POLITECNICO
MILANO 1863

POLITECNICO DI MILANO
DIPARTIMENTO DI ELETTRONICA, INFORMAZIONE E BIOINGEGNERIA
DOCTORAL PROGRAM IN INFORMATION TECHNOLOGY

PRICING AND ADVERTISING STRATEGIES IN E-COMMERCE SCENARIOS

Doctoral Dissertation of:
Giulia Romano

Supervisor:
Prof. Nicola Gatti

Tutor:
Prof. Cristina Silvano

The Chair of the Doctoral Program:
Prof. Luigi Piroddi

2023 – Cycle XXXV

Abstract

This thesis revolves around the problem of selling and advertising products on the Web and exploits techniques from the fields of algorithmic game theory, mechanism design, and online learning. We study scenarios in which strategic agents, such as sellers, advertisers, and buyers, interact on Web platforms, and we analyze optimization problems faced by each party involved in the interaction. For instance, online marketplaces matching sellers/advertisers to buyers need to design mechanisms that incentivise agents to participate, while providing guarantees on their revenue. Taking the perspective of the online platform, we employ techniques and performance criteria from the mechanism design literature in order to design novel auction mechanisms and characterize their performance, with the goal of providing solutions for new e-commerce scenarios which emerged through recent advancements of digital advertising platforms. Moreover, we study how to address problems faced by agents interacting on the platforms, such as sellers and advertisers. In particular, when an agent has to sell and/or advertise their products on the Web, they have to repeatedly interact with the mechanism operated by the platform. The structure of such interaction is distributed over time: agents are required to perform sequential actions, after which they observe a reward produced by the environment that also depends on their decisions. In this setting, online learning techniques are well suited to design *no-regret algorithms* which allow agents to learn effective strategies while addressing the exploration/exploitation dilemma. Inspired by novel real-world scenarios, we study non-standard learning processes in which, for instance, the feedback returned by the environment is affected by delays, or

agents' actions are subject to time-varying constraints. These scenarios are common in practice when, for instance, agents are financially constrained by their budget or want to reach a target profitability in the form of a return-on-investment (ROI) constraint.

To conclude, this thesis extends classical models for online markets incorporating novel e-commerce frameworks that have emerged as a result of the continuous expansion of Web platforms. In doing so, it bridges the gap between theory and the latest real-world applications.

Contents

CHAPTER 1

Introduction

In the last decade, *artificial intelligence* has been one of the main drivers of growth for digital markets. For instance, the vast majority of companies employ digital tools to advertise their products or services, and the annual spent in digital advertising reached about 150 billion USD worldwide in 2020 (IAB, 2021). Furthermore, many economic reports suggest that artificial intelligence will contribute to an increase of market value by almost 100% over the next decade (Chui et al., 2018). Indeed, the use of AI tools in digital advertising has become increasingly common, opening up new opportunities that were previously unavailable. Some of the advantages over traditional advertising channels are the possibility of profiling a user from behavioral data (Devanur and Kakade, 2009), targeting ads in a precise way (Kempe and Mahdian, 2008), running auction mechanisms to maximize specific objective functions associated with the revenue (Mohri and Medina, 2014), and evaluating investment performance in real time. Although it is impossible to optimize these processes manually because of the vast amount of data provided by platforms and the numerous parameters that need to be set, algorithms and AI tools can efficiently perform such optimization.

In this thesis, we study new scenarios originating from recent innovations introduced by Web advertising and e-commerce platforms. Our particular

focus is on the development of new economic mechanisms and learning algorithms that leverage techniques from the fields of algorithmic game theory, mechanism design, and online learning which can be applied to sell and advertise products on the Web. The thesis is organized in three parts as follows.

• In the first part, we take the perspective of a seller who aims at selling their products or services on the Web. Most of the online economic transactions are carried out by posted-price mechanisms, in which sellers need to propose a *take-it-or-leave-it* price to each potential buyer. In this part of the thesis, we study the problem of setting prices over time in scenarios where a single item or multiple units of the same item have to be sold. We first analyze the scenario in which a single unit of a single item has to be sold within a finite period of time, when the value of the item is discounted over time according to an arbitrary continuous and non-increasing discount function. Our main result is a new posted-price mechanism, for which we provide guarantees in the form of bounds on the competitive ratio, that quantifies the worst-case difference in revenue between our mechanism and an optimal one that uses additional information about the user, typically unknown to the seller. Then, we analyze the scenario in which multiple units of the same item have to be sold. The solution we propose is a new no-regret algorithm that can effectively address the problem at hand, and can also be applied to recommendation problems. Specifically, our algorithm is well-suited to situations where rewards received from the environment are distributed over a time horizon, thereby bridging the gap between non-delayed and delayed scenarios in the existing literature.

• The second part of the thesis is centered around the problem of devising mechanisms for novel advertising scenarios. Our initial focus is on investigating a new type of ad auction that displays ads for similar products together with their respective prices. This can significantly influence user behavior and presents an opportunity for jointly optimizing ad allocation and pricing. To address this challenge, we propose several auction mechanisms differing in the payment rule and the level of information requested to the participants. Subsequently, we provide a study of their efficiency. Another new problem that we study is advertising in the metaverse. Specifically, we initiate the study of a user model and algorithms to allocate ads optimally in the metaverse. Our model extends those currently adopted for search and mobile advertising. In particular, we assume that, during their experience, users will traverse several scenes during which they could be targeted with multiple ads, whose performance may depend on the specific scene in which they are displayed. Furthermore, the ads may be subject to externalities

due to their sequential display. In this setting, we study the problem of computing an optimal allocation of ads. In particular, we assess the computational complexity of finding an optimal ad allocation for several model flavors and provide approximation algorithms with tight theoretical guarantees. Finally, we study advertising from the perspective of media agencies, whose recent proliferation has been driven by the increasing complexity of digital advertising. We extensively explore the effects of coordinating the bidding strategies of a group of advertisers who are participating in the same ad auction. Such coordination can lead to significant changes in the strategic interactions underlying the auction and may result in various forms of collusion, potentially increasing the revenue for the advertisers involved. We exploit the specific structure and features of the framework to provide approximate solutions for maximizing the revenue of the agency and the social welfare of the coordinated advertisers.

• The third part of this thesis studies the problem faced by a constrained agent that has to learn effective bidding strategies. For example, advertisers need to optimize their revenue while adhering to limitations such as budget constraints or a minimum profitability threshold expressed as a ROI constraint. Our main contributions are new no-regret algorithms that can tackle general problems in which a decision maker has to take sequential actions subject to uncertain and *long-term time-varying* constraints. In particular, we propose a *best-of-both-worlds algorithm*, with no-regret guarantees both in the case in which rewards and constraints are selected according to an unknown stochastic model, and in the case in which they are selected at each round by an adversary. Our framework can be instantiated to handle *full-feedback* as well as *bandit-feedback* settings. Finally, we show how it can be applied to constrained bidding in repeated first-price and second-price auctions, since they are de facto standard in large Internet advertising platforms.

In the following, we introduce the three parts composing this thesis and summarize the original contributions presented in each of them.

1.1 Algorithmic Pricing with Temporal Dependency

Selling on E-commerce Platforms via Posted Prices. In the first part of the thesis, we study problems revolving around buying and selling products on e-commerce platforms. In this setting, platforms act as intermediaries connecting sellers to buyers, and operate mechanisms that regulates the seller-buyer interaction. Therefore, platforms have to deal with the problem of finding the best way to orchestrate such interaction. Should a seller post a

fixed price, run an auction, or negotiate the deal with buyers? Although auctions were very popular in the early days of Internet commerce, today online sellers mostly adopt posted prices. Posted-price mechanisms try to sell an item by proposing a *take-it-or-leave-it* price to each arriving user, who then decides whether to buy the item or not (Chawla et al., 2010). If an agent opts for purchasing the item, then the mechanism terminates; otherwise, the agent leaves without any further possibility of buying the item, and the mechanism goes on by proposing prices to upcoming other agents. Over the last years, growing attention has been devoted to the analysis of posted-price mechanisms, both in the economic literature (Seifert, 2006) and in computer science (Babaioff et al., 2015, 2017; Adamczyk et al., 2017; Correa et al., 2017), within artificial intelligence and machine learning (Kleinberg and Leighton, 2003; Shah et al., 2019). Several noteworthy works recent economic literature investigates posted-price mechanisms in the presence of strategic buyers (Chen and Farias, 2018). For instance, Lobel (2020) studies dynamic pricing in the presence of patient consumers, who wait a certain period of time for a lower price and will purchase the item as soon as the price is equal to or below their valuation. Moreover, Golrezaei et al. (2020) provide an optimal mechanism for pricing goods in a setting where strategic buyers differ in both their initial valuations and the rates at which their initial valuation decreases with a delay in making the purchase. The motivation behind this focus on posted-price mechanisms is driven by the overwhelming number of online economic transactions carried out through posted-price mechanisms. This happens, for example, in online travel agencies (*e.g., Expedia*), accommodation websites (*e.g., Booking.com*), and e-commerce platforms (*e.g., Amazon, eBay*). As studied by Einav et al. (2018), an increasing number of *eBay* users prefer buying goods via posted prices rather than participating in auctions.

