \in [T] : \frac{\mu_{A_t}}{\nu_{A_t}} < \lambda\right\}\right] \tag{3.10}$$

where A_t is the action selected at round t by policy $\mathfrak{U}$, Where A_t is the action selected at round t by policy $\mathfrak{U}$, μ_a and ν_a are respectively the mean of the distribution P_a and Q_a, $\forall a \in \mathcal{A}$.

Thus specify the class of ROI *η-safe policy*

Definition 3.1.6 (ROI η-safe policy)**.** *Fix $\eta \in (0, 1)$. The policy $\mathfrak{U}$ is said to be η−safe if*

$$\mathbb{E}\left[J_T\right] \leq \eta T \tag{3.11}$$

Namely, with probability at least $1-\eta$ the policy $\mathfrak{U}$ does not violate Constraint (3.8).

3.2 Impossibility theorem and quasi-safe policies

In this section, we present the impossibility theorem proved in [5]. The theorem was formulated for the CostMAB budget case but is trivially extensible to the case with ROI constraints. In [5] the authors prove the impossibility of finding a policy that on each instance of the problem admits sublinear bounds on the total regret and, at the same time, is η-safe for some $\eta \in (0, 1)$.

Theorem 3.2.1 (Regret–safety trade-off)**.** *Fix $\epsilon \in (0, 0.5)$ and a time horizon $T \in \mathbb{N}$. There is no policy $\mathfrak{U}$ such that for any instance Ψ of CostMAB problem with daily budget constraints both the following conditions hold:*

- *$\mathfrak{U}$ is $(\frac{1}{2} - \epsilon)$-safe*
- *$\mathcal{R}_T(\mathfrak{U}, \Psi) \leq (\frac{1}{2} - \epsilon)T$*

Where the definition of η-safe policy is given in Definition (3.1.3).

This theorem imposes us to relax the conditions that define the class of safe algorithms. What we will do is to introduce a tolerance threshold $\varepsilon > 0$ in the violation of the constraints. We formalize this idea below.

Budget quasi-safe policies. Let us be given a budget constrained CostMAB problem. We fix a threshold of tolerance $\varepsilon > 0$.

Definition 3.2.1 (Expected number of intolerable budget constraint violations)**.** *Let Ψ be an instance budget CostMAB problem, with daily budget b. Let $\varepsilon > 0$ be a given tolerance threshold. Fix a policy $\mathfrak{U}$ and a time horizon T. We define the expected number of intolerable constraint violations of $\mathfrak{U}$ over T rounds:*

$$\mathbb{E}\left[J_T^{\varepsilon}\right] := \mathbb{E}\left[\#\left\{t \in [T] : \nu_{A_t} \geq b + \varepsilon\right\}\right] \tag{3.12}$$

Where A_t is the action selected at round t by policy $\mathfrak{U}$, and ν_a is the mean of the distribution Q_a, $\forall a \in \mathcal{A}$

Definition 3.2.2 (Budget η-quasi-safe policy)**.** *Fix $\eta \in (0, 1)$. The policy $\mathfrak{U}$ is said to be η-quasi-safe with respect the tolerance threshold $\varepsilon > 0$ if*

$$\mathbb{E}\left[J_T^{\varepsilon}\right] \leq \eta T \tag{3.13}$$

Namely, the expected number of intolerable violations of Constraint (3.2) is upper-bounded by ηT.

ROI quasi-safe policies. In a completely analogous way we give the definitions for the ROI constrained CostMAB case. We set a tolerance threshold $\varepsilon > 0$.

Definition 3.2.3 (ROI quasi-safe policy)**.** *Let Ψ be an instance ROI CostMAB problem, with ROI threshold λ. Let $\varepsilon > 0$ be a given tolerance threshold. Fix a policy $\mathfrak{U}$ and a time horizon T. We define the expected number of intolerable constraint violations of $\mathfrak{U}$ over T rounds:*

$$\mathbb{E}\left[J_T^{\varepsilon}\right] := \mathbb{E}\left[\#\left\{t \in [T] : \frac{\mu_{A_t}}{\nu_{A_t}} \leq \lambda - \varepsilon\right\}\right] \tag{3.14}$$

Where A_t is the action selected at round t by policy $\mathfrak{U}$, μ_a and ν_a are respectively the mean of the distribution P_a and Q_a, $\forall a \in \mathcal{A}$.

Fix $\eta \in (0, 1)$. The policy $\mathfrak{U}$ is said to be η-quasi-safe if

$$\mathbb{E}\left[J_T^{\varepsilon}\right] \leq \eta T \tag{3.15}$$

In the following of the dissertation, we will propose quasi-safe algorithms that admit sublinear bounds in the regret. It is important to note that the level η of quasi-safety will, in general, depend on the selected ε tolerance threshold. We will focus our analysis on the relationship between the level of safety η and the tolerance threshold ϵ. We will show that it is possible to achieve sublinear regret and sublinear expected number of intolerable violations with a small tolerance.

3.3 Cost Multi-Task Stochastic Semi-Bandit

We extend the Multi-Task Stochastic Semi-Bandit framework described in Protocol (4) by introducing feedback on the cost of arms. The formalization of the following concepts is similar to what we have done for the CostMAB case. We briefly describe the problem as a sequential game between the decision-maker and the environment. The game environment has $M \in \mathbb{N}$ tasks composed by $K \in \mathbb{N}$ arms. At each turn $t \in [T]$ the player must select a superarm $\mathbf{A}_t \in \mathcal{S}$. The component $A_t^{(m)} \in [K]$ of the superarm represents the arm selected by the player in the m^{th} task. The environment then returns a feedback $\left(X_t^{(m)}, Y_t^{(m)}\right)$ for each task $m \in [M]$. $X_t^{(m)}$ is the partial reward of the m^{th} task in round t, $Y_t^{(m)}$ the partial cost. We call the vector $\mathbf{X}_t := \left[X_t^{(m)}\right]_{m=1}^{M}$ vector of partial rewards of round t, $\mathbf{Y}_t := \left[Y_t^{(m)}\right]_{m=1}^{M}$ vector of partial costs. The player then collects a reward $X_t := \sum_{m=1}^{M} X_t^{(m)}$ and suffers a cost $Y_t := \sum_{m=1}^{M} Y_t^{(m)}$. We assume that the distribution of the vector of partial rewards $\mathbf{X}_t$, conditional to the sequence of actions-feedbacks $\{\mathbf{A}_1, (\mathbf{X}_1, \mathbf{Y}_1), \ldots, \mathbf{A}_{t-1}, (\mathbf{X}_{t-1}, \mathbf{Y}_{t-1}), \mathbf{A}_t\}$, is $P_{\mathbf{A}_t}$. Analogously, the distribution of the vector of partial rewards $\mathbf{Y}_t$, conditional to the sequence of actions-feedbacks, is $Q_{\mathbf{A}_t}$.

Framework protocol 6 Cost Multi-Task Stochastic Semi-Bandit (CostMTSSB)

```
Input: $T$ time horizon, $K$ number arms, $M$ number of tasks, $\mathcal{S} = [K]^M$ set
       of superarms
for: $t = 1, ..., T$
  1. Select a superarm $\mathbf{A}_t \in \mathcal{S}$
  2. Observe the vector of partial rewards $\left[X_t^{(m)}\right]_{m \in [M]}$ sampled from $P_{\mathbf{A}_t}$.
  3. Observe the vector of partial costs $\left[Y_t^{(m)}\right]_{m \in [M]}$ sampled from $Q_{\mathbf{A}_t}$.
  4. Collect a reward $X_t := \sum_{m \in [M]} X_t^{(m)}$
  5. Suffer a cost $Y_t := \sum_{m \in [M]} Y_t^{(m)}$
```

Notation. *Fix $\boldsymbol{a} \in \mathcal{S}$, we define the following quantities:*

- $\boldsymbol{\mu}_{\boldsymbol{a}}$*, vector mean of the distribution* $P_{\boldsymbol{a}}$*.*
- $\mu_{\boldsymbol{a}}^{(m)}$*,* m^{th} *component of vector* $\boldsymbol{\mu}_{\boldsymbol{a}}$*.*
- $\boldsymbol{\nu}_{\boldsymbol{a}}$*, vector mean of the distribution* $Q_{\boldsymbol{a}}$*.*
- $\nu_{\boldsymbol{a}}^{(m)}$*,* m^{th} *component of vector* $\boldsymbol{\nu}_{\boldsymbol{a}}$*.*

3.3.1 Safe learning: budget constrained CostMTSSB

As we did for the CostMAB framework, we define an optimal solution against which to measure the algorithms' performance. Then, we define the class of *quasi-safe* algorithms to account for budget constraints.

Optimal constrained solution. Let $b \in (0, M]$ be the daily budget imposed by the problem. We say that the sequential game defined by Protocol (6) is subject to budget constraints if the policy $\mathfrak{U}_\star$ considered optimal prescribes at each round $t \in T$ the superarm $\mathbf{a}_\star \in \mathcal{A}$ solution of the optimization problem:

$$\operatorname*{argmax}_{\mathbf{a}\in\mathcal{S}} \sum_{m=1}^{M} \mu_{\mathbf{a}}^{(m)} \tag{3.16}$$

$$\text{s.t.} \sum_{m=1}^{M} \nu_{\mathbf{a}}^{(m)} \leq b \tag{3.17}$$

We observe that in the MTSSB case, the daily budget constraint to which the optimal solution is subject is imposed on the overall reward of the M tasks. This implies a combinatorial problem even when the tasks are independent of each other: it is possible that in the optimal solution, there are components with very high partial reward and cost, which are balanced by components with meager partial reward and cost.

Regret. Once defined the optimal constrained policy, we can define the regret.

Definition 3.3.1. *Let $\Psi := \{(P_{\boldsymbol{a}}, Q_{\boldsymbol{a}}) : \boldsymbol{a} \in \mathcal{S}\}$ be an instance of the budget constrained CostMTSSB problem. Given the policy $\mathfrak{U}$, we call regret of $\mathfrak{U}$ over the time horizon T the quantity:*

$$\mathcal{R}_{\mathfrak{U}}(T) = T \sum_{m=1}^{M} \mu_{\star}^{(m)} - \sum_{t=1}^{T} \sum_{m=1}^{M} \mathbb{E}\left[X_t^{(m)}\right] \tag{3.18}$$

Where $\boldsymbol{\mu}_{\star}$ is the mean vector of rewards relative to the optimal superarm, and the expectation is with respect to the probability measure induced by the interaction between policy $\mathfrak{U}$ and the environment.

The regret evaluation coincides with the definition in the context MTSSB without cost. Therefore, to find good policies in terms of cost constraint violations, we will need to define a class of safe policies from which we select our algorithms.

First, we observe that trade-off Theorem (3.2.1) also holds in the CostMTSSB case. Indeed, if by contradiction there exists a policy that admits sublinear regret and expected number of sublinear constraint violations, this should also hold for instances with M=1. However, CostMTSSB problems with M=1 coincide with CostMAB problems. This would contradict Theorem (3.2.1). It follows that it only makes sense to look for algorithms in the quasi-safe class.

Budget quasi-safe policies. Consider a budget-constrained CostMTSSB problem and fix a threshold $\varepsilon > 0$.

Definition 3.3.2 (Expected number of intolerable budget constraint violations)**.** *Let Ψ be an instance budget CostMTSSB problem, with daily budget b. Let $\varepsilon > 0$ be a given tolerance threshold. Fix a policy $\mathfrak{U}$ and a time horizon T. We define the expected number of intolerable constraint violations of $\mathfrak{U}$ over T rounds:*

$$\mathbb{E}\left[J_T^{\varepsilon}\right] := \mathbb{E}\left[\#\left\{t \in [T] : \sum_{m=1}^{M} \nu_{\boldsymbol{A}_t}^{(m)} \geq b + \varepsilon\right\}\right] \tag{3.19}$$

Where $\boldsymbol{A}_t$ is the superarm selected at round t by policy $\mathfrak{U}$.

Definition 3.3.3 (Budget quasi-safe policy)**.** *Fix $\eta \in (0,1)$. The policy $\mathfrak{U}$ is said to be η-quasi-safe with respect to the tolerance threshold $\varepsilon > 0$ if*

$$\mathbb{E}\left[J_T^{\varepsilon}\right] \leq \eta T \tag{3.20}$$

3.3.2 Safe learning: ROI constrained CostMTSSB

Finally, we formalize the player's goal in CostMTSSB when she wants to deal with ROI constraints. As we have done for the previous cases, we define an optimal constrained solution and a class of quasi-safe algorithms that does not violate the constraints intolerably too many times in expectation.

Optimal constrained solution. Let $\lambda \geq 1$. We say that the sequential game defined by Protocol (6) is subject to ROI constraints if the policy $\mathfrak{U}_\star$ considered optimal prescribes at each round $t \in [T]$ the superarm $\mathbf{a}_\star \in \mathcal{A}$ solution of the optimization problem:

$$\underset{\mathbf{a} \in \mathcal{S}}{\operatorname{argmax}} \sum_{m=1}^{m} \mu_{\mathbf{a}} \tag{3.21}$$

$$\text{s.t. } \frac{\sum_{m=1}^{M} \mu_{\mathbf{a}}^{(m)}}{\sum_{m=1}^{M} \nu_{\mathbf{a}}^{(m)}} \geq \lambda \ \forall t \in [T] \tag{3.22}$$

ROI quasi-safe policies. Consider a CostMTSSB problem with daily ROI requirement $\lambda \geq 1$ and fix a threshold $\epsilon > 0$.