If a seller faces buyers with private information about their willingness to pay and there are no further transaction costs, there exist auction mechanisms which are proved to be optimal (see (Harris and Townsend, 1981; Myerson, 1981; Riley and Samuelson, 1981)). An auction aggregates information and helps the seller identify the appropriate buyer and price. However, posted-price mechanisms provide many advantages over auction-style mechanisms. From the designer's perspective, posting prices requires a much lower effort than running an auction, since it avoids the burden of first *eliciting information* (the bids) from the agents, and then *computing and collecting the payments*. At the same time, posted-price mechanisms retain most of the desirable properties of classical auctions, such as truthfulness. Indeed, even though the agents are not required to report their true valuations for the item,

they are always better off by deciding whether to buy the item or not on the basis of their true valuations, without acting strategically (Babaioff et al., 2017). Finally, if buyers appear gradually over time, are impatient, or are few in number, price posting may be preferable. From the agents' perspective, participating in a posted-price mechanism may also be preferable over competing in an auction. For instance, agents may prefer revealing minimal information about their true preferences if they plan to participate in similar markets in the future. Moreover, in some real-world settings, requiring the agents to figure out their true valuations for the item might need some additional efforts on their behalf, while answering a take-it-or-leave-it offer is usually much easier.

In Part I, we focus on two aspects of posted-price mechanisms, characterized by strong temporal components that must be considered in the analysis. In Chapter 3, we focus on selling a perishable item for which buyers have valuations decreasing over time. One of the main challenges of analysing the temporal structure of the problem is balancing the classical trade-off between setting high prices so as to achieve high revenue, and setting low prices so as to increase the probability of selling the item, while taking into account that the value of the item vanishes within a finite time horizon. Then, in Chapter 4, we study the setting in which the reward derived by selling products is partitioned over multiple time instant after the sale and its temporal structure is unknown to the seller. The proposed price may have delayed effects on the reward, and the seller has to learn which is the revenue-maximizing price. One of the fundamental challenge of such a temporal structure is whether it is possible for the seller to exploit incomplete reward samples to speed up the learning procedure.

We remark that the techniques employed to tackle the two problems presented in Part I are substantially different. This is due to the different objectives of the two chapters, which are selling a single unit of a single item, and selling multiple units of a single item, respectively. Selling many units of the same item in a sequential way allows the seller to employ learning techniques in order to find a good pricing strategy. The guarantees provided by such techniques ensure that the seller does not loose too much during the learning procedure. On the other hand, no learning is possible when selling a single unit of product. In the latter case, we provide a posted price mechanism and we evaluate it by performing a competitive analysis.

Selling a Sigle Unit of a Single Item. In Chapter 3, we study posted-price mechanisms for selling a single unit of a single item within a finite period of time, when the value of the item is discounted over time according to an arbitrary

continuous and non-increasing discount function. Discounting is common in many real-world applications and widely studied for a number of economic situations, such, *e.g.*, bargaining (Rubinstein, 1982; Gatti et al., 2008) and auctions (Mao et al., 2018). We tackle settings in which agents arrive sequentially—a common assumptions in *online* mechanism design (Lavi and Nisan, 2004; Parkes, 2007)—and the number of agents is unknown *a priori*. In particular, following a mainstream approach in economics (see, *e.g.*, (Mason and Välimäki, 2011; Rosenthal, 2011)), we assume that agents' arrivals are governed by a Poisson process. Remarkably, posted pricing with Poisson arrivals has been previously investigated by Wang (1993) and Rong et al. (2018) for undiscounted settings, though without providing any theoretical result.

We assume that each agent arriving at the mechanism has a different initial (*i.e.*, undiscounted) valuation for the item, which is independently drawn according to a common probability distribution. This leads to a fundamental trade-off between setting high prices so as to achieve high revenue and, on the other side, progressively lowering posted prices so as to increase the probability of selling the item. Our assumption is that the mechanism is only aware of the range of valuations, while it does not know anything about the shape of the distribution. This is reasonable since, differently from the actual distribution, the range of valuations can be estimated from previous data or market surveys.

Lavi and Nisan (2004) and Babaioff et al. (2017) provide the main state-of-the-art results on posted-price mechanisms for single-item single-unit scenarios. However, their models do not fit to our setting, since the agents' valuations are *not* discounted over time and the number of agents is known *a priori*. As a result, these models do not embed an explicit time representation and the proposed pricing strategies are only driven by the number of agents arrived.

Our model encompasses many real-world scenarios, such as, *e.g.*, long-term rental of rooms and apartments. Think of a website renting rooms to students for fixed periods of one year. The value of a room naturally decreases over time, reflecting the fact that a future tenant will benefit from the room for a period shorter than one year. Moreover, the potential customers arrive at the renting website according to a stochastic process, which can be reasonably modeled by a Poisson process whose rate parameter can be easily estimated by looking at traffic logs of the website.

Original Contributions - Selling a Single Unit of a Single Item. We adopt the perspective of competitive analysis (Borodin and El-Yaniv, 2005) and evaluate

our mechanisms in terms of *competitive ratio*, measuring the worst-case ratio between their revenue and that of an optimal mechanism that knows the distribution of valuations. As it is customary in the literature (see, *e.g.*, (Babaioff et al., 2017; Kleinberg and Leighton, 2003)), we first focus on the identical valuation setting in which all the agents share the same initial valuation for the item. Then, we extend our results to the random valuation setting where the agents' valuations are drawn i.i.d. from the same distribution satisfying the monotone hazard rate condition (when the distributions of valuations are unrestricted, Lavi and Nisan (2004) and Babaioff et al. (2017) show that then there is no algorithm with good performances). In the identical valuation setting, we design a posted-price mechanism $\mathcal{M}_{\text{C}}$ and prove that it is optimal, *i.e.*, it provides the best possible competitive ratio. In order to derive the ratio, we first identify two crucial properties that characterize optimal mechanisms: their undiscounted price is non-increasing in time and they always guarantee the same fraction of the expected revenue of an optimal mechanism that knows the agents' valuation, independently of its actual value. For the specific case of linear discount, we discuss how the competitive ratio depends on the parameters. In the random valuation setting, we first show that mechanism $\mathcal{M}_{\text{C}}$ still provides good performances by proving that its competitive ratio is lower bounded by a constant, which does not depend on the distribution of agents' valuations. Then, motivated by real-world scenarios in which the seller is constrained not to change the posted prices too often, we propose a new mechanism $\mathcal{M}_{\text{PC}}$ defined by a piecewise constant pricing strategy and prove that its performances in terms of competitive ratio are comparable with those obtained by $\mathcal{M}_{\text{C}}$. In conclusion, we empirically compare $\mathcal{M}_{\text{C}}$ with a natural adaption of the mechanism proposed by Babaioff et al. (2017) to our setting, showing that the latter is inefficient even without time discounting. We also empirically evaluate the performances of $\mathcal{M}_{\text{C}}$ and $\mathcal{M}_{\text{PC}}$ as the frequency with which prices are allowed to change decreases, showing that, when this is not too low, then the performances of $\mathcal{M}_{\text{PC}}$ and $\mathcal{M}_{\text{C}}$ are comparable.

Selling multiple units of a single item. When multiple units of the same item are available, learning approaches based on *bandit* techniques are customarily adopted. In particular, Kleinberg and Leighton (2003) study an unlimited-supply setting where the number of buyers is fixed, and derive upper bounds on the regret. Several recent works extend the results in Kleinberg and Leighton (2003). Shah et al. (2019) study a contextual setting, providing a semi-parametric model that learns from the observation of a binary outcome which stands for acceptance or rejection of the offered price. Mohri and

Munoz (2014) study revenue-maximizing learning algorithms for posted pricing with strategic buyers. In Chapter 4, we also provide bandit techniques to tackle the problem of sequentially selling multiple units of the same item when the reward has a specific temporal structure.

Sequential decision-making occurs in many real-world scenarios such as clinical trials, recommender systems, Web advertising, and e-commerce. Inspired by these applications, many different flavours of the multi-armed bandit (MAB) setting have been investigated. A crucial role is played by the time the reward is observed. In many cases, the reward is subject to a *delay*, and such a delay, if not sufficiently short, can prevent the design of algorithms that are effective in practice. Online learning with delayed feedback has received considerable attention in recent years, and several results are available in the literature, *e.g.*, see the seminal work by Joulani et al. (2013). A major distinction in MABs with delayed feedback concerns the nature of the rewards, which may be stochastic (Mandel et al., 2015; Cella and Cesa-Bianchi, 2020) or adversarial (Bistritz et al., 2019; Thune et al., 2019; van der Hoeven and Cesa-Bianchi, 2021).

Our work focuses on a special class of bandit problems with stochastic and delayed rewards, in which we can get partial feedback over time. More precisely, we study a novel setting, namely MAB with Temporally-Partitioned Rewards (TP-MAB), in which the reward associated with an action, a.k.a. *arm*, chosen at a given round is collected during a finite number of rounds following the choice, according to an unknown probability distribution. In classical delayed-feedback bandits (see, *e.g.*, (Joulani et al., 2013)), the reward is concentrated in a single round that is (stochastically) delayed w.r.t. the round in which the learner pulled the corresponding arm. TP-MABs naturally extend this setting by allowing the reward to be partitioned into multiple elements that are collected with different delays. We call arm's *per-round reward* the partial reward observed by the learner in a single round, which is assumed to be the realization of a random variable with an unknown probability distribution. We call arm's *cumulative reward* the random variable given by the sum of all the per-round rewards obtained by pulling an arm. While the per-round reward can be observed round by round, the cumulative reward is revealed only at the end. Notice that, in a single round, the learner observes a per-round reward for each previously pulled arm whose cumulative reward is not terminated yet. Our goal is to find a policy to maximize the cumulative reward, exploiting the per-round rewards as intermediate signals on the arm performance.