Definition 3.3.4 (Expected number of intolerable budget constraint violations)**.** *Let Ψ be an instance budget CostMTSSB problem, with ROI threshold requirement $\lambda \geq 1$. Let $\varepsilon > 0$ be a given tolerance threshold. Fix a policy $\mathfrak{U}$ and a time horizon T. We define the expected number of intolerable* *constraint violations of* $\mathfrak{U}$ *over* T *rounds:*

$$\mathbb{E}\left[J_T^{\varepsilon}\right] := \mathbb{E}\left[\#\left\{t \in [T] : \frac{\sum_{m=1}^{M} \mu_{\boldsymbol{A}_t}^{(m)}}{\sum_{m=1}^{M} \nu_{\boldsymbol{A}_t}^{(m)}} \leq \lambda - \varepsilon\right\}\right] \tag{3.23}$$

Where $\boldsymbol{A}_t$ *is the superarm selected at round* t *by policy* $\mathfrak{U}$*.*

Definition 3.3.5 (ROI quasi-safe policy)**.** *Fix* $\eta \in (0, 1)$*. The policy* $\mathfrak{U}$ *is said to be* η*-quasi-safe with respect to the ROI constraint with respect to the tolerance threshold* $\varepsilon > 0$ *if*

$$\mathbb{E}\left[J_T^{\varepsilon}\right] \leq \eta T \tag{3.24}$$

3.4 Bid optimization modeling

After introducing the CostMAB and CostMTSSB frameworks, we show how they can model the bid optimization problem in advertising.

We divide two cases. The first, simpler, is the case of a single campaign that we model with the CostMAB framework. The second is the case of multiple sub-campaigns, which we model with the CostMTSSB framework.

3.4.1 Single campaign

The bid advertising problem with a single campaign can be modeled as follows. The advertiser has a set $\mathcal{I} := \{i_1, \ldots, i_N\}$ of possible advertisements that differ in several parameters such as placement on the page, format (images, text, video) or associated keywords. In each fixed period of time, hereinafter *round*, the advertiser participates in an auction. Let us assume finite the set $\mathcal{B} := \{\beta_1, ..., \beta_B\}$ of possible bids that the player can place. In each round, the advertiser chooses one among the pairs $(i, \beta) \in \mathcal{I} \times \mathcal{B}$. Each pair $(i, \beta) \in \mathcal{I} \times \mathcal{B}$ is associated with a stochastic cost that depends on the number of clicks the advertisement receives in the period until the next round. Each ad generates a return for the advertiser, which is also stochastic. We assume that the costs and returns of each pair have a fixed distribution independent of the round considered.

The advertiser's goal is to maximize the expected cumulative return over T rounds.

The problem can be framed in the CostMAB framework with set of arms $\mathcal{A} := \mathcal{I} \times \mathcal{B}$. The possible $K := N \times B$ pairs (i, β) represent the arms from which the player can choose.

The use of the CostMAB framework allows us to model the case where the advertiser has daily business constraints, such as:

- Budget Constraints. The advertiser has a daily budget b. The goal is to adopt a policy that is able to maintain the cumulative regret below a bound sublinear in T while picking arms that exceed in expectation the daily budget b only a number of times sublinear in T.
- ROI Constraints. The advertiser has a minimum threshold $\lambda \geq 1$ of ROI. The goal is to adopt a policy that is able to maintain the cumulative regret below a bound sublinear in T while picking arms that in expectation is below the daily ROI requirement λ only a number of times sublinear in T.

In the CostMAB model, algorithms that satisfy these requirements are modeled as *budget safe* and *ROI safe* algorithms. Since Theorem (3.2.1) shows the impossibility of constructing such algorithms, we relax the demands by granting that the constraints are satisfied unless there is a tolerance $\varepsilon > 0$ on the size of the violation. This relaxation coincides with looking for algorithms belonging to quasi-safe classes.

3.4.2 Multiple sub-campaigns

Let us model the advertiser's case to compose a campaign formed by M sub-campaigns. The sub-campaigns may differ from each other, for instance, by platform (social network, search engine) or format (video, image, text). Each of the M campaigns has a set $\mathcal{I} := \{i, \dots, i_N\}$ of possible advertisements. In each round, the advertiser participates in M auctions. For each of the M auctions, we assume the set $\mathcal{B} := \{\beta_1, ..., \beta_B\}$ of possible bets that the player can place to be finite. Each round the player chooses a pair $(i, \beta) \in \mathcal{I} \times \mathcal{B}$ for each sub-campaign $m \in [M]$.

Each choice corresponds to a stochastic cost that depends on the number of clicks generated by the ad and a stochastic reward. The advertiser is able to observe costs and rewards of each of the M choices. Then, she incurs a cost equal to the sum of the costs and a reward equal to the sum of the rewards. We assume costs and rewards to have unknown but fixed distributions independent across sub-campaigns.

The problem can be framed in the CostMTSSB context with set of superarms $\mathcal{S} := (\mathcal{I} \times \mathcal{B})^M$.

The goal is to maximize the expected cumulative reward over T rounds while satisfying daily business constraints. As in the single campaign case, we consider as possible constraints:

- Budget constraints. The advertiser has a daily budget b. The goal is to adopt a policy that is able to maintain the cumulative regret below a bound sublinear in T while picking superarms that exceed in expectation the daily budget b only a number of times sublinear in T.

- ROI Constraints. The advertiser has a minimum threshold $\lambda \geq 1$ of ROI. The goal is to adopt a policy that is able to maintain the cumulative regret below a bound sublinear in T while picking superarms that in expectation is below the daily ROI requirement λ only a number of times sublinear in T.

As in the CostMAB case, we relax these requirements by granting that the constraints are satisfied up to a tolerance $\varepsilon > 0$ on the size of the violation. At a modeling level, this coincides with looking for algorithms belonging to the budget quasi-safe and ROI quasi-safe classes, respectively.

Chapter 4

CostMAB algorithms

In this chapter, we focus on the CostMAB framework introduced in the previous chapter. We propose two algorithms: the first belongs to the budget quasi-safe class, the second to the ROI quasi-safe class. Both are inspired by the UCB1 algorithm presented in Chapter (2). Clearly, it is necessary to use particular expedients to take into account the cost feedback and restrict the policies to the quasi-safe class. For both policies, the analysis is developed as follows. First, we present the algorithm and the main ideas used to account for the constraints. Then, we show that the algorithm under analysis belongs to the class of quasi-safe policies, emphasizing the relationship between the expected number of intolerable violations of the constraints and the tolerability threshold. Finally, we show that the proposed algorithm admits a bound for the regret that is sublinear in the number of rounds T.

4.1 BudgetLUCB algorithm

In this section we propose an algorithm for the CostMAB problem with daily budget constraints. The algorithm is based on the principle of Optimism in the Face of Uncertainty: at each round we elaborate an optimistic estimate on the expected reward and the expected cost of each arm. Such an estimate will be an upper confidence bound for the expected reward, a lower confidence bound for the expected cost. Therefore, we call our algorithm BudgetLUCB (Budget Lower/Upper Confidence Bounds).

4.1.1 Algorithm description

The algorithm uses at each round optimistic guests on the expected cost and the expected reward. These estimations are defined as:

$$\texttt{UCB}_a(t-1) := \begin{cases} +\infty & \text{if } N_a(t-1) = 0 \\ \hat{\mu}_a(t-1) + \sqrt{\frac{2\log(T)}{N_a(t-1)}} & \text{otherwise} \end{cases} \tag{4.1}$$

$$\texttt{LCB}_a(t-1) := \begin{cases} -\infty & \text{if } N_a(t-1) = 0 \\ \max\left\{\hat{\nu}_a(t-1) - \sqrt{\frac{2\log(T)}{N_a(t-1)}}, 0\right\} & \text{otherwise} \end{cases} \tag{4.2}$$

Where $N_a(t)$ is the number of times the arm a has been sampled up to round t. $\hat{\mu}_a(t) = \frac{1}{N_a(t)} \sum_{s=1}^{t} X_s \mathbb{1}\{A_s = a\}$ is the average of rewards of arm a collected up to round t. $\hat{\nu}_a(t) = \frac{1}{N_a(t)} \sum_{s=1}^{t} Y_s \mathbb{1}\{A_s = a\}$ is the average of costs of arm a, suffered up to round t.

At each round the algorithm computes UCB and LCB for each arm $a \in \mathcal{A}$ and solves the optimistic optimization problem:

$$\underset{a \in \mathcal{A}}{\operatorname{argmax}} \, \texttt{UCB}_a(t-1) \tag{4.3}$$

$$\text{s.t } \texttt{LCB}_a(t-1) \leq b \tag{4.4}$$

We assume the existence of a *null arm*, that has zero cost and reward almost surely. So, we can ensure the set of feasible arms to be non-empty.

We summarize here the algorithm.

Algorithm 4 BudgetLUCB

```
Input: $\mathcal{A} = [K]$ arms set, $T \geq K$ time horizon, $b$ daily budget
for $t$= 1,...,$T$ do
    Choose action $A_t$ solution of the constrained optimization problem:
        $\underset{a \in \mathcal{A}}{\operatorname{argmax}} \, \texttt{UCB}_a(t-1)$    (4.5)
            s.t $\texttt{LCB}_a(t-1) \leq b$    (4.6)
    Observe reward $X_t$ and cost $Y_t$ and update confidence bounds.
```

The idea is to be optimistic both on how high the reward of unexplored arms could be, and on how low the cost these arms could be.

It is worth summarizing the notation used in the following.

Notation.
Fix $a \in \mathcal{A}$, $t \in [T]$, $j \in [T]$.

- μ_a, *mean of distribution* P_a
- ν_a, *mean of distribution* Q_a
- $N_a(t)$, *number of times arm a has been sampled up to round t*
- $\hat{\mu}_a(t) := \sum_{s=1}^{t} X_t \mathbb{1}\{A_t = a\}$, *sample mean of rewards sampled up to round t from arm a*
- $\hat{\nu}_a(t) := \sum_{s=1}^{t} Y_t \mathbb{1}\{A_t = a\}$, *sample mean of costs sampled up to round t from arm a*
- $\hat{\mu}_{a,j}$, *average of the first j values sampled independently from distribution* P_a
- $\hat{\nu}_{a,j}$, *average of the first j values sampled independently from distribution* Q_a

4.1.2 Safety analysis

We show that Algorithm (4) belongs to the class of budget quasi-safe policies.

The crucial point in the analysis of the proposed algorithms will be the concept of Clean Event. Intuitively, the Clean Event is an event that happens with high probability and under which the algorithm has access to good estimations of the expected arm feedback[1]. We formalize this concept below.

Definition 4.1.1 (Clean Event)**.** *Let* $\Psi := \{(P_a, Q_a) : a \in \mathcal{A}\}$ *be an instance of the CostMAB problem. We define Clean Event ($\mathcal{E}$) the event:*

$$\mathcal{E} := \left\{ \forall a \in \mathcal{A}\ \forall t \in [T],\ |\mu_a - \hat{\mu}_a(t)| < \sqrt{\frac{2\log(T)}{N_a(t)}} \wedge |\nu_a - \hat{\nu}_a(t)| < \sqrt{\frac{2\log(T)}{N_a(t)}} \right\}$$

It is important to emphasize the meaning of the Clean Event. The goal of the decision-maker is to choose the arm with the maximum expected reward that satisfies the Constraint (3.1.1). Under the Clean Event, the player knows an interval around value of the expected reward and the value

[1] We say that and event A happens (almost surely) *under* the event B, if $P(A|B) = 1$.

of the expected cost. This means that, at each round $t \in [T]$, the player knows $\forall a \in \mathcal{A}$ that:

$$\mu_a \in \left[\hat{\mu}_a - \sqrt{\frac{2\log(T)}{N_a(t)}}, \hat{\mu}_a + \sqrt{\frac{2\log(T)}{N_a(t)}}\right] \tag{4.7}$$

$$\nu_a \in \left[\hat{\nu}_a - \sqrt{\frac{2\log(T)}{N_a(t)}}, \hat{\nu}_a + \sqrt{\frac{2\log(T)}{N_a(t)}}\right] \tag{4.8}$$

Note that the intervals shrink as the number of times the arm is played $N_a(t)$ increases: intuitively, this allows us to learn which arms respect the constraints in a tolerable way. We will show that we can learn this performing number of sampling $N_a(t)$ sublinear in T.

In particular the following Lemma holds.

Lemma 4.1.1. *Let Ψ be an instance of the CostMAB problem with daily budget b. Let $\{A_t\}_{t=1}^{T}$ be the arms selected by Algorithm (4) when applied to instance Ψ. Fix a tolerance threshold $\varepsilon > 0$. For every $t \in [T]$, the event*

$$\{\nu_{A_t} > b + \varepsilon\} \tag{4.9}$$

is impossible, under the Clean Event $\mathcal{E}$, if $N_{A_t}(t-1) > \frac{8\log(T)}{\varepsilon^2}$.

Proof. First, notice that

$$N_{A_t}(t-1) > \frac{8\log(T)}{\varepsilon^2} \iff \varepsilon > 2\sqrt{\frac{2\log(T)}{N_{A_t}(t-1)}} \tag{4.10}$$

Second, according to Algorithm (4) A_t can be chosen only if

$$\texttt{LCB}_{A_t}(t-1) \le b \tag{4.11}$$

Moreover, under $\mathcal{E}$, must hold:

$$\texttt{LCB}_{A_t}(t-1) = \hat{\nu}_{A_t}(t-1) - \sqrt{\frac{2\log(T)}{N_{A_t}(t-1)}} \ge \nu_{A_t} - 2\sqrt{\frac{2\log(T)}{N_{A_t}(t-1)}} \tag{4.12}$$

Suppose by contradiction $\nu_{A_t} > b + \varepsilon$, then

$$\begin{align} \nu_{A_t} &> b + \varepsilon \tag{4.13}\\ &> b + 2\sqrt{\frac{2\log(T)}{N_{A_t}(t-1)}} \tag{4.14}\\ &> \texttt{LCB}_{A_t}(t-1) + 2\sqrt{\frac{2\log(T)}{N_{A_t}(t-1)}} \tag{4.15}\\ &> \nu_{A_t} \tag{4.16} \end{align}$$

where in Line (4.14) we use Inequality (4.10), in Line (4.15) we use Inequality (4.11), in Line (4.16) we use Inequality (4.12). Hence, we reach a contradiction. □

This Lemma assures us that, under the Clean Event, if Algorithm (4) selects an arm that has already been played at least $\frac{8\log(T)}{\varepsilon^2}$ times, no ε-intolerable violation of the budget constraint can be committed. So if the Clean Event holds, we can commit at most a logarithmic number of violations per arm.

We now show that the Clean Event happens with high probability.

Lemma 4.1.2 (Probability of Clean Event)**.** *Let $\Psi \in \Xi_{BS(0,1)}$ be an instance of the CostMAB problem. Then, the probability of the event $\mathcal{E}$ in Definition (4.1.1) is at least $1 - \frac{4}{T^2}$.*

Proof. Observe that:

$$\begin{aligned}\mathcal{E} := &\left\{\forall a \in \mathcal{A}\, \forall t \in [T],\ |\mu_a - \hat{\mu}_a(t)| < \sqrt{\frac{2\log(T)}{N_a(t)}} \wedge |\nu_a - \hat{\nu}_a(t)| < \sqrt{\frac{2\log(T)}{N_a(t)}}\right\}\\ &\supset \left\{\forall i \in [K]\, \forall j \in [T],\ |\mu_i - \hat{\mu}_{i,j}| < \sqrt{\frac{2\log(T)}{j}} \wedge |\nu_i - \hat{\nu}_{i,j}| < \sqrt{\frac{2\log(T)}{j}}\right\}\\ &=: \tilde{\mathcal{E}} \end{aligned} \tag{4.17}$$

Moreover, $\forall i \in [K]$, $\forall j \in [T]$:

$$\begin{aligned}&\mathbb{P}\left(|\mu_i - \hat{\mu}_{i,j}| > \sqrt{\frac{2\log(T)}{j}} \vee |\nu_i - \hat{\nu}_{i,j}| > \sqrt{\frac{2\log(T)}{j}}\right)\\ &\leq \mathbb{P}\left(|\mu_i - \hat{\mu}_{i,j}| > \sqrt{\frac{2\log(T)}{j}}\right) + \mathbb{P}\left(|\nu_i - \hat{\nu}_{i,j}| > \sqrt{\frac{2\log(T)}{j}}\right)\\ &\leq 4T^{-4} \end{aligned} \tag{4.18}$$

where we used Union Bound and Chernoff-Hoeffding Bound (2.1.1).

Thus, assuming $T \geq K$, we can conclude:

$$\mathbb{P}(\mathcal{E}) \geq \mathbb{P}\left(\tilde{\mathcal{E}}\right) \tag{4.19}$$

$$\geq 1 - 4T^{-4}KT \geq 1 - 4T^{-2}. \tag{4.20}$$

Where, in Line (4.19) we used Equation (4.17) and in Line (4.20) we used Inequality (4.18) and the Union Bound. □

Exploiting the previous two Lemmas, we can show that Algorithm (4) belongs to the budget quasi-safe class.

Theorem 4.1.1 (BudgetLUCB budget quasi-safety)**.** *Let $\Psi \in \Xi_{BS(0,1)}$ be an instance of the CostMAB problem with daily budget b. Fix a tolerance threshold $\varepsilon > 0$. Let $\{A_t\}_{t=1}^T$ be the arms selected by Algorithm (4) when applied to the Ψ. Then the expected number of intolerable constraint violations is bounded as:*

$$\mathbb{E}\left[J_T\right] := \mathbb{E}\left[\sum_{t=1}^{T} \mathbb{1}\left\{\nu_{A_t} > b + \varepsilon\right\}\right] \leq \mathcal{O}\left(\frac{K\log(T)}{\varepsilon^2}\right) \tag{4.21}$$

Proof. The proof is based on the two previous Lemmas.

$$\begin{aligned}
\mathbb{E}\left[J_T\right] :=& \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left\{\nu_{A_t} > b + \varepsilon\right\}\right] & (4.22)\\
=& \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left\{\nu_{A_t} > b + \varepsilon\right\} | \mathcal{E}\right] \mathbb{P}\left(\mathcal{E}\right) + \\
&+ \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left\{\nu_{A_t} > b + \varepsilon\right\} | \mathcal{E}^c\right] \mathbb{P}\left(\mathcal{E}^c\right) & (4.23)\\
\leq& \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left\{\nu_{A_t} > b + \varepsilon\right\} | \mathcal{E}\right] + 4TT^{-2} & (4.24)\\
\leq& \frac{8\log(T)}{\varepsilon^2} K + \frac{4}{T} = \mathcal{O}\left(\frac{K\log(T)}{\varepsilon^2}\right) & (4.25)
\end{aligned}$$

where in Line (4.23) we used the Law of Total Expectation, in Line (4.24) we used Lemma (4.1.2), in Line (4.25) we used Lemma (4.1.1) to state that each selected arm can commit an ε-intolerable violation at most $\frac{8\log(T)}{\varepsilon^2}$ times. □

Remark 4. We underline two aspects of this result.

1. Fix a safety level $\eta \in (0, 1)$. From Theorem (4.1.1) is trivial to find a threshold ε such that Algorithm (4) is η-quasi-safe with respect to budget constraint with a tolerance ε. It suffices to select $\varepsilon \geq \sqrt{\frac{8K\log(T)}{T\eta}}$. Observe that for small values of η, the tolerability threshold ε can even become greater than 1. Such an high tolerability threshold does not make sense, since $b \in [0, 1]$. We have to manage the trade-off between the level of safety η and the threshold ε.
2. In particular, observe that to ensure a sublinear expected number of intolerable constraint violations, we have only to require $\epsilon = \mathcal{O}\left(\frac{1}{T^\alpha}\right)$

with $\alpha < 0.5$. Thus the larger is T, the smaller the tolerance level can be to maintain sublinear number expected of intolerable constraint violation.

4.1.3 Regret analysis

In this section, we want to show that Algorithm (4) admits a instance independent bound on the total regret that is sublinear in the time horizon T. The idea of the proof is to show that the optimal arm is never considered infeasible in the Optimization Problem (4.4). Ensured this, we can trace our proof back to arguments similar to the ones used in [19] to prove UCB1 regret bounds.

We first give the following definitions.

Definition 4.1.2 (Budget feasible arm)**.** *Fix an instance Ψ of the CostMAB problem with daily budget b. An arm $a \in \mathcal{A}$ is said to be feasible at round t if*

$$\nu_a \leq b. \tag{4.26}$$

We call $\mathcal{A}_f := \{a \in \mathcal{A} : \nu_a \leq b \; \forall t \in [T]\}$ the set of feasible arms.

Remark 5. Trivially, an optimal arm $a_\star$ solution of the Optimization Problem (3.1) belongs to $\mathcal{A}_f$.

Definition 4.1.3 (Budget empirically feasible arm)**.** *Fix an instance Ψ of the CostMAB problem with daily budget b. Fix a tolerance threshold $\varepsilon > 0$. Algorithm (4) considers an arm $a \in \mathcal{A}$ empirically feasible at round $t \in [T]$ if*

$$\texttt{LCB}_a(t-1) \leq b \tag{4.27}$$

To find a bound for the regret, we exploit again the idea of Clean Event of Definition (4.1.1).

Lemma 4.1.3. *Fix $t \in [T]$. Under the Clean Event, for any arm $a \in \mathcal{A}_f$, it holds:*

$$\texttt{LCB}_a(t-1) \leq b \tag{4.28}$$

Proof. Under the Clean Event,

$$\texttt{LCB}_a(t-1) = \hat{\nu}_a(t-1) - \sqrt{\frac{2\log(T)}{N_a(t-1)}} \leq \nu_a \tag{4.29}$$

Therefore,

$$\texttt{LCB}_a(t-1) \leq b \tag{4.30}$$

using the fact that arm a is feasible. □

This Lemma states that, under the Clean Event, every feasible arm is considered empirically feasible by Algorithm (4). Follows,

Corollary 4.1.1. *Consider an instance Ψ of the CostMAB algorithm with daily budget b. Let $\{A_t\}_{t=1}^{T}$ be the sequence of arm selected by the algorithm BudgetLUCB when applied to Ψ. The event*

$$\{\mathtt{UCB}_{A_t}(t-1) < \mathtt{UCB}_{\star}(t-1)\} \tag{4.31}$$

is impossible under the Clean Event.

Proof. Because of the definition of Algorithm (4), at round $t \in [T]$ arm A_t is selected only if

$$\begin{aligned} &\mathtt{UCB}_{A_t}(t-1) \geq \mathtt{UCB}_a(t-1) \\ &\text{for all } a \in \mathcal{A} \text{ such that } \mathtt{LCB}_a(t-1) \leq b \end{aligned}$$

Applying Lemma (4.1.3) and Remark (5), follows the thesis. □

In other words, under the Clean Event, a suboptimal arm is never preferred to the optimal one because of the budget constraint: if a suboptimal arm is chosen is because we still have a large uncertainty about its mean reward.

This fact is extremely important: in UCB-like algorithm, when the condition of Corollary (4.1.1) holds, we are able to bound the optimality gaps. Furthermore, these bounds decrease with the number of times that an arm is sampled. We formalize this fact in the following Lemma, that will be the pillar in finding a bound for the regret.

Lemma 4.1.4 (Optimality gap bound)**.** *Consider an instance Ψ of the CostMAB algorithm with daily budget b. Let $\{A_t\}_{t=1}^{T}$ be the sequence of arm selected by the algorithm BudgetLUCB applied to Ψ. For every arm $a \in \mathcal{A}$, let us indicate with $\Delta_a := \mu_\star - \mu_a$ its optimality gap. Then, the for any $a \in \mathcal{A}$, the event*

$$\mathcal{N}_a := \left\{ \Delta_a > \sqrt{\frac{8\log(T)}{N_a(T)}} \right\} \tag{4.32}$$

is impossible under the Clean Event.

Proof. For any $a \in \mathcal{A}$, for any $t \in [T]$ consider the event:

$$\mathcal{N}_a(t) := \left\{ A_t = a, \Delta_{A_t} > 2\sqrt{\frac{2\log(T)}{N_{A_t}(t-1)}} \right\} \tag{4.33}$$

$$\subset \left\{ A_t = a, \mu_\star > \hat{\mu}_{A_t} + \sqrt{\frac{2\log(T)}{N_{A_t}(t-1)}} \right\} \tag{4.34}$$

$$\equiv \{A_t = a, \mu_\star > \texttt{UCB}_{A_t}(t-1)\} \tag{4.35}$$

$$\subset \{\mu_\star > \texttt{UCB}_\star(t-1)\} \tag{4.36}$$

where in Line (4.34) we used the fact that we assume the Clean Event to be true, in Line (4.35) we use the definition of `UCB` and in Line (4.36) we use Corollary (4.1.1). Thus $\mathcal{N}_a(t)$ is impossible under $\mathcal{E}$, since the event in Line (4.36) is impossible under $\mathcal{E}$.

The thesis follows from the observation that

$$\mathcal{N}_a = \left\{ A_{\tau_a} = a, \Delta_a > \sqrt{\frac{8\log(T)}{N_a(T)}} \right\} \subset \bigcup_{t=1}^{T} \mathcal{N}_a(t) \tag{4.37}$$

where we denote with $\tau_a := \max\{t \in [T] : A_t = a\}$, so that $N_a(\tau_a) = N_a(T)$. Note that $\forall a \in \mathcal{A}$, $\tau_a \geq 1$. Finally, the event in Line (4.37) is impossible being countable union of impossible events, thus the thesis. □

Finally, we are able to prove that Algorithm (4) admits a sublinear regret bound.

Theorem 4.1.2 (Instance independent regret bound for BudgetLUCB)**.** *For any instance $\Psi \in \Xi_{BS(0,1)}$ of the CostMAB problem with daily budget b, the regret of Algorithm (4) on Ψ is bounded as:*

$$\mathcal{R}_T \leq \mathcal{O}\left(\sqrt{KT\log(T)}\right) \tag{4.38}$$

Where K is the cardinality of the arms' set $\mathcal{A}$ and $T \in \mathbb{N}$ the time horizon.

Proof. To prove the statement we will exploit the regret decomposition presented in Lemma (2.1.2).

$$
\begin{aligned}
\mathcal{R}_T =& \sum_{a=1}^{K} \mathbb{E}\left[N_a(T)\Delta_a\right] && (4.39)\\
=& \sum_{a=1}^{K} \mathbb{E}\left[N_a(T)\Delta_a|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) + \sum_{a=1}^{K} \mathbb{E}\left[N_a(T)\Delta_a|\mathcal{E}^c\right]\mathbb{P}\left(\mathcal{E}^c\right) && (4.40)\\
\leq& \sum_{a=1}^{K} \mathbb{E}\left[\sqrt{8\log(T)N_a(T)}\Big|\mathcal{E}\right] + \mathcal{O}\left(\frac{1}{T}\right) && (4.41)\\
\leq& \mathbb{E}\left[\sqrt{8\log(T)K\sum_{a=1}^{K} N_a(T)}\Bigg|\mathcal{E}\right] + \mathcal{O}\left(\frac{1}{T}\right) && (4.42)\\
\leq& \mathcal{O}\left(\sqrt{KT\log(T)}\right) && (4.43)
\end{aligned}
$$

Where in Line (4.40) we use the law of total expectation. In Line (4.41) we use the fact that $\mathbb{P}\left(\mathcal{E}\right) \leq 1$, $\mathbb{P}\left(\mathcal{E}^c\right) \leq 4T^{-2}$ for Lemma (4.1.2) and $K \leq T$. In Line (4.42) we use the linearity of the operator $\mathbb{E}\left[\cdot\right]$ and Jensen inequality on the concave map $x \mapsto \sqrt{x}$. Finally, in Line (4.43) we use the fact that $\sum_{a=1}^{K} N_a(T) = T$. □

4.2 ROI-LUCB algorithm

In this section, we propose an algorithm for the CostMAB problem with daily ROI constraints. Again, the algorithm is inspired by the idea of Optimism in the Face of Uncertainty. At each round we elaborate an optimistic guess on both expected reward and expected costs and, based on these guess, we elaborate a guess on the ROI of each arm. Is important to underline that in this case we don't assume to base our choices directly on noisy observations of the ROI: in the CostMAB setting we have access only to noisy observations of μ_a and ν_a for each sampled arm $a \in \mathcal{A}$. The fact that $ROI_a := \frac{\mu_a}{\nu_a}$ is non linear in (μ_a, ν_a) will lead to the need of a more sophisticated algorithm to ensure quasi-safety.