The TP-MAB framework is general and captures many of real-world scenarios, among which, the problem of pricing multiple units of the same

product presenting the temporally-partitioned reward structure. An example is the problem of pricing subscription-based music, podcast or video streaming services. In this setting, each arm corresponds to the monthly price and the reward is proportional to the number of times the user pays that price for benefiting from the service. The goal is to find the price maximizing the reward over a time horizon. Suppose that a platform providing such service has the possibility of proposing a standand or a discounted monthly price to new users for a fixed time period. The standard price may be a good short-term choice, however, it may discourage the user to renew the subscription several times. The opposite reasoning holds for the discounted price. At each round, the platform propose a price to a new user, whose appreciation of the service is revealed through multiple steps. In particular, every partial observation and the related reward correspond to the payment of a monthly subscription. If the user, does not purchase the subscription for a month, the corresponding reward is non-positive. The cumulative reward provided by setting a monthly price corresponds to the sum of monthy subscription paid over the prefixed time horizon. In the classical delayed-feedback bandit setting, the feedback on the proposed price is obtained only at the end of the time horizon. However, in this setting, the platform is able to monitor each month if the subscription is paid. This provides us useful hints on the performance of the chosen arm before the end of the time horizon. In Chapter 4 we provide a further discussion on other real-world applications of the TP-MAB framework.

Original Contributions - Selling multiple units of a single item. Initially, we focus on the lower bound of TP-MABs, showing that the TP-MAB setting has the same regret lower bound of the standard delayed MAB setting when there is no further assumption about how the rewards are partitioned over time. Since in many practical applications of interest the cumulative reward of each arm does not concentrate excessively in a short sub-range of rounds, we introduce a property describing how the maximum per-round reward distributes. We call this property α-*smoothness* where $\alpha \geq 1$. In particular, the minimum value of $\alpha = 1$ corresponds to the case in which there is no structure and, therefore, the maximum per-round reward can be the entire cumulative reward. On the other hand, the maximum value of α is equal to the maximum delay and corresponds to the case in which the cumulative reward distributes evenly over time. Thus, the maximum per-round reward decreases as the value of α increases. We show that the lower bound of this setting is of a factor $1/\alpha$ smaller than that when α-smoothness does not hold. Then, we design two novel algorithms, namely `TP-UCB-FR` and

TP−UCB−EW, suited for the TP-MAB setting, which exploit partial feedback and the α-smoothness property. We show that the regret of TP−UCB−FR is $\mathcal{O}(\ln T/\alpha)$, where T is the time horizon of the learning process, and the regret of TP−UCB−EW is $\mathcal{O}(\ln T)$. A comprehensive analysis the regret bounds of our and state-of-the-art algorithms in various settings can be found in Tables 4.1, 4.2, 4.3. Finally, we experimentally show that our algorithms outperform the state of the art over synthetically generated and a real-world playlist recommendation scenario.

1.2 Algorithmic Advertising: Price Displaying, Collusion and the Metaverse

In the second part of the thesis, we study recently-emerged advertising settings originated through the advancements of digital advertising platforms. In particular, AI will allow the optimization of more and more intricate economic settings, in which multiple different activities need to be jointly automated. This is the case of, *e.g.*, *Google Hotel Ads* and *Tripadvisor*, where auctions are used to display ads of similar products or services together with their prices. Consider, for instance, users who search for the availability of an hotel room in a given date. The Web page of results shows a ranking of banners advertising similar hotel rooms that match the search criteria. Each banner displays the name of the advertiser providing the online booking service, together with the per-night selling price of the room. Such settings are similar to standard ad auctions, since the ads are ranked depending on the advertisers' bids. On the other hand, they also fundamentally differ from standard ad auctions, as the ad allocation must also take prices into account, and these are displayed inside the banners so as to provide a direct comparison among them. This dramatically affects users' behavior, as well as the efficiency and the properties of the mechanism. In Chapter 5, we investigate how the additional degree of freedom introduced by the possibility of choosing prices influences the problem of finding an optimal ad allocation and the revenue of the mechanisms.

Recent years have witnessed another noteworthy phenomenon, which is the proliferation of *media agencies*. Media agencies claim to play the role of intermediaries between *advertisers* and *platforms* selling ad slots. This trend has been driven by the increasing complexity of digital advertising—due to, *e.g.*, a large amount of available data and of parameters to be set on advertising platforms—and the rising competition among a growing number of advertisers. When a group of competing advertisers is managed by a common agency, it frequently happens that the agency has to place bids on

behalf of different advertisers participating in the same ad auction. This dramatically changes the strategic interaction underlying ad auctions, since agencies can coordinate advertisers' bidding strategies by implementing many forms of *collusion—e.g.*, bid rigging—, with the goal of increasing advertisers' performance. In particular, simple examples show that colluding in ad auctions can reward the colluders with a utility that is arbitrarily larger than what they would get without doing that. Moreover, a recent empirical study on real-world data by one of the major US agencies (Aegis-Dentsu-Merkle) shows that collusion is pervasive and leads to a significant reduction in the average cost-per-click (Decarolis and Rovigatti, 2017).

Finally, we study advertising problems on *metaverse* platforms, which will revolutionize advertising in the next decade (Taylor, 2022). In the metaverse, users are offered a real-time immersive experience which enables several new marketing opportunities unseen before. One of the central questions today concerns *which ads* to display to the users, and at *which time* of their experience.

In what follows, we present our contribution to the study of these novel advertising settings.

Online Advertising with Price Displaying - Original Contributions. The price-displaying feature introduces *externalities among the ads*, since the probability that a user clicks on an ad depends on the prices displayed with both the ad being clicked and the other ads in the allocation. Several forms of externalities are investigated in the literature on ad auctions. However, to the best of our knowledge, no previous work takes into account price displaying in ad auctions. For instance, Kempe and Mahdian (2008) and Aggarwal et al. (2008) introduce a basic user model that is currently adopted by most of the mechanisms. In this model, a Markovian user observes the slots in a top-down fashion, moving down slot by slot with a given continuation probability and stopping on a slot to observe the corresponding ad with the remaining probability. Kempe and Mahdian (2008) propose models where the probability with which a user moves from a slot to the next one depends on the ad actually displayed in the former. In this case, it is *not* known whether the ad allocation problem admits a polynomial-time algorithm; however, Farina and Gatti (2016, 2017) provide several algorithms showing that in special cases a constant approximation can be achieved. Further externalities models are explored by Fotakis et al. (2011) and Gatti et al. (2018). However, when these models are adopted, the ad allocation problem is NP-hard and, in some cases, even inapproximable.

In our model, we assume that the probability with which a user clicks

on an ad depends on the price displayed with the ad *and* on the lowest among all displayed prices. In particular, we model the click probability as a monotonically decreasing function of the ad price, assuming that the demand curve is monotonically decreasing in the price and that it is unlikely that a user clicks on an ad with a price larger than their reserve value. We also assume that the click probability is monotonically decreasing in the difference between the ad price and the lowest displayed price, as the user's interest in any feature different from price (*e.g.*, brand and loyalty) decreases as such difference increases.

In our setting, the private information of each advertiser (*i.e.*, their type) is a pair composed by the probability with which a user visiting the advertiser's Web page produces a conversion (*e.g.*, a purchase) and the advertiser's cost for a unit of product or service. On the other hand, the prices constitute an additional degree of freedom that can be controlled by either the advertisers or the mechanism.

As a first step, we present a direct-revelation mechanism that maximizes the social welfare by jointly optimizing over the ad allocations and the prices displayed with the ads. Differently from what happens in most of the externalities models studied in the literature, such optimization problem can be solved in polynomial time for a given discretization of price values. We also study the properties of the direct-revelation mechanism when VCG payments are used, showing that incentive compatibility, individual rationality and weak budget-balance hold in our setting.

In real-world scenarios, it is unlikely that the advertisers let the mechanism select prices on their behalf, as required by the direct-revelation mechanism. In the (indirect-revelation) mechanisms that are currently adopted in real-world applications, the optimization over ad allocations and that over prices are decoupled. In particular, each advertiser finds their optimal price and bid, while the mechanism optimizes over ad allocations once prices and bids are given. As for the direct-revelation mechanism, the best ad allocation can be found in polynomial time given prices and bids. Indirect-revelation mechanisms allow the advertisers not to reveal private (and potentially sensitive) information, however, they can lead to inefficient equilibria.

We investigate the equilibrium inefficiency of indirect-revelation mechanisms with GSP and VCG payments, in terms of *Price of Anarchy* (PoA) and *Price of Stability* (PoS) in complete information settings. In the literature, PoA and PoS are commonly-adopted efficiency metrics for standard ad auctions, in which the price variable is not taken into account. For instance, Paes Leme and Tardos (2010), Caragiannis et al. (2011), Lucier and Leme (2011), and Caragiannis et al. (2015) show that the PoA for the social wel-

fare of the GSP is upper bounded by 1.3 with complete information and by 3 with incomplete information, while Farina and Gatti (2017) and Giotis and Karlin (2008) study the inefficiency with specific externalities. In our setting, the presence of externalities precludes the adoption of the tools provided by Roughgarden et al. (2017) and Hartline et al. (2014) to bound the inefficiency of equilibria for the social welfare and the revenue, respectively, thus pushing us to develop *ad hoc* approaches. In particular, we show that, in our setting, the inefficiency of the indirect-revelation mechanisms with VCG and GSP payments is much higher than that of the classical mechanisms without prices, even when excluding overbidding, since the PoS for the revenue may be unbounded even with two slots and the PoA for the social welfare may be as large as the number of slots. Furthermore, with VCG payments, the PoS for the social welfare is 1, while, with GSP payments, it is at least 2, suggesting that indirect-revelation mechanisms with GSP payments perform worse than those with VCG ones.