4.2.1 Algorithm description

The algorithm computes at each round the optimistic estimates of expected reward and expected cost for every arm, respectively `UCB` and `LCB` already defined in Equations (4.1, 4.2). In the case of ROI constraint, we would like to solve an optimistic version of the Optimization Problem (3.7). Issues arise when the estimates of expected reward and expected cost are close to zero. Suppose we are able to estimate an interval with high confidence level for μ_{A_t} and ν_{A_t}: this in general would not help us in estimating μ_{A_t}/ν_{A_t} for small costs. If, for example, we are able to assure with high probability that, for some $\alpha, \beta > 0$, $\mu_{A_t} \in [\alpha, \alpha+\varepsilon]$ and $\nu_{A_t} \in [\beta, \beta+\varepsilon]$, the only thing we can assure about μ_{A_t}/ν_{A_t} is that it belongs to $[\alpha/(\beta+\varepsilon), (\alpha+\varepsilon)/\beta]$. To require a confidence interval on the μ_{A_t}/ν_{A_t} of length $\mathcal{O}(\varepsilon)$, we need ε to have a equal or lower order of magnitude of α and β[2]. The problem is that for very small values of μ_{A_t} and ν_{A_t}, achieving such a precise confidence interval requires a large number of sampling. The algorithm we propose circumvents this problem by requiring that the selected arms have a sufficiently large estimate on the expected reward. We formalize this idea below.

At each round $t \in [T]$ the algorithm chooses the arm solution of the optimistic constrained problem:

$$\underset{a \in \mathcal{A}}{\text{argmax}} \quad \texttt{UCB}_a(t-1) \tag{4.44}$$

$$s.t. \quad \frac{\texttt{UCB}_a(t-1)}{\texttt{LCB}_a(t-1)} \geq \lambda \tag{4.45}$$

$$\texttt{UCB}_a(t-1) > T^{-1/3} \tag{4.46}$$

Algorithm 5 ROI-LUCB

```
Input: $T$ time horizon, $\mathcal{A} = \{1, ..., K\}$ arms set, $\lambda$ minimum ROI requirement
for $t$=1,...,$T$ do
    Choose arm $A_t$ solution of:
        $\underset{a \in \mathcal{A}}{\text{argmax}} \quad \texttt{UCB}_a(t-1)$
        $s.t. \quad \frac{\texttt{UCB}_a(t-1)}{\texttt{LCB}_a(t-1)} \geq \lambda$
        $\texttt{UCB}_a(t-1) > T^{-1/3}$
    Observe reward $X_t$ and cost $Y_t$ and update confidence bounds
```

Observe that the Constraint (4.46) ensures that the selected arm has a significant upper confidence bound on the ROI. The price of this constraint is that the algorithm may not asymptotically play the optimal arm if it has a low expected reward. As we will see in the regret analysis, the choice of the

[2] For instance suppose $\alpha = \beta = T^{-5}$ and $\varepsilon = T^{-4}$. The length of confidence intervals on μ_{A_t} and ν_{A_t} is $\mathcal{O}\left(T^{-4}\right)$, while the length of the confidence interval on μ_{A_t}/ν_{A_t} is $\mathcal{O}\left(T\right)$ that is too large to be meaningful.

threshold $T^{-1/3}$ allows us to ensure sublinear regret even in this eventuality. We assume the existence of a *null arm*, that has zero cost and reward and is always considered feasible.

4.2.2 Safety analysis

We show that Algorithm (5) belongs to the ROI quasi-safe class of policies. Like in the case of BudgetLUCB, the safety analysis relies on the concept of Clean Event.

First observe that Definition (4.1.1) of Clean Event and Lemma (4.1.2) are properties of the CostMAB framework and hold also with daily ROI requirements.

We want to show that, under the Clean Event, Algorithm (5) can commit no more than a sublinear number of intolerable violations per arm.

Lemma 4.2.1. *Let Ψ be an instance of the CostMAB problem with daily ROI requirement $\lambda \geq 1$. Let $\{A_t\}_{t=1}^{T}$ be the arms selected by Algorithm (5) when applied to instance Ψ. Fix a tolerance threshold $\varepsilon > 0$. There exist $h = h(T; \varepsilon, \lambda)$ such that, for every $t \in [T]$, the event*

$$\left\{ \frac{\mu_{A_t}}{\nu_{A_t}} < \lambda - \varepsilon \right\} \tag{4.47}$$

is impossible, under the Clean Event $\mathcal{E}$, if $N_{A_t}(t-1) > h$. In particular this holds for:

$$h(T; \epsilon, \lambda) = 8T^{2/3} \log(T) \left(\frac{\lambda(\lambda+1)}{\epsilon} \right)^2$$

Proof. Let's first remark that A_t is selected only if the following hold:

$$\frac{\texttt{UCB}_{A_t}(t-1)}{\texttt{LCB}_{A_t}(t-1)} > \lambda \tag{4.48}$$

$$\texttt{UCB}_{A_t}(t-1) > T^{-1/3} \tag{4.49}$$

We want to show that exists $h(T; \epsilon, \lambda)$ such that if $N_{A_t}(t-1) \geq h(T; \epsilon, \lambda)$ then (4.48) implies $\mu_{A_t}/\nu_{A_t} \geq \lambda - \epsilon$.
For sake of clearness in the computations, let us define the following variables:

$$\hat{N}_{A_t}(t-1) := T^{-2/3} N_{A_t}(t-1) \qquad \eta := \sqrt{\frac{2\log(T)}{\hat{N}_{A_t}(t-1)}}$$

From these definitions and under the $\mathcal{E}$ holds:

$$r_{A_t}(t-1) := \sqrt{\frac{2\log(T)}{N_{A_t}(t-1)}} = T^{-1/3}\eta \tag{4.50}$$

Moreover:

$$\mu_{A_t} > \texttt{UCB}_{A_t}(t-1) - 2r_{A_t}(t-1) \geq T^{-1/3}(1-2\eta) \tag{4.51}$$

where the first inequality comes from the assumption that $\mathcal{E}$ holds and the last from (4.50).
Hence:

$$r_{A_t}(t-1) \leq T^{-1/3}\eta = \frac{\eta}{1-2\eta}(1-2\eta)T^{-1/3} \leq \frac{\eta}{1-2\eta}\mu_{A_t} \tag{4.52}$$

using (4.50) and (4.51).
So we can find an upper-bound for $\texttt{UCB}_{A_t}(t-1)$ as follows:

$$\texttt{UCB}_{A_t}(t-1) \leq \mu_{A_t} + 2r_{A_t}(t-1) \leq \mu_{A_t}\left(1 + 2\frac{\eta}{1-2\eta}\right) \tag{4.53}$$

exploiting again that the event $\mathcal{E}$ holds in the first inequality and using (4.52) in the second. For what concerns the costs we can divide the analysis in two cases.
<u>Case 1:</u>

$$\nu_{A_t} \leq T^{-1/3}/\lambda \tag{4.54}$$

In this case:

$$\frac{\mu_{A_t}}{\nu_{A_t}} \geq \frac{T^{-1/3}(1-2\eta)}{T^{-1/3}/\lambda} = (1-2\eta)\lambda \geq \lambda - \varepsilon \tag{4.55}$$

where we used (4.51) and (4.54) in the first inequality and the last holds if

$$N_{A_t}(t-1) \geq \frac{8\lambda^2 T^{2/3}\log(T)}{\varepsilon^2} \tag{4.56}$$

<u>Case 2:</u>

$$\nu_{A_t} > T^{-1/3}/\lambda \tag{4.57}$$

In this case we have to lower-bound $\texttt{LCB}_{A_t}(t-1)$ to confirm the thesis.
Let's first notice that:

$$r_{A_t}(t-1) < T^{-1/3}\eta = \frac{T^{-1/3}}{\lambda}\lambda\eta \leq \nu_{A_t}\lambda\eta \tag{4.58}$$

using respectively (4.50) and (4.57).
It follows:

$$\texttt{LCB}_{A_t}(t-1) \geq \nu_{A_t} - 2r_{A_t}(t-1) \geq \nu_{A_t}(1-2\eta\lambda) \tag{4.59}$$

Finally:

$$\lambda \leq \frac{\texttt{UCB}_{A_t}(t-1)}{\texttt{LCB}_{A_t}(t-1)} \leq \frac{\mu_{A_t}}{\nu_{A_t}}\frac{1}{(1-2\eta)(1-2\eta\lambda)} \tag{4.60}$$

where the first inequality raises from (4.48) and the second from (4.53) and (4.59).
To conclude the proof we have to find $h(T;\epsilon;\lambda)$ such that $n_{A_t}(t-1) \leq h(T;\epsilon,\lambda)$ implies:

$$\lambda(1-2\eta)(1-2\eta\lambda) \geq \lambda - \epsilon \tag{4.61}$$

and such that the condition (4.56) holds.
It's easy to check that:

$$h(T;\epsilon,\lambda) = 8T^{2/3}\log(T)\left(\frac{\lambda(\lambda+1)}{\epsilon}\right)^2 \tag{4.62}$$

leads to that conclusion.
In fact in this case $N_{A_t}(t-1) \geq h(T;\epsilon,\lambda)$ implies:

$$\hat{N}_{A_t}(t-1) \geq 8\log(T)\left(\frac{\lambda(\lambda+1)}{\epsilon}\right)^2 \tag{4.63}$$

$$\implies \sqrt{\frac{2\log(T)}{\hat{N}_{A_t}(t-1)}} \leq \frac{\epsilon}{2\lambda(\lambda+1)} \tag{4.64}$$

$$\implies \eta \leq \frac{\epsilon}{2\lambda(\lambda+1)} \tag{4.65}$$

$$\implies 1-2\eta(1+\lambda) \geq \frac{\lambda-\epsilon}{\lambda} \tag{4.66}$$

$$\implies 1-2\eta-2\eta\lambda+4\eta^2\lambda \geq \frac{\lambda-\epsilon}{\lambda} \tag{4.67}$$

$$\implies \lambda(1-2\eta)(1-2\eta\lambda) \geq \lambda-\epsilon \tag{4.68}$$

Note that (4.56) is implied by $N_{A_t}(t-1) \geq h(T;\epsilon,\lambda)$, concluding the proof. □

This result leads us to conclude that each arm, under the $\mathcal{E}$, can commit at most $h(T;\epsilon;\lambda)$ ϵ-intolerable violations. From follows this argument the ROI quasi-safety property of Algorithm (5).

Theorem 4.2.1 (ROI-LUCB quasi-safety). *Let $\Psi \in \Xi_{BS(0,1)}$ be an instance of the CostMAB problem with daily ROI requirement $\lambda \geq 1$. Fix $\varepsilon > 0$, tolerance threshold. Let $\{A_t\}_{t=1}^T$ be the arms selected by Algorithm (5) applied to the Ψ. Then the expected number of intolerable constraint violations is bounded as:*

$$\mathbb{E}\left[J_T^\varepsilon\right] := \mathbb{E}\left[\#\left\{t \in [T] : \frac{\mu_{A_t}}{\nu_{A_t}} < \lambda - \epsilon\right\}\right] \leq \mathcal{O}\left(Kh(T;\epsilon,\lambda)\right)$$

with $h(T;\epsilon,\lambda) := 8T^{2/3}\log(T)\left(\frac{\lambda(\lambda+1)}{\epsilon}\right)^2$

Proof.

$$\begin{aligned}
\mathbb{E}\left[J_T^{\varepsilon}\right] :=& \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left\{\frac{\mu_{A_t}}{\nu_{A_t}} < \lambda - \varepsilon\right\}\right] && (4.69)\\
=& \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left\{\frac{\mu_{A_t}}{\nu_{A_t}} < \lambda - \varepsilon\right\}\middle|\mathcal{E}\right] \mathbb{P}\left(\mathcal{E}\right) + \\
& + \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left\{\frac{\mu_{A_t}}{\nu_{A_t}} < \lambda - \varepsilon\right\}\middle|\mathcal{E}^c\right] \mathbb{P}\left(\mathcal{E}^c\right) && (4.70)\\
\leq& \sum_{t=1}^{T} \mathbb{E}\left[\mathbb{1}\left\{\frac{\mu_{A_t}}{\nu_{A_t}} < \lambda - \varepsilon\right\}\middle|\mathcal{E}\right] + 4TT^{-2} && (4.71)\\
\leq& h(T;\varepsilon,\lambda)K + \frac{4}{T} = \mathcal{O}\left(h(T;\varepsilon,\lambda)K\right) && (4.72)
\end{aligned}$$

where in Line (4.70) we used the Law of Total Expectation, in Line (4.71) we used Lemma (4.1.2) and in Line (4.72) we used Lemma (4.2.1) to state that each selected arm can commit an ε-intolerable violation at most $h(T;\varepsilon,\lambda)$ times, with $h(T;\varepsilon,\lambda) := 8T^{2/3}\log(T)\left(\frac{\lambda(\lambda+1)}{\epsilon}\right)^2$. □

Remark 6. We underline two aspects of this result.

1. Fix a safety level $\eta \in (0,1)$. From Theorem (4.2.1) is trivial to find a threshold ε such that Algorithm (5) is η-quasi-safe with respect to ROI constraint with a tolerance ε. It suffices to select $\varepsilon \geq \frac{2\lambda(\lambda+1)\sqrt{2K\log(T)}}{T^{1/6}\sqrt{\eta}}$. Observe that for small values of η, the tolerability threshold ε can even become greater than $\lambda - 1$. Such an high tolerability threshold does not make sense, since in this case we are not considering the constraint anymore. We have to manage the trade-off between the level of safety η and the threshold ε.

2. In particular, observe that to ensure a sublinear expected number of intolerable constraint violations, we have only to require $\epsilon = \mathcal{O}\left(\frac{1}{T^{\alpha}}\right)$ with $\alpha < \frac{1}{6}$. Thus, to maintain sublinear expected number of intolerable constraint violation, the larger is T the smaller the tolerance level can be.

4.2.3 Regret analysis

To find a bound to the cumulative regret of Algorithm (5), we exploit again the idea of Clean Event in Definition (4.1.1).

Definition 4.2.1 (ROI feasible arm)**.** *Fix an instance Ψ of the CostMAB problem with daily ROI requirement $\lambda > 0$. An arm $a \in \mathcal{A}$ is said to be feasible at round t if*

$$\frac{\mu_a}{\nu_a} \geq \lambda \tag{4.73}$$

We call $\mathcal{A}_f := \left\{ a \in \mathcal{A} : \frac{\mu_a}{\nu_a} \geq \lambda \, \forall t \in [T] \right\}$ set of feasible arms.

Definition 4.2.2 (ROI empirically feasible arm)**.** *Fix an instance Ψ of the CostMAB problem with daily ROI requirement $\lambda \geq 1$. An arm $a \in \mathcal{A}$ is said to be empirically feasible at round t if*

$$\frac{\mathtt{UCB}_a(t-1)}{\mathtt{LCB}_a(t-1)} \geq \lambda \tag{4.74}$$

$$\mathtt{UCB}_a(t-1) \geq T^{-1/3} \tag{4.75}$$

We now would like to show that the optimal arm is always considered empirically feasible, as we have done for the BudgetLUCB algorithm in Corollary (4.1.1). This would lead us to bound the optimality gaps and then the regret. Unfortunately, this is not the case. In fact because of Constraint (4.75), if the optimal arm has an expected reward lower than $T^{-1/3}$, it could be considered empirically unfeasible, eventually. However we show that if this happens, choosing any suboptimal arm every round still leads to a sublinear regret. We formalize this idea in the following Lemma.

Lemma 4.2.2. *Let Ψ be an instance of the CostMAB problem with daily ROI requirement $\lambda \geq 1$. Under the Clean Event, the event*

$$\left\{ \sum_{a \in \mathcal{A}} \Delta_a N_a(T) \leq \sqrt{8KT \log(T)} + T^{2/3} \right\} \tag{4.76}$$

holds true. Where $N_a(T)$ is the number of time arm $a \in \mathcal{A}$ is chosen, running Algorithm (5) on the instance Ψ.

Proof. We split the proof in two cases: $\mu_\star < T^{-1/3}$ and $\mu_\star \geq T^{-1/3}$.
<u>Case 1 $\mu_\star < T^{-1/3}$:</u>
In this case the analysis is trivial: the optimality gap $\Delta_a := \mu_\star - \mu_a \leq T^{-1/3}$ for any arm $a \in \mathcal{A}$, thus

$$\sum_{a \in \mathcal{A}} \Delta_a N_a(T) \leq T^{-1/3} \sum_{a \in \mathcal{A}} N_a(T) = T^{2/3} \tag{4.77}$$

<u>Case 2 $\mu_\star \geq T^{-1/3}$:</u>

First, we underline that the optimal arm cannot be considered empirically unfeasible under the clean event.

Constraint (4.74) is always satisfied by the optimal arm observing that under the Clean Event, for any $t \in [T]$, $\texttt{UCB}_\star(t-1) \geq \mu_\star$ and $\texttt{LCB}_\star(t-1) \leq \nu_\star$. Constraint (4.75) is satisfied under $\mathcal{E}$ by the optimal arm because $\texttt{UCB}_\star(t-1) \geq \mu_\star \geq T^{-1/3}$.

This observation leads to the analogous result of Corollary (4.1.1): for any $t \in [T]$ the event

$$\{\texttt{UCB}_{A_t}(t-1) < \texttt{UCB}_\star(t-1)\} \tag{4.78}$$

is impossible under $\mathcal{E}$.

Thus we are able to replicate the exact same argument of Lemma (4.1.4) and conclude that $\forall a \in \mathcal{A}$ under the Clean Event:

$$\Delta_a \leq \sqrt{\frac{8\log(T)}{N_{A_t}(T)}} \tag{4.79}$$

almost surely.

We can conclude:

$$\sum_{a\in\mathcal{A}} \Delta_a N_a(T) \leq \sum_{a\in\mathcal{A}} \sqrt{8\log(T) N_a(T)} \tag{4.80}$$

$$\leq \sqrt{8\log(T) K \sum_{a\in\mathcal{A}} N_a(T)} = \sqrt{8KT\log(T)} \tag{4.81}$$

Where in the first inequality we used Inequality (4.79), and in the second Jensen's Inequality. □

We now combine the result of Lemma (4.2.2) and the high probability of Clean Event given by Lemma (4.1.2), to obtain an instance independent regret bound for Algorithm (5).

Theorem 4.2.2 (Instance independent regret bound for ROI-LUCB)**.** *For any instance $\Psi \in \Xi_{BS(0,1)}$ of the CostMAB problem with daily ROI requirement $\lambda \geq 1$, the regret of Algorithm (5) on Ψ is bounded as:*

$$\mathcal{R}_T \leq \mathcal{O}\left(T^{2/3}\right) \tag{4.82}$$

Where K is the cardinality of the arms' set $\mathcal{A}$ and $T \in \mathbb{N}$ the time horizon.

Proof. To prove the statement we will exploit the regret decomposition presented in Lemma (2.1.2).

$$
\begin{aligned}
\mathcal{R}_T =& \sum_{a=1}^{K} \mathbb{E}\left[N_a(T)\Delta_a\right] && (4.83)\\
=& \sum_{a=1}^{K} \mathbb{E}\left[N_a(T)\Delta_a|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) + \sum_{a=1}^{K} \mathbb{E}\left[N_a(T)\Delta_a|\mathcal{E}^c\right]\mathbb{P}\left(\mathcal{E}^c\right) && (4.84)\\
\leq& \sqrt{8KT\log(T)} + T^{2/3} + T^{-1} \leq \mathcal{O}\left(T^{2/3}\right) && (4.85)
\end{aligned}
$$

Where in Line (4.84) we use the law of total expectation. In Line (4.84) we use the fact that $\mathbb{P}\left(\mathcal{E}\right) \leq 1$, $\mathbb{P}\left(\mathcal{E}^c\right) \leq 4T^{-2}$ for Lemma (4.1.2), the fact that $\sum_{a=1}^{K} N_a(T) = T$ and the result from Lemma (4.2.2). □

Chapter 5

CostMTSSB algorithms

In this chapter, we focus on the CostMTSSB framework introduced in Chapter (3). We again propose two algorithms: the first one belonging to the budget quasi-safe class, the second one belonging to the ROI quasi-safe class. It is worth summarizing the idea behind the construction of the algorithms proposed in Chapter (4), as the same ideas will drive the construction of the algorithms in this chapter.

We start from a UCB-like algorithm that admits sublinear regret (in the CostMAB case it was UCB1, in the CostMTSSB case is CombUCB). This algorithm does not consider cost constraints and thus does not belongs to the quasi-safe class. Therefore, we modify the policy by adding constraints. These constraints must satisfy with high probability, i.e., under the Clean Event, two objectives:

- Make the algorithm quasi-safe
- Not lose the sublinearity property of the regret.

For the first objective, it is necessary to accurately choose optimistic constraints and verify that they make the algorithm quasi-safe. For the second goal, it is essential to verify that the selected constraints never eliminate the optimal arm (superarm in the CostMTSSB case): if this is verified, the analysis focuses on the rewards and we can use arguments similar to the unconstrained case, obtaining sublinear regret bounds. If the algorithm considers empirically non-feasible the optimal choice, we must ensure that playing a suboptimal action does not significantly impact the total regret.

We will therefore focus the analysis of the algorithms in this direction: verify that they are quasi-safe and that the optimal action is never considered empirically infeasible under the Clean Event, unless this results in a low impact on regret.

It is important to underline that we assume independence among the M tasks of the CostMTSSB problem. This is a strong assumption but, as explained in Chapter (3), because of the safety constraints the resulting problem is not trivial. The algorithm has to solve a combinatorial problem to select the best superarm that, in general, does not coincide with the superarm with the highest expected partial reward for each task.

5.1 CombBudgetLUCB algorithm

In this section we propose an algorithm for the CostMTSSB problem with daily budget b.

5.1.1 Algorithm description

The algorithm uses at each round the following optimistic guesses for the expected reward and expected cost of each of the $K \times M$ arms.

$$\texttt{UCB}_a^{(m)}(t-1) := \begin{cases} +\infty & \text{if } N_a^{(m)}(t-1) = 0 \\ \hat{\mu}_a^{(m)}(t-1) + \sqrt{\frac{2\log(T)}{N_a^{(m)}(t-1)}} & \text{otherwise} \end{cases} \tag{5.1}$$

$$\texttt{LCB}_a^{(m)}(t-1) := \begin{cases} -\infty & \text{if } N_a^{(m)}(t-1) = 0 \\ \max\left\{\hat{\nu}_a^{(m)}(t-1) - \sqrt{\frac{2\log(T)}{N_a^{(m)}(t-1)}}, 0\right\} & \text{otherwise} \end{cases} \tag{5.2}$$

Where $\hat{\mu}_a^{(m)}(t)$, $\hat{\nu}_a^{(m)}(t)$ are respectively the sample mean of partial rewards and partial costs based on the observations up to round t. Observe that, assuming independence between tasks, we will update these statistics every time the arm a is the m^{th} component of the selected superarm. At each round $t \in [T]$ the superarm is selected solving the optimistic optimization problem:

$$\underset{\mathbf{a}\in\mathcal{S}}{\operatorname{argmax}} \sum_{m=1}^{M} \texttt{UCB}_{\mathbf{a}}^{(m)}(t-1) \tag{5.3}$$

$$\text{s.t.} \sum_{m=1}^{M} \texttt{LCB}_{\mathbf{a}}^{(m)}(t-1) \le b \tag{5.4}$$

Algorithm 6 CombBudgetLUCB

```
Input: $\mathcal{A} = [K]$ arms set, $T \geq K$ time horizon, $b$ daily budget
for $t$= 1,...,$T$ do
    Choose superarm $\mathbf{A}_t$solution of the constrained optimization problem:
        $$\operatorname*{argmax}_{\mathbf{a} \in \mathcal{S}} \sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{a}}^{(m)}(t-1) \quad (5.5)$$
        $$\text{s.t} \sum_{m=1}^{M} \mathtt{LCB}_{\mathbf{a}}^{(m)}(t-1) \leq b \quad (5.6)$$
    Observe partial rewards $X_t^{(m)}$ and partial costs $Y_t^{(m)}$ for each task $m \in$
     $[M]$ and update confidence bounds.
```

5.1.2 Safety analysis

We prove that Algorithm (6) belongs to the class of budget quasi-safe algorithms defined in Definition (3.3.3).

Similarly to the CostMAB case, in the analysis of the proposed algorithms will be crucial the concept of Clean Event. We reformulate the definition of Clean Event to deal with the CostMTSSB case.

Definition 5.1.1 (Clean Event). *Let* $\Psi := \{(P_{\boldsymbol{a}}, Q_{\boldsymbol{a}}) : \boldsymbol{a} \in \mathcal{S}\}$ *be an instance of the CostMTSSB problem. We define Clean Event* ($\mathcal{E}$) *the event:*

$$\mathcal{E} := \Bigg\{ \forall (m, a, t) \in [M] \times [K] \times [T] \ \left|\mu_a^{(m)} - \hat{\mu}_a^{(m)}(t)\right| < \sqrt{\frac{2\log(T)}{N_a^{(m)}(t)}} \wedge$$
$$\wedge \left|\nu_a^{(m)} - \hat{\nu}_a^{(m)}(t)\right| < \sqrt{\frac{2\log(T)}{N_a^{(m)}(t)}} \Bigg\} \tag{5.7}$$

Intuitively, under the Clean Event, we can ensure for each arm of each task that the relative expected partial reward and expected partial cost are distant from their sample mean at most $\sqrt{\frac{2\log(T)}{N_a^{(m)}}}$

We now show that the Clean Event happens with high probability.

Lemma 5.1.1 (Probability of Clean Event). *Let* Ψ *be an instance of the CostMTSSB problem. Then, the probability of the event* $\mathcal{E}$ *is at least* $1 - \frac{4M}{T^2}$

Proof. To be concise we define

$$r_a^{(m)}(t) =\sqrt{\frac{2\log(T)}{N_a^{(m)}(t)}} \tag{5.8}$$

$$r_j =\sqrt{\frac{2\log(T)}{j}}, \quad \forall j \in [T] \tag{5.9}$$

Observe that:

$$\begin{aligned}
\mathcal{E} :=& \Big\{ \forall m \in [M]\ \forall a \in [K]\ \forall t \in [T],\ \left|\mu_a^{(m)} - \hat{\mu}_a^{(m)}(t)\right| < r_a^{(m)}(t) \wedge \\
& \wedge \left|\nu_a^{(m)} - \hat{\nu}_a^{(m)}(t)\right| < r_a^{(m)}(t) \Big\} && (5.10)\\
\supset& \Big\{ \forall m \in [M]\ \forall i \in [K]\ \forall j \in [T],\ \left|\mu_i^{(m)} - \hat{\mu}_{i,j}^{(m)}\right| < r_j \wedge \\
& \wedge \left|\nu_i^{(m)} - \hat{\nu}_{i,j}^{(m)}\right| < r_j \Big\} && (5.11)\\
=:& \ \tilde{\mathcal{E}} && (5.12)
\end{aligned}$$

Moreover, $\forall (i, j, m) \in [K] \times [T] \times [M]$:

$$\begin{aligned}
&\mathbb{P}\left(\left|\mu_i^{(m)} - \hat{\mu}_{i,j}^{(m)}\right| > r_j \vee \left|\nu_i^{(m)} - \hat{\nu}_{i,j}^{(m)}\right| > r_j\right) \\
&\leq \mathbb{P}\left(\left|\mu_i^{(m)} - \hat{\mu}_{i,j}^{(m)}\right| > r_j\right) + \mathbb{P}\left(\left|\nu_i^{(m)} - \hat{\nu}_{i,j}^{(m)}\right| > r_j\right) \\
&\leq 4T^{-4} && (5.13)
\end{aligned}$$

where we used the Union Bound and the Chernoff-Hoeffding Bound (2.1.1).

Thus we can conclude,

$$\begin{aligned}
\mathbb{P}(\mathcal{E}) \geq& \mathbb{P}\left(\tilde{\mathcal{E}}\right) && (5.14)\\
\geq& 1 - 4T^{-4}MKT \geq 1 - 4MT^{-2} && (5.15)
\end{aligned}$$

where, in Line (5.14) we used Equation (5.12) and in Line (5.15) we used Inequality (5.13) and the Union Bound. □

We now show that, under the Clean Event, Algorithm (6) cannot commit too many intolerable constraint violations.