A crucial question is whether inefficiency can be reduced when letting the advertisers choose their prices. We show that, under some assumptions, simple modifications to the indirect-revelation mechanism with VCG payments—requiring each advertiser to report an additional price—achieve a PoS of 1 for the revenue.

Media Agencies Coordinating Bidders - Original Contributions. We study the computational problem faced by a media agency that has to coordinate the bidding strategies of a group of colluders, under GSP and VCG mechanisms. We assume that the media agency knows the private valuations of the colluders (*i.e.*, how much they value a click on their ad), and that it decides colluders' bids on their behalf. Moreover, the media agency is in charge of paying the auction mechanism for a click on an allocated ad, and at the same time it requires *monetary transfers* to and from the colluders. These are necessary to enforce some *individual rationality constraints* ensuring that the colluders do *not* leave the agency and participate in the ad auction individually. In this chapter, we study two settings that differ for the kind of monetary transfers that they allow for: the *arbitrary transfers* setting, where any kind of monetary transfer to and from the advertisers is allowed, and the more realistic *limited liability* setting, in which no advertiser can be paid by the media agency. Finally, we assume that the bids of the advertisers external to the media agency are drawn according to some probability distribution. As a first result, we introduce an abstract bid optimization problem, called *weighted utility problem* (WUP), which works for any finite set of possible bid values and is useful in proving our main results in the rest of the chapter.

In order to solve such a problem, we first show that the utilities of bidding strategies are related to the length of paths in a directed acyclic weighted graph, whose structure and weights depend on the mechanism under study (either GSP or VCG mechanism). This allows us to solve WUP instances in polynomial time by finding a shortest path of the graph. Next, we switch the attention to the original media agency problem, starting from the arbitrary transfers setting. A major challenge is dealing with a potentially continuous set of possible bids. Notably, we show that it is possible to reduce the attention to a finite set of bidding strategies, only incurring in a small additive loss in the value of the obtained solution and relaxing the incentive constraints by a small additive amount. The set is built by recursively splitting the interval of possible bid values, until one gets sub-intervals such that the probability that an external bid is in a given sub-interval is sufficiently small. Then, the resulting sub-intervals are used to define the desired finite set of bids. In conclusion, we cast the problem as a WUP instance and solve it by our graph-based algorithm in polynomial time. This gives a bi-criteria additive FPTAS for the original problem, since the (additive) approximation is in terms of both objective value and incentive constraints. Finally, we study the limited liability setting. In this case, we leverage the same finite set of bidding strategies defined for the arbitrary transfers setting in order to formulate the problem as a *linear program* (LP) with exponentially-many variables and polynomially-many constraints. Since we use only a finite set of bids, we need to relax the individual rationality constraints by an arbitrary small amount to guarantee the existence of a feasible solution. We solve such an LP in polynomial time by applying the ellipsoid algorithm (Grötschel et al., 1981) to its dual, which features polynomially-many variables and exponentially-many constraints. This requires solving a suitable separation problem in polynomial time, which can be done by reducing it to a WUP instance. As in the arbitrary transfer setting, the resulting algorithm is a bi-criteria additive FPTAS.

While a longstanding literature investigates the role of mediators in ad auctions—see, *e.g.*, the seminal works by Vorobeychik and Reeves (2008) and Ashlagi et al. (2009)—, collusion is currently emerging as one of the central problems in advertising, as the adoption of AI algorithms can concretely support an agency to find the best collusive behaviors (OECD, 2017). Motivated by the recent study by Decarolis and Rovigatti (2017), some works provide theoretical contributions to assess how collusion can be conducted by an agency. Decarolis et al. (2020) study a setting in which there is no monetary transfer between the agency and bidders by providing equilibrium conditions. They show that, in simple settings, GSP is more

inefficient than VCG, both in terms of efficiency and revenue. Lorenzon (2018) studies a setting with two slots and three bidders that are all controlled by an agency in a GSP auction. Furthermore, a monetary transfer is possible. The author shows collusive stable behaviors in which the redistribution is uniform over the three bidders. Collusion has also been studied in the context of sequential games where a team of agents has to coordinate (*i.e.*, collude) against an adversary (Farina et al., 2018; Celli and Gatti, 2018; Basilico et al., 2017).

Advertising in the Metaverse - Original Contributions. To the best of our knowledge, the question of which ads to display and at what point in the user's experience is unexplored in the metaverse, whereas several works investigate it in the case of the Web. In particular, the design of an attention model describing how the user observes the slots in which ads are displayed is crucial. Indeed, a user model needs to address the tradeoff between, on one side, a sufficiently accurate description of the user behavior and, on the other side, the possibility to design allocation algorithms running in polynomial time to scale up to concrete applications. The seminal model, called *cascade* and proposed by Kempe and Mahdian (2008), assumes that users observe the slots sequentially. The authors also propose some algorithms for special cases, while Farina and Gatti (2017) provide algorithms for the general case. Fotakis et al. (2011) and Gatti et al. (2018) propose detailed models incorporating negative externalities between ads. These models do not admit constant approximation algorithms. We also mention that Gatti et al. (2014) adopt a similar approach in the case of mobile advertising.

We initiate the study of a user model and algorithms to allocate ads optimally in the metaverse. Our model extends those currently adopted for search and mobile advertising. In particular, we assume that, during their experience, users will traverse several scenes (*e.g.*, sports events, concerts, job meetings, tourist sites, lectures, and conferences) during which they could be targeted with multiple ads of different formats, whose performance (usually, referred to as *quality*) may depend on the specific scene in which they are displayed (*e.g.*, an ad may attract the user attention differently if shown in a sports event or a concert). Furthermore, the ads may be subject to externalities due to their sequential display. More precisely, displaying an ad in a scene may raise negative forward externalities to other ads shown in future scenes (*e.g.*, when two ads related to products that are strategic substitutes are displayed sequentially, as shown by Deng and Pekec (2011)). However, it is unlikely that a user recalls every ad seen in the past. Thus, we assume that the users' behavior is affected only by the ads displayed

in the last k scenes, where k is a finite number. We also allow an ad to be displayed multiple times, as it is common in real-world scenarios to induce awareness effects.

In our model, an allocation of ads specifies which ad to display in each scene, where the possible scenes a user can traverse are connected according to a tree structure rooted by the scene in which the user is initially. Let us notice that, in principle, the optimal allocation may prescribe not to allocate any ad in some scenes so as to exclude externalities to other ads allocated in future scenes. Indeed, the externalities due to some ads to those displayed in the following scenes may reduce the latter's values so much that it would be more advantageous not to display them at all. We also study the problem of computing an optimal allocation of ads. In particular, we assess the computational complexity of finding an optimal ad allocation for several model flavors and provide approximation algorithms with tight theoretical guarantees. Interestingly, allowing the ads to have different qualities in different scenes makes the problem APX-Complete, and we provide a polynomial-time algorithm with an approximation factor of $(1 - 1/e)$. Instead, introducing externalities among the ads makes the problem Poly-APX-Complete, and we provide a polynomial-time algorithm with an approximation factor of $1/(k + 1)$, which is tight and shows that the problem is in APX when k is fixed. Similar upper and lower complexity bounds hold when adopting the model in the general case. Interestingly, we show that our algorithms provide approximations arbitrarily better than allocation algorithms disregarding basic user features in the metaverse. Furthermore, our algorithms are greedy with a running time compatible with real-world applications. We also discuss under which conditions our approximation algorithms are *weakly monotone* in the sense of Myerson, thus leading to truthful auction mechanisms. In particular, we show that, when the qualities are scene-dependent, our algorithms are not weakly monotone in the sense of Myerson.

1.3 Learning with Constraints in Digital Advertising

In the third part of this thesis, we study online learning problems concerning ad auctions with constrained decision makers. In the classical online learning framework an agent repeatedly interacts with the environment. At each round the agent selects an action and the environment returns an outcome. This interaction can be stochastic or adversarial. In the first case, the outcome is stochastically selected according to a distribution unknown to the decision maker, while, in the latter, an adversary chooses it. We focus on set-

tings where actions are subject to a set of time-varying constraints. Agents are allowed to make decisions that are not feasible, provided that, across a fixed time horizon, the cumulated violations of the overall sequence of decisions are bounded with respect to a *safety* measure. Different notions of safety may be employed to define a *safe* process. For instance, in Chapter 8 the decision maker is allowed to violate the constraints only with a small probability during the learning process, while in Chapter 9 we look for a cumulative constraints violation which has to be sublinear in the time horizon. At the same time, guarantees on the reward cumulated over the learning process are required. The problem becomes that of finding a sequence of decisions which guarantees a reward close to that of the best fixed decision in hindsight while satisfying the constraints. The main challenge addressed in Part III is dealing with general (*i.e.*, not necessary *packing*) time-varying constraints. There are many real-world scenarios in which providing guarantees with respect to general constraints is crucial. One example is ad auctions with financially constrained bidders. Bidders repeatedly interact with an auction mechanism and aim at learning revenue-maximizing bidding strategies while satisfying constraints relating to their financial resources, such as *budget* and *return on investment* (ROI) constraints (Auerbach et al. (2008); Golrezaei et al. (2021a); Li et al. (2020)). Even if traditional budget-pacing mechanisms are suited for settings that involve only resource consumption constraints (e.g., budget), it is shown that in real-world scenarios advertisers take into account covering constraints (e.g., ROI) (Golrezaei et al., 2021b). This reflects their willingness to achieve a tradeoff between high volumes and high profitability. In what follows, we propose two different perspective through which we provide guarantees during the learning process under time-varying constraints. First, we address the problem in the specific setting of ad auctions with budget and ROI constraints, in which the uncertainty of constraints and the reward depends on to the stochasticity of the environment. Then, we introduce a general framework with long-term time-varying constraints encompassing both adversarial and stochastic settings.