Lemma 5.1.2. *Let Ψ be an instance of the CostMTSSB problem with daily budget b. Let $\{\boldsymbol{A}_t\}_{t=1}^{T}$ be the set of superarms selected by Algorithm (6) when applied to instance Ψ. Fix a tolerance threshold $\varepsilon > 0$. For every $t \in [T]$, the event*

$$\left\{ \sum_{m=1}^{M} \nu_{\boldsymbol{A}_t}^{(m)} > b + \varepsilon \right\} \tag{5.16}$$

is impossible, under the Clean Event $\mathcal{E}$, if $\forall m \in [M]$, $N_{\boldsymbol{A}_t}^{(m)}(t-1) > \frac{8M^2 \log(T)}{\varepsilon^2}$

Proof. First observe that:

$$\left\{ \forall m \in [M],\ N_{\mathbf{A}_t}^{(m)}(t-1) > \frac{8M^2 \log(T)}{\varepsilon^2} \right\} \equiv \left\{ \forall m \in [M] \frac{\varepsilon}{M} > 2 \sqrt{\frac{2 \log(T)}{N_{\mathbf{A}_t}^{(m)}(t-1)}} \right\} \tag{5.17}$$

Moreover, $\mathbf{A}_t$ is selected only if:

$$\sum_{m=1}^{M} \mathtt{LCB}_{\mathbf{A}_t}^{(m)}(t-1) \leq b \tag{5.18}$$

Suppose now by contradiction that $\sum_{m=1}^{M} \nu_{\mathbf{A}_t}^{(m)} > b + \varepsilon$.

$$\sum_{m=1}^{M} \nu_{\mathbf{A}_t}^{(m)} > b + \varepsilon \tag{5.19}$$

$$\implies \sum_{m=1}^{M} \left[\nu_{\mathbf{A}_t}^{(m)} - \mathtt{LCB}_{\mathbf{A}_t}^{(m)}(t-1) \right] > \varepsilon \tag{5.20}$$

$$\implies \exists m \in [M] :\ \nu_{\mathbf{A}_t}^{(m)} - \mathtt{LCB}_{\mathbf{A}_t}^{(m)}(t-1) > \varepsilon / M \tag{5.21}$$

$$\implies \exists m \in [M] :\ \nu_{\mathbf{A}_t}^{(m)} - \hat{\nu}_{\mathbf{A}_t}^{(m)} > \sqrt{\frac{2 \log(T)}{N_{\mathbf{A}_t}^{(m)}(t-1)}} \tag{5.22}$$

where Line (5.20) is implied by Inequality (5.18), Line (5.21) can be trivially proved by contradiction, Line (5.22) follows by the definition of LCB and Hypothesis (5.17).

We conclude the proof observing that the event in Line (5.22) contradicts the assumption of Clean Event. □

In other words Lemma (5.1.2) ensures that Algorithm (6) can commit at most $\frac{8M^2 \log(T)}{\varepsilon^2}$ intolerable violations for each of the MK arms.

We can conclude that CombBudgetLUCB is quasi-safe, as stated in the following theorem.

Theorem 5.1.1 (CombBudgetLUCB quasi-safety)**.** *Let Ψ be an instance of the CostMTSSB problem with daily budget b. Fix a tolerance threshold ε. Let $\{A_t\}_{t=1}^T$ be the sequence of arms selected by Algorithm (6) when applied to Ψ. Then the expected number of intolerable constraint violations is bounded as:*

$$\mathbb{E}\left[J_T\varepsilon\right] = \mathbb{E}\left[\#\left\{t \in [T] : \sum_{m=1}^{M} \nu_{\boldsymbol{A}_t}^{(m)} > b + \varepsilon\right\}\right] \leq \mathcal{O}\left(\frac{M^3 K \log(T)}{\varepsilon^2}\right) \tag{5.23}$$

Proof.

$$\begin{aligned}
\mathbb{E}\left[J_T^\varepsilon\right] =& \sum_{t=1}^{T}\sum_{m=1}^{M} \mathbb{E}\left[\mathbb{1}\left\{\nu_{\mathbf{A}_t}^{(m)} > b + \varepsilon\right\}\right] && (5.24)\\
=& \sum_{t=1}^{T}\sum_{m=1}^{M} \mathbb{E}\left[\mathbb{1}\left\{\nu_{\mathbf{A}_t}^{(m)} > b + \varepsilon\right\}\middle|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) + \\
& + \sum_{t=1}^{T}\sum_{m=1}^{M} \mathbb{E}\left[\mathbb{1}\left\{\nu_{\mathbf{A}_t}^{(m)} > b + \varepsilon\right\}\middle|\mathcal{E}^c\right]\mathbb{P}\left(\mathcal{E}^c\right) && (5.25)\\
\leq& \frac{8M^3\log(T)}{\varepsilon^2} + 4MT^{-1} && (5.26)\\
\leq& \mathcal{O}\left(\frac{M^3 K \log(T)}{\varepsilon^2}\right) && (5.27)
\end{aligned}$$

where in Line (5.60) we used the Law of Total Expectation, in Line (5.61) we used Lemma (5.1.2) and Lemma (5.1.1). □

5.1.3 Regret analysis

We shortly discuss the regret analysis showing that, under the Clean Event, Algorithm (6) never considers empirically infeasible the optimal superarm. This property, exactly as in the analysis of BudgetLUCB algorithm for the CostMAB case, leads to bounds that are the same as the unconstrained algorithm CombUCB (3).

Lemma 5.1.3. *Let Ψ be an instance of the CostMTSSB problem with daily budget b. Let $\{\boldsymbol{A}_t\}_{t=1}^T$ be the sequence of superarms selected by Algorithm (6) when applied to Ψ. Then, for every round $t \in [T]$:*

$$\sum_{m=1}^{M} \mathtt{UCB}_{\boldsymbol{A}_t}^{(m)}(t-1) \geq \sum_{m=1}^{M} \mathtt{UCB}_{\star}^{(m)}(t-1) \tag{5.28}$$

where we indicate with $\mathtt{UCB}_\star(t-1)$ the vector of upper confidence bounds of partial rewards of the optimal superarm.

Proof. By contradiction, if exists $t \in [T]$:

$$\sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{A}_t}^{(m)}(t-1) < \sum_{m=1}^{M} \mathtt{UCB}_{\star}^{(m)}(t-1) \tag{5.29}$$

then,

$$\sum_{m=1}^{M} \mathtt{UCB}_{\star}^{(m)}(t-1) > b \tag{5.30}$$

but under the Clean Event this implies:

$$\sum_{m=1}^{M} \mu_{\star}^{(m)} > b \tag{5.31}$$

that is in contradiction with the definition of optimal superarm. □

Exploiting this fact and Lemma (5.1.1), we can conclude:

Theorem 5.1.2. *Let Ψ be an instance of the CostMTSSB with daily budget b. Then Algorithm (6) applied to Ψ ensures a regret bounded as:*

$$\mathcal{R}_T \leq \mathcal{O}\left(47\sqrt{T\log(T)KM^2} + \left(\frac{\pi^2}{3}+1\right)KM^2\right)$$

Proof (Sketch). First, we observe that in absence of constraints (i.e. for $b > M$) the CombBudgetLUCB algorithm is a particular case of CombUCB, Algorithm (3).

Consider now the constrained case and focus on the event in which $\mathcal{E}$ holds true. The algorithm CombBudgetLUCB eliminates due to constraints only suboptimal arms, as proved in Lemma(5.1.3). Thus, we can analyze the regret conditional to $\mathcal{E}$ miming the proof of Theorem(2.2.2). The idea is that at each round the algorithm selects a superarm from a subset in which there's the optimal one, thus the same reasoning of the proof of Theorem (2.2.2) holds. We thus obtain

$$\mathbb{E}\left[R_T|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) \leq 47\sqrt{T\log(T)KM^2} + \left(\frac{\pi^2}{3}+1\right)KM^2, \tag{5.32}$$

where we indicate with R_T the stochastic regret cumulated up to T.

Recalling that $\mathcal{E}$ has probability at least $1 - MT^{-2}$ thanks to Lemma (5.1.1) and using the trivial bound $\mathbb{E}\left[R_T|\mathcal{E}^c\right] \leq T$, we obtain the thesis using the Law of Total Expectation:

$$\begin{aligned}\mathcal{R}_T =& \mathbb{E}\left[R_T|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) + \mathbb{E}\left[R_T|\mathcal{E}^c\right]\mathbb{P}\left(\mathcal{E}\right)\\ \leq& 47\sqrt{T\log(T)KM^2} + \left(\frac{\pi^2}{3}+1\right)KM^2 + 4MT^{-1}\end{aligned}$$

□

5.2 CombROI-LUCB

In this section we propose an algorithm for the CostMTSSB problem with daily ROI requirements.

5.2.1 Algorithm description

The algorithm uses at each round the optimist guesses for the expected reward and expected cost of each of the $K \times M$ arms, that we have defined in Equations (5.1, 5.2). At each round $t \in [T]$ the superarm is selected solving the optimistic optimization problem:

$$\operatorname*{argmax}_{\mathbf{a}\in\mathcal{S}} \sum_{m=1}^{M} \texttt{UCB}_{\mathbf{a}}^{(m)}(t-1) \tag{5.33}$$

$$\text{s.t.} \sum_{m=1}^{M} \frac{\texttt{UCB}_{\mathbf{a}}^{(m)}(t-1)}{\texttt{LCB}_{\mathbf{a}}^{(m)}(t-1)} \geq \lambda \tag{5.34}$$

$$\sum_{m=1}^{M} \texttt{UCB}_{\mathbf{a}}^{(m)} > T^{-1/3} \tag{5.35}$$

Algorithm 7 CombROI-LUCB

```
Input: $T$ time horizon, $\mathcal{S} = [K]^M$ superarm set, $\lambda$ daily ROI requirement
for t=1,...,T do
    Choose the superarm $\mathbf{A}_t$ solution of:
        $\operatorname*{argmax}_{\mathbf{a}\in\mathcal{S}} \sum_{m=1}^{M} \texttt{UCB}_{\mathbf{a}}^{(m)}(t-1)$
        $s.t. \frac{\sum_{m=1}^{M} \texttt{UCB}_{\mathbf{a}}^{(m)}(t-1)}{\sum_{m=1}^{M} \texttt{LCB}_{\mathbf{a}}^{(m)}(t-1)} \geq \lambda$
        $\sum_{m=1}^{M} \texttt{UCB}_{\mathbf{a}}^{(m)}(t-1) > T^{-1/3}$
    Observe $\forall m \in [M]$ the rewards $X_t^{(m)}$ and the costs $Y_t^{(m)}$ and update confidence bounds
```

The idea of the algorithm is the same of the ROI-LUCB, Algorithm (5): the policy excludes those superarms which reward is estimated to be too low. Doing this we can show that the algorithm belongs to the ROI quasi-safe class. The price of this choice is a looser regret bound.

5.2.2 Safety analysis

We prove that Algorithm (7) belongs to the class of ROI quasi-safe algorithms defined in Definition (3.3.5). Again we exploit the idea of Clean Event, that is the same of Definition (5.1.1). Note that the Definition (5.1.1) and Lemma (5.1.1) are properties of the CostMTSSB framework independently from the constraints we apply. Thus, it's enough to show that, under the Clean Event, Algorithm (7) cannot commit too many intolerable constraint violations.

Lemma 5.2.1. *Let Ψ be an instance of the CostMTSSB problem with daily ROI requirement $\lambda \geq 1$. Let $\{\boldsymbol{A}_t\}_{t=1}^T$ be the set of superarms selected by Algorithm (7) when applied to instance Ψ. Fix a tolerance threshold $\varepsilon > 0$. For every $t \in [T]$, the event*

$$\left\{ \sum_{m=1}^{M} \frac{\mu_{\boldsymbol{A}_t}^{(m)}}{\nu_{\boldsymbol{A}_t}^{(m)}} < \lambda - \varepsilon \right\} \tag{5.36}$$

is impossible, under the Clean Event $\mathcal{E}$ if $\forall m \in [M]$, $N_{\boldsymbol{A}_t}^{(m)}(t-1) > h(T; \lambda, \varepsilon)$ with

$$h(T; \epsilon, \lambda) = 8T^{2/3} \log(T) \left(\frac{\lambda(\lambda+1)M}{\epsilon} \right)^2$$

Proof. Let's first remark that $\mathbf{A}_t$ can be selected only if the following hold:

$$\frac{\sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{A}_t}^{(m)}(t)}{\sum_{m=1}^{M} \mathtt{LCB}_{\mathbf{A}_t}^{(m)}(t)} > \lambda \tag{5.37}$$

$$\sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{A}_t}^{(m)}(t) > T^{-1/3} \tag{5.38}$$

We want to show that exists $h(T; \epsilon, \lambda)$ such that if $N_{\mathbf{A}_t}(t) \geq h(T; \epsilon, \lambda)$ then (5.37) implies $\mu_{\mathbf{A}_t}/\nu_{\mathbf{A}_t} \geq \lambda - \epsilon$.