ROI and Budget Stochastic Constraints - Original Contributions In Web advertising, advertisers' usually set *bids* so as to balance the tradeoff between achieving *high volumes*, corresponding to maximizing the sales of the products to advertise, and *high profitability*, corresponding to maximizing ROI. Companies' business units need simple ways to address this tradeoff, and, customarily, they maximize the volumes while constraining ROI to be above a given threshold. The importance of ROI constraints, in addition to standard budget constraints, is remarked by several empirical studies. We mention,

e.g., the data analysis on the auctions performed on Google's AdX by Golrezaei et al. (2021b), showing that many advertisers take into account ROI constraints, particularly in hotel booking. However, no platform provides features which guarantee ROI constraints to be satisfied, and some platforms (*e.g.*, TripAdvisor and Trivago) do not even allow the setting of daily budget constraints. Thus, the problem of satisfying these constraints is a challenge that the advertisers need to address by designing suitable bidding strategies. In this picture, uncertainty plays a crucial role as the revenue and cost of the advertising campaigns are unknown beforehand and need to be estimated online by learning algorithms during the sequential arrival of data. As a result, the constraints are subject to uncertainty, and wrong estimations of the parameters can make the ROI and budget constraints be arbitrarily violated. Such violations represent today the major obstacles to adopting AI tools in real-world applications as often considered unacceptable risks by the advertisers. In particular, this issue is crucial in the early stages of the learning process as adopting algorithms with an uncontrolled exploration when a small amount of data is available can make the advertising campaigns' performance oscillate with a large magnitude. Therefore, controlling the exploration in order to mitigate risks and provide safety guarantees during the entire learning process is of paramount importance.

As customary in the online advertising literature (see, *e.g.*, (Devanur and Kakade, 2009)), we make the assumption of stochastic (*i.e.*, non-adversarial) clicks, and we adopt Gaussian Processes (GPs) to model the problem. In particular, our model combines an optimization and a learning problem. Initially, we focus on studying our optimization problem without uncertainty, showing that no approximation within any strictly positive factor is possible with ROI and budget constraints unless $\mathsf{P} = \mathsf{NP}$, even in simple, realistic instances. However, when dealing with a discretized space of the bids as it happens in practice, the problem admits an exact pseudo-polynomial time algorithm based on dynamic programming. Most importantly, when the problem is with uncertainty, we show that no online learning algorithm can violate the ROI and/or budget constraints a sublinear number of times while guaranteeing a sublinear pseudo-regret. Notably, this result holds in general bandit settings beyond advertising when the constraints are subject to uncertainty and the arm space or constraints are not convex. We provide an algorithm, namely GCB, providing pseudo-regret sublinear in the time horizon T at the cost of a linear number of violations of the constraints. We also provide its safe version, namely $\mathsf{GCB}_{\texttt{safe}}$, guaranteeing w.h.p. a constant upper bound on the number of constraints' violations at the cost of a regret linear in T. Inspired by the two previous algorithms, we design

a new algorithm, namely $\mathsf{GCB}_{\mathtt{safe}}(\psi, \phi)$, guaranteeing both the violation w.h.p. of the constraints for a constant number of times and a pseudo-regret sublinear both in T and the maximum information gain of the GP when accepting tolerances ψ and ϕ in the satisfaction of the ROI and budget constraints, respectively. We experimentally compare our algorithms in terms of pseudo-regret/constraint-violation tradeoff in settings generated from real-world data, showing the importance of adopting safety constraints in practice and the effectiveness of our algorithms. In particular, using small tolerances ψ and ϕ with $\mathsf{GCB}_{\mathtt{safe}}(\psi, \phi)$ guarantees very smooth dynamics and a negligible loss in reward.

General Time-varying Constraints - Original Contributions We study online learning problems where a decision maker takes decisions over T rounds. At each round t, the decision $\boldsymbol{x}_t \in \mathcal{X}$ is chosen before observing a reward function f_t together with a set of m *time-varying* constraint functions g_t. The decision maker is allowed to make decisions that are *not* feasible, provided that the overall sequence of decisions obeys the *long-term constraints* $\sum_{t=1}^{T} g_t(\boldsymbol{x}_t) \leq \mathbf{0}$, up to a small cumulative violation across the T rounds. The problem becomes that of finding a sequence of decisions $\boldsymbol{x}_t$ which guarantees a reward close to that of the best fixed decision in hindsight while satisfying long-term constraints. This type of framework was first proposed by Mannor et al. (2009), and it has numerous applications ranging from wireless communication (Mannor et al., 2009) and multi-objective online classification (Bernstein et al., 2010), to *safe* online learning (Amodei et al., 2016).

Mannor et al. (2009) show that guaranteeing sublinear regret and sublinear cumulative constraints violation is impossible even when f_t and g_t are simple linear functions. Therefore, previous works either focus on the case in which constraints are generated i.i.d. according to some unknown stochastic model, without providing any guarantees for the adversarial case, or provide results for adversarially-generated constraints under some strong assumptions on the structure of the problem or using a weaker baseline (a detailed discussion of related works can be found in Chapter 9. A few examples in the latter case are Sun et al. (2017); Yi et al. (2020); Chen et al. (2017); Cao and Liu (2018). In the former setting (*i.e.*, stochastic constraints), Wei et al. (2020) consider a weaker baseline that is feasible for each constraint g_t, going against the basic idea of long-term constraints. A notable exception is the work by Yu et al. (2017), who employ the same baseline as ours, and provide an upper bound of $\tilde{O}(T^{1/2})$ for both regret and constraints violation (see Table 9.1). We also mention that there are some works studying the problem

in which constraints are *static* (see, *e.g.*, Jenatton et al. (2016); Mahdavi et al. (2012); Yu and Neely (2020); Yuan and Lamperski (2018)), or focus on specific types of constraints, such as *knapsack constraints* Badanidiyuru et al. (2018); Immorlica et al. (2019). Our framework differs from those works as we deal with *arbitrary* and *time-varying* constraints. Moreover, it also extends the *online convex optimization* framework introduced by Zinkevich (2003) by allowing for general non-convex loss functions f_t, arbitrary feasibility sets $\mathcal{X}$, and arbitrary time-varying long-term constraints.

Given the negative result by Mannor et al. (2009), a natural question is what kind of guarantees we can reach in the adversarial setting, when adopting the standard baseline of the best fixed decision in hindsight satisfying (in expectation) the long-term constraints. We provide the first positive result going in this direction, by designing a no-α-regret algorithm that guarantees a sublinear cumulative constraints violation. Moreover, we make a step forward in the line of work initiated by Bubeck and Slivkins (2012), by showing that our algorithm is also the first *best-of-both-worlds* algorithm for problems with arbitrary long-term constraints. This allows our algorithm to guarantee good worst-case performance (adversarial case), while being able to exploit well-behaved problem instances (stochastic case). The only assumption which we require is the existence of a decision that is strictly feasible with respect to the sequence of constraints. We denote by ρ the "margin" by which this decision is strictly feasible (see Section 9.1 for a definition). At the same time, we show that even without this assumption, we can recover sublinear regret and violation with stochastic constraints.

Previous work usually assumes that ρ is a given *constant*. In that case, our algorithm matches the guarantees by Yu et al. (2017) when constraints are generated i.i.d. according to an unknown distribution, and has no-α-regret with $\alpha = \rho/(1+\rho)$ in the adversarial case (see Table 9.1). Our algorithm only requires a lower bound on the real value of the feasibility parameter ρ. In the stochastic case, the lower bound may even be unknown, and the algorithm can efficiently estimate it from data. Moreover, we argue that if ρ is allowed to depend on T and take arbitrarily small values, then there are certain values ($\rho \leq T^{-1/4}$), for which any regret bound depending on $1/\rho$ would be useless (*i.e.*, *not* sublinear in T, see Section 9.2). This setting is usually overlooked by previous work, which assumes ρ to be a given constant. We show that, in the case of an arbitrary feasibility parameter ρ, in the stochastic setting our algorithm guarantees an upper bound of $\tilde{O}(T^{3/4})$ for regret and cumulative constraints violation.

Our framework employs traditional regret minimizers as black-box components. Therefore, by instantiating it with an appropriate choice of regret

minimizers it can handle *full-feedback* as well as *bandit-feedback* settings. In the former case, after playing $\boldsymbol{x}_t$, the decision maker gets to observe f_t and g_t, while in the latter case only the realized values $f_t(\boldsymbol{x}_t)$ and $g_t(\boldsymbol{x}_t)$ are observed. Moreover, employing a suitable regret minimizer for non-convex losses allows the decision maker to seamlessly handle scenarios with non-convex reward and constraints (see, *e.g.*, Suggala and Netrapalli (2020)). Our algorithm is based on a two-stage approach in which *primal* and *dual* players interact through *Lagrangian games*. In the first (*play*) phase, the primal player tries to balance out the maximization of their rewards with constraints violation. In the second (*recovery*) phase, the primal player only makes "safe decisions" to avoid violating constraints too much. It is possible to prove that, in the case of stochastic rewards and constraints, the algorithm never enters phase two. This property is particularly relevant for budget-pacing mechanisms in repeated auctions, since it is related to how budget is allocated. Our framework can also be instantiated to perform budget allocation subject to constraints that were previously *not* tractable by traditional mechanisms, such as ROI constraints Balseiro and Gur (2019); Conitzer et al. (2021).

1.4 Structure of the Thesis

In this section, we describe the structure of the thesis. In Chapter 2, we introduce some fundamental concepts related to game theory, mechanism design and online learning. In particular, we define games and equilibria, we introduce general economic mechanisms, we summarize the most popular auction mechanisms, and describe the ad auction framework. Finally, we provide some notions on the online learning framework and multi-armed bandit problems.

Part I of the thesis focuses on pricing problems where the time component plays a fundamental role. In particular, we study mechanisms and strategies for selling a single unit or multiple units of an item/service in non-standard scenarios. Our contributions are organized as follows:

- Chapter 3 characterizes a posted-price mechanism to sell a single unit of a single item within a finite period of time. Buyers arrive online and their valuations are drawn from an unknown distribution and discounted over time.

- Chapter 4 studies a novel bandit setting, namely Multi-Armed Bandit with Temporally-Partitioned Rewards (TP-MAB), which may be applied to several real-world scenarios, such as the pricing of products or

services providing a revenue distributed over a time span following the purchase.

The results of Part I were published as Romano et al. (2021) at AAAI-2021, and Romano et al. (2022a) at IJCAI-2022.

Part II of the thesis focuses on advertising mechanisms and strategies, studied from the perspective of advertising platforms and advertisers, respectively. We focus on recent settings characterized by specific features which we exploit to provide solutions well suited for each scenario.

- Chapter 5 studies a novel type of auction in which ads of similar products or services are displayed together with their prices.
- Chapter 6 studies the computational problem faced by a media agency that has to coordinate the bids of a group of colluding agents, under GSP and VCG mechanisms.
- Chapter 7 initiates the study of advertising on the metaverse platform, providing a user model and algorithms to optimally allocate ads in this brand new setting.

The results of Part II were published as Castiglioni et al. (2022c) at AAAI-2022, Romano et al. (2022b) at IJCAI-2022. The results of Chapter 7 are under review.

Part III of the thesis focuses on the problem of providing guarantees to constrained agents during learning processes. For instance, this is the case of financially constrained advertisers repeatedly bidding in ad auctions.

- Chapter 8 provides algorithms for the bid optimization of advertising campaigns subject to uncertain budget and ROI constraints.
- Chapter 9 studies online learning problems in which a decision maker has to take a sequence of decisions subject to long-term general constraints.

The results of Chapter 8, are under review as Castiglioni et al. (2022d), while the results of Chapter 9 were published at NeurIPS-2022 as Castiglioni et al. (2022b).

Finally, Chapter 10 concludes the thesis with some possible directions for future research.

Table 1.1 summarizes the scope and the techniques of each chapter. Specifically, the first row of the table groups problems which require techniques from the fields of mechanism design and optimization to be solved, while the second row is about online learning problems. The first column

	PRICING	ADVERTISING
MECHANISM DESIGN	Chapter 5	Chapter 5
	Chapter 3	Chapter 6
		Chapter 7
ONLINE LEARNING	Chapter 4	Chapter 8
		Chapter 9

Table 1.1: *Summary of the main problems and techniques of each chapter. Yellow cells are for problems studied from sellers' perspective, orange cells are for problems studied from advertisers' perspective, and blue cells are for problems addressed from the point of view of Web platforms.*

of the table is related to the problem of finding a pricing strategy while the second one to problems concerning Web advertising. The yellow cells are for problems studied from the perspective of a seller proposing their products on the Web who needs to specify a pricing strategy in order to maximize their revenue. The blue cells are for problems studied by the point of view of a Web platform. In particular, Chapter 5 deals with an ad allocation problem in which ads of similar products report the selling prices. We study a model in which the platform jointly optimizes prices and allocations, and, then, the more realistic scenario in which the platform optimizes the allocation, given some prices fixed by the advertisers. This motives the presence of Chapter 5 in both columns. Finally, the orange cells are for problems from the perspective of advertisers, who participate in ad auctions with the goal of finding bidding strategies that maximize their utility.

CHAPTER 2

Preliminaries

In this chapter, we introduce some concepts and notable results from the literature that will be useful in the reminder of the dissertation. In particular, we introduce notions from algorithmic game theory and, specifically, from the mechanism design literature. These concepts are essential when designing mechanisms involving rational agents, such as, for example, auctions mechanisms employed by e-commerce or advertising platforms. Moreover, we introduce the online learning framework, which is a fundamental model of sequential decision making. This can be applied, for instance, when an agents aims at finding effective bidding strategies though repeated interactions with an auction mechanism. Section 2.1 introduces the classical representation of finite games, *i.e.*, normal-form games, and defines the ubiquitous solution concept of Nash Equilibrium. Section 2.2 revolves around mechanism design and auction theory. In this section we present some of the most popular auction mechanisms such as fist-price, second-price and Vickrey-Clarke-Groves (VCG) auctions. In the following, we use the term *"agent"* to generally identify an actor participating in a game or mechanism. Furthermore, some synonyms are used for specific contexts. For instance, *"players"* are agents involved in a game, *"bidders"* or *"advertisers"* participate in auctions, while *"users"* or *"buyers"* participate in a posted-price

Symbol	Description
	Pricing Mechanisms - Chapter 3
$\mathcal{M}$	Pricing mechanism
$p_{\mathcal{M}}(t)$	Selling price assigned by $\mathcal{M}$ at time t to an item
V_i	Buyer i's private valuation for the item
ξ_i	Discount function
W_i	Buyer i's arrival time
D_i	Buyer i's discounted valuation defined as $D \coloneqq V_i \xi_i(W_i)$
	Ad Auctions - Chapters 5 6
N	Set of agents defined as $N \coloneqq \{1, \dots, n\}$
M	Set of slots defined as $M \coloneqq \{1, \dots, m\}$
θ_i	Agent i's type
v_i	Agent i's valuation
f	Allocation function defined as $f \colon N \to M \cup \{\perp\}$
λ_j	Prominence of slot j
b_i	Agent i's bid
π_i	Agent i's payment
u_i	Agent i's utility
SW	Social Welfare of an allocation
Rev	Revenue of an allocation
	Ad Auctions with Price Displaying - Chapter 5
p_i	Selling price of agent i's product
c_i	Supply cost of agent i's product
α_i	Conversion probability of agent i's product
$p_{\min}$	Minimum displayed price
$q_i(p_i, p_{\min})$	Agent i's quality
$\mathcal{M}_{\mathrm{D}}$	Direct-revelation mechanism
$\mathcal{M}_{\mathrm{I}}$	Indirect-revelation mechanism
	Metaverse - Chapter 7
T	Tree of scenes
s	Scene in which a user can be
Π^s	Reach probability of scene s from the root node
$\pi_{s,s'}$	Transition probability from scene s to scene s'
a	Ad
$\gamma_{a,a'}$	externalities of ad a on ad a'
$q_{a,s}$	quality of ad a displayed in scene s

Table 2.1: *Summary of the notation used in the thesis for ad auctions and pricing mechanisms.*

Symbol	Description
	Online learning Frameworks - Chapters 4 8 9
T	Time horizon
R^T	Pseudo-regret at time T
	TP-MAB - Chapter 4
$\mathcal{A}$	Set of arms
$\tau_{\max}$	Time span over which the reward is partitioned
$\boldsymbol{w}_t^i$	Vector of realized per-round rewards collected from the pull of arm i at round t
$W_{t,j}^i$	Random variable of the per-round reward observed at round j from the pull of arm i at round $t \leq j$
$w_{t,j}^i$	Realization of random variable $X_{t,j}^i$
R_t^i	Random variable of the cumulative reward collected from the pull of arm i at round t
r_t^i	Realization of random variable R_t^i
μ_i	Expected cumulative reward of arm i
i^*	Optimal arm
α	Smoothness parameter
	Chapter 8
$\mathcal{C}$	Advertising campaign composed of N subcampaigns C_j, for $j \in \{1, \dots, N\}$
$x_{j,t}$	Bid for subcampaign C_j an time t
$\hat{x}_{j,t}$	Bid for subcampaign C_j an time t suggested by a learning policy
λ	ROI threshold
β	Daily budget
$n_j(x_{j,t})$	Expected number of clicks given the bid $x_{j,t}$ for subcampaign C_j
$c_j(x_{j,t})$	Expected cost given the bid $x_{j,t}$ for subcampaign C_j
v_j	Value per click for subcampaign C_j
	Chapter 9
$\mathcal{X}$	Set of strategies
$\boldsymbol{x}_t$	Action taken by the agent at time t
f_t	Reward function selected by the environment at time t
g_t	Constraint function selected by the environment at time t
m	Number of constraints
Ξ	Set of strategy mixtures
ξ	Strategy mixture
d_g	Largest possible value for which there exists a strategy mixture satisfying the constraints by a margin of at least d_g
$\mathcal{L}_{f,g}$	Lagrangian function given reward function $f$and constraint function g
ρ	Feasibility parameter
$\hat{\rho}$	Lower bound on ρ
V^T	Cumulative violation at time T