For sake of clearness in the computations, let us define the following variables:

$$\hat{N}_{\mathbf{A}_t}^{(m)}(t) := T^{-2/3} N_{\mathbf{A}_t}^{(m)}(t) \qquad \eta^{(m)} := \sqrt{\frac{2\log(T)}{\hat{N}_{\mathbf{A}_t}^{(m)}(t)}} \qquad \eta := \sum_{m=1}^{M} \eta^{(m)}$$

It follows:

$$r_{\mathbf{A}_t}^{(m)}(t) := \sqrt{\frac{2\log(T)}{N_{\mathbf{A}_t}^{(m)}(t)}} = T^{-1/3} \eta^{(m)} \tag{5.39}$$

From these definitions and under the $\mathcal{E}$ holds:

$$\sum_{m=1}^{M} \mu_{\mathbf{A}_t}^{(m)} \geq \sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{A}_t}^{(m)}(t) - 2r_{\mathbf{A}_t}^{(m)}(t) \geq T^{-1/3}(1-2\eta) \tag{5.40}$$

where the first inequality comes from $\mathcal{E}$ assumption and the last from (5.39) summing over $m \in [M]$.
Hence:

$$\sum_{m=1}^{M} r_{\mathbf{A}_t}^{(m)}(t) \leq T^{-1/3}\eta = \frac{\eta}{1-2\eta}(1-2\eta)T^{-1/3} \leq \frac{\eta}{1-2\eta}\sum_{m=1}^{M} \mu_{\mathbf{A}_t}^{(m)} \tag{5.41}$$

using (5.39) and (5.40).
So we can find an upper-bound for $\sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{A}_t}^{(m)}(t)$ as follows:

$$\sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{A}_t}^{(m)}(t) \leq \sum_{m=1}^{M} \left[\mu_{\mathbf{A}_t} + 2r_{\mathbf{A}_t}^{(m)}(t)\right] \leq \left(1 + 2\frac{\eta}{1-2\eta}\right)\sum_{m=1}^{M} \mu_{\mathbf{A}_t}^{(m)} \tag{5.42}$$

exploiting again the $\mathcal{E}$ in the first inequality and using (5.41) in the second.
For what concerns the costs we can divide the analysis in two cases.
Case 1:

$$\sum_{m=1}^{M} \nu_{\mathbf{A}_t}^{(m)} \leq T^{-1/3}/\lambda \tag{5.43}$$

In this case the thesis is valid under an hypothesis we'll later check to be true:

$$\frac{\sum_{m=1}^{M} \mu_{\mathbf{A}_t}^{(m)}}{\sum_{m=1}^{M} \nu_{\mathbf{A}_t}^{(m)}} \geq \frac{T^{-1/3}(1-2\eta)}{T^{-1/3}/\lambda} = (1-2\eta)\lambda \geq \lambda - \varepsilon \tag{5.44}$$

where we used (5.40) and (5.43) in the first inequality and the last holds if

$$\forall m \in [M],\ N_{\mathbf{A}_t}(t-1) \geq \frac{8\lambda^2 M^2 T^{2/3}\log(T)}{\varepsilon^2} \tag{5.45}$$

Case 2:

$$\sum_{m=1}^{M} \nu_{\mathbf{A}_t}^{(m)} > T^{-1/3}/\lambda \tag{5.46}$$

In this case we have to lower-bound $\sum_{m=1}^{M} \mathtt{LCB}_{\mathbf{A}_t}^{(m)}(t)$ to confirm the thesis.
Let's first notice that:

$$\sum_{m=1}^{M} r_{\mathbf{A}_t}^{(m)}(t) = T^{-1/3}\eta = \frac{T^{-1/3}}{\lambda}\lambda\eta \leq \lambda\eta\sum_{m=1}^{M} \nu_{\mathbf{A}_t} \tag{5.47}$$

using respectively (5.39) and (5.46).
It follows:

$$\sum_{m=1}^{M} \mathtt{LCB}^{(m)}_{\mathbf{A}_t}(t) \geq \sum_{m=1}^{M} \nu^{(m)}_{\mathbf{A}_t} - 2\sum_{m=1}^{M} r^{(m)}_{\mathbf{A}_t}(t) \geq (1-2\eta\lambda)\sum_{m=1}^{M} \nu^{(m)}_{\mathbf{A}_t} \tag{5.48}$$

Finally:

$$\lambda \leq \frac{\sum_{m=1}^{M} \mathtt{UCB}^{(m)}_{\mathbf{A}_t}(t)}{\sum_{m=1}^{M} \mathtt{LCB}^{(m)}_{\mathbf{A}_t}(t)} \leq \frac{\sum_{m=1}^{M} \mu^{(m)}_{\mathbf{A}_t}}{\sum_{m=1}^{M} \nu^{(m)}_{\mathbf{A}_t}} \frac{1}{(1-2\eta)(1-2\eta\lambda)} \tag{5.49}$$

where the first inequality raises from (5.37) and the second from (5.42) and (5.48).
To conclude the proof we have to find $h(T;\epsilon;\lambda)$ such that

$$\forall m \in [M] \;\; N^{(m)}_{\mathbf{A}_t}(t) \geq h(T;\epsilon,\lambda)$$

implies:

$$\lambda(1-2\eta)(1-2\eta\lambda) \geq \lambda - \epsilon \tag{5.50}$$

and such that the condition (5.45) holds.
It's easy to check that:

$$h(T;\epsilon,\lambda) = 8T^{2/3}\log(T)\left(\frac{\lambda(\lambda+1)M}{\epsilon}\right)^2 \tag{5.51}$$

leads to that conclusion.
In fact in this case $\forall m \in [M] \;\; N^{(m)}_{\mathbf{A}_t}(t) \geq h(T;\epsilon,\lambda)$ implies:

$$\forall m \in [M] \;\; N^{(m)}_{\mathbf{A}_t}(t) \geq 8\log(T)\left(\frac{\lambda(\lambda+1)M}{\epsilon}\right)^2 \tag{5.52}$$

$$\implies \forall m \in [M] \;\; \sqrt{\frac{2\log(T)}{\hat{N}^{(m)}_{\mathbf{A}_t}(t)}} \leq \frac{\epsilon/M}{2\lambda(\lambda+1)} \tag{5.53}$$

$$\implies \eta \leq \frac{\epsilon}{2\lambda(\lambda+1)} \tag{5.54}$$

$$\implies 1-2\eta(1+\lambda) \geq \frac{\lambda-\epsilon}{\lambda} \tag{5.55}$$

$$\implies 1-2\eta-2\eta\lambda+4\eta^2\lambda \geq \frac{\lambda-\epsilon}{\lambda} \tag{5.56}$$

$$\implies \lambda(1-2\eta)(1-2\eta\lambda) \geq \lambda-\epsilon \tag{5.57}$$

Note that (5.45) is implied by (5.52), concluding the proof. □

This result leads us to conclude that each arm, under the $\mathcal{E}$, can commit at most $h(T; \epsilon; \lambda)$ intolerable violations of the constraint.

We can conclude that CombROI-LUCB is quasi-safe, as stated in the following theorem

Theorem 5.2.1 (CombROI-LUCB quasi-safety). *Let Ψ be an instance of the CostMTSSB problem with daily ROI minimum requirement $\lambda \geq 1$. Fix a tolerance threshold ε. Let $\{A_t\}_{t=1}^{T}$ be the sequence of arms selected by Algorithm (7) when applied to Ψ. Then the expected number of intolerable constraint violations is bounded as:*

$$\mathbb{E}\left[J_T^{\varepsilon}\right] = \mathbb{E}\left[\#\left\{t \in [T] : \sum_{m=1}^{M} \frac{\mu_{\mathbf{A}_t}^{(m)}}{\nu_{\mathbf{A}_t}^{(m)}} < \lambda - \varepsilon\right\}\right] \leq \mathcal{O}\left(MKh(T; \lambda, \varepsilon)\right) \quad (5.58)$$

with $h(T; \epsilon, \lambda) := 8T^{2/3}\log(T)\left(\frac{\lambda(\lambda+1)}{\epsilon}\right)^2$

Proof.

$$\begin{aligned}
\mathbb{E}\left[J_T^{\varepsilon}\right] =& \sum_{t=1}^{T}\sum_{m=1}^{M} \mathbb{E}\left[\mathbb{1}\left\{\frac{\mu_{\mathbf{A}_t}^{(m)}}{\nu_{\mathbf{A}_t}^{(m)}} < \lambda - \varepsilon\right\}\right] && (5.59)\\
=& \sum_{t=1}^{T}\sum_{m=1}^{M} \mathbb{E}\left[\mathbb{1}\left\{\frac{\mu_{\mathbf{A}_t}^{(m)}}{\nu_{\mathbf{A}_t}^{(m)}} < \lambda - \varepsilon\right\}\middle|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) + \\
&+ \sum_{t=1}^{T}\sum_{m=1}^{M} \mathbb{E}\left[\mathbb{1}\left\{\frac{\mu_{\mathbf{A}_t}^{(m)}}{\nu_{\mathbf{A}_t}^{(m)}} < \lambda - \varepsilon\right\}\middle|\mathcal{E}^c\right]\mathbb{P}\left(\mathcal{E}^c\right) && (5.60)\\
\leq& MKh(T; \lambda, \varepsilon) + 4MT^{-1} && (5.61)\\
\leq& \mathcal{O}\left(MKh(T; \lambda, \varepsilon)\right) && (5.62)
\end{aligned}$$

where in Line (5.60) we used the Law of Total Expectation and in Line (5.61) we used Lemma (5.2.1) and Lemma (5.1.1). □

5.2.3 Regret analysis

We shortly discuss the regret analysis showing that, under the Clean Event, Algorithm (7) cannot consider empirically infeasible the optimal superarm, unless its expected reward is lower than $T^{-1/3}$. This property, exactly as in the analysis of ROI-LUCB algorithm for the CostMAB case, leads sublinear regret bounds. In particular, if the expected reward of the optimal arm is greater than $T^{-1/3}$, under the Clean Event, the analysis coincides with the one of Theorem (2.2.2). If the expected reward of the optimal arm is lower than $T^{-1/3}$, we obtain the trivial bound $T^{2/3}$, since the maximum optimality gap is $T^{-1/3}$.

Lemma 5.2.2. *Let Ψ be an instance of the CostMTSSB problem with daily ROI requirement $\lambda \geq 1$. Let $\{\boldsymbol{A}_t\}_{t=1}^{T}$ be the sequence of superarms selected by Algorithm (7) when applied to Ψ. Then, for every round $t \in [T]$ if $\sum_{m=1}^{M} \mathtt{UCB}_{\star}^{(m)} > T^{-1/3}$:*

$$\sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{A}_t}^{(m)}(t-1) \geq \sum_{m=1}^{M} \mathtt{UCB}_{\star}^{(m)}(t-1) \tag{5.63}$$

where we indicate with $\mathtt{UCB}_{\star}(t-1)$ the vector of upper confidence bounds of partial rewards of the optimal superarm.

Proof. By contradiction, if exists $t \in [T]$:

$$\sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{A}_t}^{(m)}(t-1) < \sum_{m=1}^{M} \mathtt{UCB}_{\star}^{(m)}(t-1) \tag{5.64}$$

then,

$$\sum_{m=1}^{M} \frac{\mathtt{UCB}_{\star}^{(m)}(t-1)}{\mathtt{UCB}_{\star}^{(m)}(t-1)} < \lambda \tag{5.65}$$

but under the Clean Event this implies:

$$\sum_{m=1}^{M} \frac{\mu_{\star}^{(m)}}{\nu_{\star}^{(m)}} < \lambda \tag{5.66}$$

that is in contradiction with the definition of optimal arm. □

We can conclude that the worst case regret bound is $\mathcal{O}\left(T^{2/3}\right)$, obtained if $\sum_{m=1}^{M} \mu_{\star}^{(m)} < T^{-1/3}$, as stated in the following theorem.

Theorem 5.2.2. *Let Ψ be an instance of the CostMTSSB with daily ROI requirement $\lambda \geq 1$. Then Algorithm (7) applied to Ψ ensures a regret bounded as:*

$$\mathcal{R}_T \leq \mathcal{O}\left(T^{2/3}\right)$$

Proof (Sketch). We divide the analysis in two cases: $\sum_{m=1}^{M} \mu_{\star}^{(m)} < T^{-1/3}$ and $\sum_{m=1}^{M} \mu_{\star}^{(m)} \geq T^{-1/3}$.

<u>Case 1:</u> If $\sum_{m=1}^{M} \mu_{\star}^{(m)} < T^{-1/3}$ we can trivially bound the regret. In fact $\forall \mathbf{a} \in \mathcal{S}$ $\sum_{m=1}^{M} \mu_{\star}^{(m)} - \sum_{m=1}^{M} \mu_{\mathbf{a}}^{(m)} < T^{-1/3}$. Thus,

$$\mathcal{R}_T \leq \sum_{t=1}^{T} T^{-1/3} = T^{2/3}.$$

Case 2: We observe that in absence of constraints the CombROI-LUCB algorithm is a particular case of CombUCB, Algorithm (3). Consider now the constrained case and focus on the event in which $\mathcal{E}$ holds. Under $\mathcal{E}$ $\sum_{m=1}^{M} \mu_{\star}^{(m)} \geq T^{-1/3}$ implies that Constraint (5.35) is always satisfied. Thus, for Lemma (5.2.2), the algorithm never eliminates optimal arms. Thus, we can analyze the regret conditional to $\mathcal{E}$ miming the proof of Theorem(2.2.2) obtaining

$$\mathbb{E}\left[R_T|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) \leq 47\sqrt{T\log(T)KM^2} + \left(\frac{\pi^2}{3}+1\right)KM^2, \tag{5.67}$$

where we indicate with R_T the stochastic regret cumulated up to T.

Recalling that $\mathcal{E}$ has probability at least $1 - MT^{-2}$ thanks to Lemma (5.1.1) and using the trivial bound $\mathbb{E}\left[R_T|\mathcal{E}^c\right] \leq T$, we obtain the thesis using the Law of Total Expectation:

$$\begin{aligned}\mathcal{R}_T =& \mathbb{E}\left[R_T|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) + \mathbb{E}\left[R_T|\mathcal{E}^c\right]\mathbb{P}\left(\mathcal{E}^c\right)\\ \leq& 47\sqrt{T\log(T)KM^2} + \left(\frac{\pi^2}{3}+1\right)KM^2 + 4MT^{-1}\end{aligned}$$

We consider as bound the worst between Case 1 and Case 2.

□

5.3 A ROI-budget quasi-safe algorithm

We conclude the discussion by showing how the proposed CombBudgetLUCB and CombROI-LUCB algorithms can be combined to obtain a quasi-safe algorithm with respect to both budget and ROI constraints.