Table 2.2: *Summary of the notation used in the thesis for online learning frameworks.*

mechanism. In each of the following chapters, we will specify the context, nonetheless, in many cases these terms are interchangeable. For example, an auction with strategical bidders can be seen as a game. Moreover, we use the term *"auctioneer"* to denote the agent running an auction. We use *"platform"* as a synonym of "auctioneer" when we deal with ad auctions run by Web platforms, and we use the synonym *"seller"* in the context of pricing problems. Finally, we provide Table 2.1 and Table 2.2, which summarizes the notation used in the thesis. Specifically, Table 2.1 specifies the notation used in chapters revolving around mechanism design problems while Table 2.2 specifies the notation used in chapters dealing with online learning problems. Notice that, in ad auctions we define bidder i's bid as b_i, while in the online learning framework we denote by x_t the action taken by an agent at time t. In Chapter 8 we study a learning problem in which a bidder has to select a bid, *i.e.*, take an action, at each time instant. Being in the online learning framework, we use the consistent notation x_t to denote the action even if, in this specific case, it coincides with a bid.

2.1 Games and Equilibria

Games provide a mathematical representation of the strategic interactions among rational agents. A game is defined by a set of players, a set of actions for each player, and a set of utility functions mapping the space of players' strategies to the space of outcomes.

We introduce the normal-form representation of a game. It provides a *static* representation of the game, and perfectly describe, for instance, simultaneous-move games. Formally, we can define a normal-form game as follows.

Definition 2.1 (Normal-form game)**.** *A normal-form game is a tuple* (N, A, U) *such that:*

- $N \coloneqq \{1, \dots, n\}$ *is the set of players;*
- $A \coloneqq \{A_1, \dots, A_n\}$ *is the set of action profiles, where* A_i *denotes the set of actions available to player* i *and* $|A_i|$ *is the number of actions available to player* i*;*
- $U \coloneqq \{U_1, \dots, U_n\}$ *is the set of utility functions and* $U_i : A_1 \times \dots \times A_n \to \mathbb{R}$ *is the utility function of player* i*.*

An action profile $a = (a_i)_{i \in N}$ is a tuple, with $a_i \in A_i$ for every $i \in N$. We denote with $a_{-i} = (a_1, \dots, a_{i-1}, a_{i+1} \dots, a_n)$ the actions of all the

players except for player i. When a player deterministically chooses an action, they are playing a *pure* strategy. Otherwise, if a player randomizes among actions, they are playing a *mixed* strategy. We can represent player i's mixed strategies as $x_i \in \Delta_{A_i} \coloneqq \{x_i \in [0,1]^{|A_i|} : \sum_{j \in A_i} x_{i,j} = 1\}$, where $x_{i,j}$ denotes the probability that player i plays action $j \in A_i$ and Δ_{A_i} denotes the simplex over the space of actions A_i. We denote with $x = (x_1, \ldots, x_n)$ a mixed strategy profile that specifies a mixed strategy $x_i \in \Delta_{A_i}$ for each player $i \in N$. Finally, we define $u_i(x) \coloneqq \sum_{a \in A} U_i^a \prod_{i \in N} x_{i,a_i}$ the expected utility of player $i \in N$, where U_i^a correspond to the utility of player i when the players play action profile $a \in A$.

In this thesis, we consider many game theoretic situations in which a player is unsure about the preferences or intentions of others. Incomplete information introduces additional strategic interactions. Formally, we define Bayesian games, or incomplete-information games, as follows.

Definition 2.2 (Bayesian Game)**.** *A Bayesian game cosists of*

- $N \coloneqq \{1, \ldots, n\}$*, a set of players;*
- $A \coloneqq \{A_1, \ldots, A_n\}$*, the set of action profiles, where* A_i *denotes the set of actions available to player* i *and* $|A_i|$ *is the number of actions available to player* i*;*
- $\Theta \coloneqq \{\Theta_1, \ldots, \Theta_n\}$*, the set of types, where* Θ_i *denotes the set of types of player* i*. Moreover, there is a prior distribution* D_i *on* Θ_i*. A type* $\theta_i \in \Theta_i$ *is a private information of player* i *and* $D_i(\theta_i)$ *is the a priori probability that player* i *is of type* θ_i*;*
- $U \coloneqq \{U_1, \ldots, U_n\}$*, the set of utility functions, where* $U_i : \Theta_i \times A_1 \times \ldots \times A_n \to \mathbb{R}$ *is player* i*'s utility function and* $U_i(\theta_i, a_1, \ldots, a_n)$ *is the utility achieved by player* i *when their type is* θ_i *and the action profile of all players is* $(a_1, \ldots, a_n)$*.*

2.1.1 Solution Concepts

In what follows, we assume that players are rational and aim at maximizing their utilities. While in single-agent problems it is clear that the best solution is to optimize the player's objective, in games including multiple agents with different objectives, more complex solution concepts are needed. Various solution concepts has been defined in the game theory literature and, usually, they represent an equilibrium, *i.e.*, a stable solution in which the players have no incentive to modify their strategies or leave the game.

The Nash Equilibrium (NE), introduced by Nash (1950), is the most famous and used solution concept. This equilibrium is a strategy profile such that no player has an incentive to unilaterally deviate from their strategy, when other players' strategies remain unchanged. Formally:

Definition 2.3. *A mixed strategy profile $x = (x_1, ..., x_n) \in \times_{i \in N} \Delta_{A_i}$ is a Nash Equilibrium of a normal-form game (N, A, U) if for every player $i \in N$ and strategy $x'_i \in \Delta_{A_i}$ it holds that*

$$u_i(x) \geq u_i(x'_i, x_{-i})$$

If players are allowed to play mixed strategies, then any normal-form game admits at least a Nash Equilibrium.

Theorem 2.1 ((Nash, 1950))**.** *Every Normal-Form game admits at least one Nash Equilibrium.*

Now, we provide the notion of equilibrium for Bayesian games.

Definition 2.4 (Bayesian Nash Equilibrium)**.** *A pure strategy of player i is a function $s_i : \Theta_i \to A_i$. A profile of strategies $s = (s_1, \ldots, s_n)$ is a Bayesian-Nash Equilibrium (BNE) if for every player i and every type θ_i we have that $s_i(\theta_i)$ is the best response that i has to s_{-i} when player i's type is θ_i, in expectation over other players' types. Formally, for all i, θ_i and, a'_i:*

$$\mathbb{E}_{D_{-i}}[u_i(\theta_i, s_i(\theta_i), s_{-i}(t_{-i}))] \geq \mathbb{E}_{D_{-i}}[u_i(\theta_i, a'_i, s_{-i}(t_{-i}))]$$

where $\mathbb{E}_{D_{-i}}[\cdot]$ is the expectation over other players' types θ_{-i} drawn according to distribution D_{-i}.

A mixed strategy of a player i is a function $s'_i : \Theta_i \to \Delta_{A_i}$. An analogous definition of BNE holds for mixed strategies. In this case, the expected utilities are computed with respect to the distribution of types and the randomicity of the strategies.

2.2 Mechanism Design and Auction Theory

Mechanism Design is a subfield of economics and game theory that studies the design of economic mechanisms or incentives toward desired objectives. Mechanism design is broadly applied in areas ranging from economics and politics to networked-systems. Some examples related to the first areas are market design, auction theory and social choice theory, while Internet interdomain routing and sponsored search auctions are related to the latter. The goal of the designed mechanisms is summarized in a *social choice*

function which aggregates the preferences expressed by the participants toward a single decision point. We start by providing the formulation of a social choice function. Consider a set $N \coloneqq \{1, \ldots, n\}$ of agents. Each agent $i \in N$ is described by a private type $\theta_i \in \Theta_i$ and the tuple $\theta \coloneqq (\theta_1, \ldots, \theta_n)$ is called type profile.

Definition 2.5 (Social Choice Function)**.** *A social choice function $f : \Theta_1 \times \ldots \times \Theta_n \to X$ assigns an outcome $x \in X$ to each possible profile of agents' types.*

In what follows, we introduce some definitions and properties of mechanisms. We will often refer to the notions provided in this section, particularly in Part II of the thesis. First, we provide the definition of economic mechanism.

Definition 2.6 (Economic mechanism)**.** *An economic mechanism is a tuple $(A_1, \ldots, A_n, X, g)$ where A_i is agent i's set of actions, X is the set of outcomes and $g : A_1 \times ... \times A_n \to X$ is the outcome function.*

Each agent i chooses their action a_i without knowing other agents' types, which, at the end of the interaction, affect the utility of i by determining other agents' actions. The behavior of agent i is defined by a strategy function $s_i : \Theta_i \to A_i$, that specifies an action $a_i \in A_i$, given a type $\theta_i \in \Theta_i$. Therefore, each agent i chooses action $s_i(\theta_i)$ and the mechanism determines an outcome $x = g(s_1, ..., s_n)$. Now we define a class of mechanisms in which the actions available to each agent i are given by the set Θ_i of types of that agent and the outcome function is a social choice function.