Given daily budget b and the minimum ROI threshold $\lambda \geq 1$, we define the following algorithm

Algorithm 8 ROI-BudgetCLUCB

```
Input: $T$ time horizon, $\mathcal{S} = [K]^M$ superarm set, $\lambda$ daily ROI requirement,
       $b$ daily budget
for $t$=1,...,$T$ do
    Choose the superarm $\mathbf{A}_t$ solution of:
        $$\begin{aligned} \underset{\mathbf{a}\in\mathcal{S}}{\text{argmax}} \quad & \sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{a}}^{(m)}(t-1) \\ s.t. \quad & \frac{\sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{a}}^{(m)}(t-1)}{\sum_{m=1}^{M} \mathtt{LCB}_{\mathbf{a}}^{(m)}(t-1)} \geq \lambda \\ & \sum_{m=1}^{M} \mathtt{UCB}_{\mathbf{a}}^{(m)}(t-1) > T^{-1/3} \\ & \sum_{m=1}^{M} \mathtt{LCB}_{\mathbf{a}}^{(m)}(t-1) \leq b \end{aligned}$$
    Observe $\forall m \in [M]$ the rewards $X_t^{(m)}$ and the costs $Y_t^{(m)}$ and update
      confidence bounds
```

Once analyzed the quasi-safety of algorithm CombBudgetLUCB and CombROI-LUCB is trivial to extend the results to the ROI-BudgetCLUCB to conclude that it is quasi-safe with respect to both budget and ROI constraints. Indeed, we can reproduce the proofs of Lemma (5.1.2) and Lemma (5.2.1) to state:

Lemma 5.3.1. *Let Ψ be an instance of the CostMTSSB problem with daily budget $b \in (0, M]$ and ROI requirement $\lambda \geq 1$. Let $\{\boldsymbol{A}_t\}_{t=1}^T$ be the set of superarms selected by ROI-BudgetCLUCB when applied to instance Ψ. Fix a tolerance threshold $\varepsilon > 0$. For every $t \in [T]$, the event*

$$\left\{ \sum_{m=1}^{M} \frac{\mu_{\boldsymbol{A}_t}^{(m)}}{\nu_{\boldsymbol{A}_t}^{(m)}} < \lambda - \varepsilon \vee \sum_{m=1}^{M} \nu_{\boldsymbol{A}_t}^{(m)} > b + \varepsilon \right\} \tag{5.68}$$

is impossible, under the Clean Event $\mathcal{E}$, if

$$\forall m \in [M],\ N_{\boldsymbol{A}_t}^{(m)}(t-1) > \max\left\{ h(T;\lambda,\varepsilon),\ \frac{8M^2\log(T)}{\varepsilon^2} \right\}$$

with

$$h(T;\epsilon,\lambda) = 8T^{2/3}\log(T)\left(\frac{\lambda(\lambda+1)M}{\epsilon}\right)^2$$

Combining this lemma with the high probability of the Clean Event, we can prove the budget and ROI quasi-safety of Algorithm (8).

Theorem 5.3.1 (ROI-BudgetCLUCB quasi-safety). *Let Ψ be an instance of the CostMTSSB problem with daily ROI minimum requirement λ and daily budget b. Fix a tolerance threshold ε. Let $\{\boldsymbol{A}_t\}_{t=1}^{T}$ be the sequence of superarms selected by the ROI-BudgetCLUCB algorithm when applied to Ψ. Then the expected number of intolerable constraint violations is bounded as:*

$$\mathbb{E}\left[\#\left\{t \in [T] : \sum_{m=1}^{M} \frac{\mu_{\boldsymbol{A}_t}^{(m)}}{\nu_{\mathbf{A}_t}^{(m)}} < \lambda - \varepsilon \vee \sum_{m=1}^{M} \nu_{\mathbf{A}_t}^{(m)} > b + \varepsilon\right\}\right] \\ \leq \mathcal{O}\left(MK \max\left\{h(T;\lambda,\varepsilon),\ \frac{8M^2 \log(T)}{\varepsilon^2}\right\}\right)$$

with $h(T;\epsilon,\lambda) := 8T^{2/3}\log(T)\left(\frac{\lambda(\lambda+1)}{\epsilon}\right)^2$

Proof (Sketch). If $\mathcal{E}$ holds true we state that the events

$$\left\{\sum_{m=1}^{M} \frac{\mu_{\mathbf{A}_t}^{(m)}}{\nu_{\mathbf{A}_t}^{(m)}} < \lambda - \varepsilon\right\} \qquad \text{and} \qquad \left\{\sum_{m=1}^{M} \nu_{\mathbf{A}_t}^{(m)} > b + \varepsilon\right\}$$

are impossible if $\forall m \in [M]$ $N_{\mathbf{A}_t}^{(m)}(t) \geq \max\left\{h(T;\lambda,\varepsilon),\ \frac{8M^2\log(T)}{\varepsilon^2}\right\}$. This can be proved reproducing the proofs of Lemma (5.2.1) and Lemma (5.1.2), respectively. Thus, using the Union Bound the event

$$\left\{\sum_{m=1}^{M} \frac{\mu_{\mathbf{A}_t}^{(m)}}{\nu_{\mathbf{A}_t}^{(m)}} < \lambda - \varepsilon \vee \sum_{m=1}^{M} \nu_{\mathbf{A}_t}^{(m)} > b + \varepsilon\right\}$$

is impossible, under $\mathcal{E}$, if $\forall m \in [M]$ $N_{\mathbf{A}_t}^{(m)}(t) \geq \max\left\{h(T;\lambda,\varepsilon),\ \frac{8M^2\log(T)}{\varepsilon^2}\right\}$. This means that any of the MK arms can be part of a superarm that commits an intolerable violation at most *max* $\left\{h(T;\lambda,\varepsilon),\ \frac{8M^2\log(T)}{\varepsilon^2}\right\}$ times.

We can conclude the proof combining this fact and the high probability of the Clean Event with the Law of Total Expectation, exactly as we have done in the proof of Theorem (5.2.1)

□

Finally, we show that the regret admits a sublinear bound.

Theorem 5.3.2. *Let Ψ be an instance of the CostMTSSB with daily ROI requirement λ and daily budget b. Then Algorithm (8) applied to Ψ ensures a regret bounded as:*

$$\mathcal{R}_T \leq \mathcal{O}\left(T^{2/3}\right)$$

Proof (Sketch). The idea of the proof is the same of the one of Theorem (5.2.2). We divide two cases.

Case 1: $\sum_{m=1}^{M} \mu_{\star}^{(m)} < T^{-1/3}$
In this case $\forall \mathbf{a} \in \mathcal{S}$ $\sum_{m=1}^{M} \mu_{\star}^{(m)} - \sum_{m=1}^{M} \mu_{\mathbf{a}}^{(m)} < T^{-1/3}$. Thus,

$$\mathcal{R}_T \leq \sum_{t=1}^{T} T^{-1/3} = T^{2/3}.$$

Case 2: $\sum_{m=1}^{M} \mu_{\star}^{(m)} \geq T^{-1/3}$
We observe that in absence of constraints the ROI-BudgetCLUB algorithm is a particular case of CombUCB, Algorithm (3). Consider now the constrained case and focus on the event in which $\mathcal{E}$ holds true. Under $\mathcal{E}$ the algorithm never eliminates optimal arms. This can be proven mimic proofs of Lemma (5.1.3) and Lemma(5.2.2). Thus, we can analyze the regret conditional to $\mathcal{E}$ as we have done in the proof of Theorem(2.2.2) obtaining

$$\mathbb{E}\left[R_T|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) \leq 47\sqrt{T\log(T)KM^2} + \left(\frac{\pi^2}{3}+1\right)KM^2, \tag{5.69}$$

where we indicate with R_T the stochastic regret cumulated up to T.

Recalling that $\mathcal{E}$ has probability at least $1 - MT^{-2}$ thanks to Lemma (5.1.1) and using the trivial bound $\mathbb{E}\left[R_T|\mathcal{E}^c\right] \leq T$, we obtain the thesis using the Law of Total Expectation:

$$\begin{aligned}\mathcal{R}_T =& \mathbb{E}\left[R_T|\mathcal{E}\right]\mathbb{P}\left(\mathcal{E}\right) + \mathbb{E}\left[R_T|\mathcal{E}^c\right]\mathbb{P}\left(\mathcal{E}^c\right)\\ \leq& 47\sqrt{T\log(T)KM^2} + \left(\frac{\pi^2}{3}+1\right)KM^2 + 4MT^{-1}\end{aligned}$$

We consider as bound the worst between Case 1 and Case 2.

□

Chapter 6

Conclusions and future works

6.1 Conclusions

We analyzed the bid optimization problem with daily budget and daily ROI constraints. We introduced the Multi-Armed Bandit framework with cost feedback (CostMAB) and the Multi-Task Stochastic Semi-Bandit framework with cost feedback (CostMTSSB) to model the case of single and multiple campaigns, respectively. The main goal of the thesis was to overcome the limitations imposed by the impossibility to propose safe algorithms that admit sublinear regret. We studied budget constraints and ROI constraints separately. We introduced the class of budget (resp. ROI) quasi-safe algorithms, i.e., algorithms that admit a sublinear bound on the expected number of budget (resp. ROI) constraints violations, provided a threshold of tolerability $\varepsilon > 0$ in the size of those violations. We exploited the idea of Optimism in the Face of Uncertainty to propose four algorithms belonging to the quasi-safe class and with sublinear regret bounds. The first two apply to the CostMAB context and are BudgetLUCB and ROI-LUCB. The former admits a regret bound that is $\mathcal{O}(\sqrt{KT\log(T)})$. It belongs to the budget quasi-safe class admitting a bound on the expected number of intolerable constraint violations that is logarithmic in T and proportional to $1/\varepsilon^2$. The latter admits a regret bound that is $\mathcal{O}\left(T^{2/3}\right)$. It belongs to the ROI quasi-safe class admitting a bound on the expected number of intolerable constraint violation that is $\mathcal{O}\left(T^{2/3}\log(T)\right)$ with respect to T and proportional to $1/\varepsilon^2$.

The second two policies apply to the CostMTSSB context and are CombBudgetLUCB and CombROI-LUCB. The former admits a regret bound that is $\mathcal{O}(\sqrt{M^2KT\log(T)})$. It belongs to the budget quasi-safe class admitting a bound on the expected number of intolerable constraint violations that is logarithmic in T and proportional to $1/\varepsilon^2$. The latter admits a regret bound that is $\mathcal{O}\left(T^{2/3}\right)$. It belongs to the ROI quasi-safe class admitting a bound on the expected number of intolerable constraint violation that is $\mathcal{O}\left(T^{2/3}\log(T)\right)$ with respect to T and proportional to $1/\varepsilon^2$.

Finally, we showed how CombBudgetLUCB and CombROI-LUCB can be combined to obtain ROI-BudgetCLUCB algorithm that is quasi-safe with respect to both the budget and ROI constraints.

6.2 Future works

There are several possible developments of this work, both from a modeling and theoretical point of view.

In order to have a more realistic model of the bid optimization problem, one could extend the idea of a quasi-safe ROI algorithm to the Combinatorial Multi-Armed Bandit context with cost feedback in which correlation between arms is assumed. Another development line could be to propose a different definition of regret that explicitly includes cost feedback, penalizing high-cost arms. In this way one should develop algorithms that are safe and offer strong theoretical guarantees on the so defined regret. Finally, from a theoretical perspective, one could extend the problem to a continuous space of actions by studying a constrained Stochastic Bandit problem. This problem has recently been studied in [1] in the case of linear constraints. An interesting line of work is to extend the analysis to the case of nonlinear convex constraints.

Bibliography

[1] Sanae Amani, Mahnoosh Alizadeh, and Christos Thrampoulidis. Linear stochastic bandits under safety constraints. *arXiv preprint arXiv:1908.05814*, 2019.

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

[3] Sébastien Bubeck and Nicolo Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *arXiv preprint arXiv:1204.5721*, 2012.

[4] Robert R Bush and Frederick Mosteller. A stochastic model with applications to learning. *The Annals of Mathematical Statistics*, pages 559–585, 1953.

[5] Matteo Castiglioni, Alessandro Nuara, Giulia Romano, Giorgio Spadaro, Francesco Trovò, and Nicola Gatti. Safe online bid optimization with return-on-investment constraints. Working paper, 2021.

[6] Olivier Chapelle and Lihong Li. An empirical evaluation of thompson sampling. *Advances in neural information processing systems*, 24:2249–2257, 2011.

[7] Wenkui Ding, Tao Qin, Xu-Dong Zhang, and Tie-Yan Liu. Multi-armed bandit with budget constraint and variable costs. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 27, 2013.

[8] Aurélien Garivier, Pierre Ménard, and Gilles Stoltz. Explore first, exploit next: The true shape of regret in bandit problems. *Mathematics of Operations Research*, 44(2):377–399, 2019.

[9] Margherita Gasparini, Alessandro Nuara, Francesco Trovò, Nicola Gatti, and Marcello Restelli. Targeting optimization for internet adver-

tising by learning from logged bandit feedback. In *2018 International Joint Conference on Neural Networks (IJCNN)*, pages 1–8. IEEE, 2018.

[10] Negin Golrezaei, Ilan Lobel, and Renato Paes Leme. Auction design for roi-constrained buyers. *Available at SSRN 3124929*, 2018.

[11] EM Italia, A Nuara, Francesco Trovo, Marcello Restelli, N Gatti, and E Dellavalle. Internet advertising for non-stationary environments. In *International Workshop on Agent-Mediated Electronic Commerce*, pages 1–15, 2017.

[12] David Kempe, Jon Kleinberg, and Éva Tardos. Maximizing the spread of influence through a social network. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 137–146, 2003.

[13] Branislav Kveton, Zheng Wen, Azin Ashkan, and Csaba Szepesvari. Tight regret bounds for stochastic combinatorial semi-bandits. In *Artificial Intelligence and Statistics*, pages 535–543. PMLR, 2015.

[14] Tze Leung Lai and Herbert Robbins. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1):4–22, 1985.

[15] Tor Lattimore and Csaba Szepesvári. *Bandit algorithms.* Cambridge University Press, 2020.

[16] David JC MacKay and David JC Mac Kay. *Information theory, inference and learning algorithms.* Cambridge university press, 2003.

[17] Alessandro Nuara, Francesco Trovo, Nicola Gatti, and Marcello Restelli. A combinatorial-bandit algorithm for the online joint bid/budget optimization of pay-per-click advertising campaigns. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[18] Alessandro Nuara, Francesco Trovò, Nicola Gatti, and Marcello Restelli. Online joint bid/daily budget optimization of internet advertising campaigns. *arXiv preprint arXiv:2003.01452*, 2020.

[19] Aleksandrs Slivkins. Introduction to multi-armed bandits. *arXiv preprint arXiv:1904.07272*, 2019.

[20] Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction.* MIT press, 2018.

[21] William 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.

[22] Francesco Trovò, Stefano Paladino, Marcello Restelli, and Nicola Gatti. Budgeted multi–armed bandit in continuous action space. In *Proceedings of the Twenty-second European Conference on Artificial Intelligence*, pages 560–568, 2016.