Definition 2.7 (Direct Revelation Mechanism)**.** *Given a social choice function $f : \Theta_1 \times ... \times \Theta_n \to X$, a mechanism $(\Theta_1, ...\Theta_n, X, f)$ is called direct revelation. A mechanism that is not a direct revelation is called indirect revelation.*

For instance, in the auction setting, an auction mechanism is direct-revelation if all bidders bid their true valuation, otherwise the auction mechanism is indirect revelation.

Since agents are rational, given a mechanism, they select strategic actions to maximize their utility functions. This behavior induces a Bayesian game.

Definition 2.8. *An economic mechanism $(A_1, \ldots, A_n, X, g)$ combined together with each agents i's possible types Θ_i, probability distributions Ω_i over types, and agents' utility functions is a Bayesian game.*

Definition 2.9. *A mechanism* $(A_1, \ldots, A_n, X, g)$ *implements a social choice function* $f : \Theta_1 \times \ldots \times \Theta_n \rightarrow X$ *if there is a pure-strategy equilibrium* $(s_1^*, \ldots, s_n^*)$ *of the Bayesian game induced by the mechanism such that* $g(s_1^*(\theta_1), \ldots, s_n^*(\theta_n)) = f(\theta_1, \ldots, \theta_n), \forall(\theta_1, \ldots, \theta_n) \in \Theta_1 \times \ldots \times \Theta_n$, *where* $s_i^*(\theta_i)$ *denotes the optimal strategy of agent* i *when their type is* θ_i.

The implementation of a social choice function can be achieved by different solution concepts such as *dominant-strategy* equilibrium, if $(s_1^*, \ldots, s_n^*)$ is a (weak) dominant-strategy equilibrium, or *Bayes-Nash* equilibrium, if $(s_1^*, \ldots, s_n^*)$ is a Bayes-Nash equilibrium. Now, we define mechanisms that induce the agents to report their true type.

Definition 2.10. *A social choice function* $f : \Theta_1 \times \ldots \times \Theta_n \rightarrow X$ *is said incentive compatible (or truthful, or strategyproof) if the Bayesian game induced by the direct revelation mechanism* $(\Theta_1, \ldots, \Theta_n, X, f)$ *has a pure equilibrium* $s^* = (s_1^*, \ldots, s_n^*)$ *such that* $s_i^*(\theta_i) = \theta_i$ *for every type* $\theta_i \in \Theta_i$ *and agent* $i \in N$.

Definition 2.11. *Incentive compatibility can be satisfied according to different solution concepts, such as:*

- *Dominant-strategy incentive compatibility (DSIC), if* $(s_1^*, \ldots, s_n^*)$, *where* $s_i^*(\theta_i) = \theta_i$, *for every agent* i *and type profile* θ *is a (weak) dominant-strategy equilibrium*
- *Bayesian incentive compatibility (BNIC), if* $(s_1^*, \ldots, s_n^*)$, *where* $s_i^*(\theta_i) = \theta_i$, *for every agent* i *and type* θ_i *is a Bayes-Nash equilibrium.*

The fact that a social choice function f is incentive compatible means that there is a direct-revelation mechanism that implements f. Since direct-revelation economic mechanisms constitute a strict subclass of economic mechanisms, the fact that a social choice function f is not incentive compatible suggests the problem of finding an indirect economic mechanism that implements f, if such mechanism exists.

Theorem 2.2 (Revelation Principle)**.** *Given a social choice function* $f : \Theta_1 \times \ldots \times \Theta_n \rightarrow X$, *if there is a mechanism* $(A_1, \ldots, A_n, X, g)$ *implementing* f *in dominant-strategy , or Bayes-Nash, equilibrium, then* f *is DSIC, or BNIC, respectively.*

The revelation principle shows that, for dominant-strategy equilibria and Bayes-Nash equilibria, if a social choice function f is not incentive compatible and, therefore, the direct-revelation mechanism does not implement

it, then there is no indirect economic mechanism implementing it. Therefore, given a social choice function f, there exists an economic mechanism implementing it if and only if f is incentive compatible.

Mechanisms Performance Metrics. In the literature, the *Price of Anarchy* (PoA) and the *Price of Stability* (PoS) are commonly-adopted efficiency metrics for economic mechanisms. In particular, consider a game induced by a mechanism, and its equilibria. The price of anarchy is the most popular measure of inefficiency of equilibria and resolves the issue of multiple equilibria by adopting a worst-case approach. Precisely, the PoA of a game is defined as the ratio between the worst objective function value of an equilibrium of the game and that of an optimal outcome. Notice that the PoA is defined with respect to the choice of an objective function and that of an equilibrium concept. The PoS is a measure of inefficiency designed to differentiate between games in which all equilibria are inefficient and those in which some equilibria are inefficient. Formally, the PoS of a game is the ratio between the best objective function value of one of its equilibria and that of an optimal outcome. In a game with a unique equilibria, the PoA and the PoS are identical. The range of PoA and PoS is $[0, 1]$. Many works from the literature define the PoA and the PoS as the inverse of the ratios defined above. In this case, their range is $[1, +\infty]$.

2.2.1 Auctions

The agents interacting in an auction mechanism are the *auctioneer* and a set of *bidders* $N \coloneqq \{1, \dots, n\}$. The set of item sold through the auction is $M \coloneqq \{1, \dots, m\}$. In this thesis, we mainly consider *single parameter environments*, which are defined as follows. The auctioneer sells a product and each bidder i has a private valuation v_i per unit of product that they get. The term "private" means that the valuation is unknown to the seller and to other bidders. Then, there is a *feasible set* Y and each element of Y is a tuple $y = (y_i)_{i \in N}$, where $y_i \in Y_i$ denotes the amount of product given to bidder i, and Y_i is bidder i's feasibility set. For instance, in a single item auction, $y_i \in [0, 1]$ if the item is divisible or $y_i \in \{0, 1\}$ if it is indivisible. For all $i \in N$, y is such that $\sum_{i=1}^{n} y_i \leq 1$. Alternatively, if there are k identical goods and there is a constraint imposing that each bidder gets at most one of them, the feasible set is such that $y_i = [0, 1]$ or $y_i = \{0, 1\}$ for all i and $\sum_{i=1}^{n} y_i \leq k$. In ad-auction settings where each slot is assigned at most one bidder, and vice versa, if bidder i is assigned to slot j, then the component y_i equals the click probability on that slot.

In this environment, each bidder i is characterized by a type θ_i, that is their private valuation v_i. The tuple collecting all bidders' valuations is defined as $v = (v_i)_{i \in N}$, while v_{-i} is the tuple of all bidders' valuations except for v_i. Moreover, $v \in V$ and $v_i \in V_i$, where $V = V_1 \times \ldots \times V_n$ and V_i is the set of possible valuation of bidder i. Then, each bidder i privately communicates a bid $b_i \in B_i$ to the auctioneer, where B_i is the set of bidder i's bids. The bid is an offer reflecting how much they are willing to pay for the item. This motives the common assumption in standard settings that bidders do not overbid, *i.e.*, $0 \leq b_i \leq v_i$. In the next chapters, we will specify the scenarios in which bidders can overbid, *i.e.*, bid a value larger than their valuation. We denote the bid profile by $b = (b_i)_{i \in N}$, while b_{-i} is the tuple of all bids except for b_i. Moreover, $b \in B$, where $B = B_1 \times \ldots \times B_n$.

In the following chapters, we frequently consider the class of *sealed-bid auctions*, described by the following steps.

- The auctioneer collects the bids $b_1, \ldots, b_n$
- Chooses a feasible allocation $y(b) \in Y$ as a function of the bids (*i.e.*, *allocation rule*)
- Chooses payments $\pi(b)$ as a function of the bids (*i.e.*, *payment rule*)

Intuitively, the auction is defined when the auctioneer fixes the quantity of items to assign to each bidder and the amount each bidder has to pay for it.

The adopted bidder utility model is called *quasilinear utility model* and is defined as follows. Given a bid profile b, an allocation rule y, and a payment rule π, bidder i has utility of

$$u_i(b) = v_i y_i(b) - \pi_i(b).$$

When the payment is such that $\pi_i(b) \geq 0$, it means that the auctioneer cannot pay the bidders. In this case, monetary transactions are unidirectional from bidders to the auctioneer. When the payment is such that $\pi_i(b) \leq b_i y_i(b)$, it means that a truthtelling bidder receives nonnegative utility.

We define the social welfare of an auction $\mathrm{SW} \coloneqq \sum_{i \in N} v_i y_i(b)$ as the sum of bidders' revenues and its revenue $\mathrm{Rev} \coloneqq \sum_{i \in N} \pi_i(b)$ as the sum of bidders' payments, given the outcome $y(b)$.

Finally, the *Price of Anarchy* (PoA) and the *Price of Stability* (PoS) are commonly-adopted efficiency metrics for auctions mechanisms.

Now we survey some broadly used auction mechanisms.

First-price Auction. Consider a single item that is auctioned for sale among n bidders. In a first-price auction the allocation rule is such that the winner

is bidder $i \in N$ with the highest bid value b_i among all the bidders. The payment rule is such that the winner pays their own bid value for getting the item, *i.e.*, $\pi_i(b) = \max_j b_j$, and all the other bidders pay zero. Notice that bidding truthfully in a first-price auction guarantees zero utility. The ideal amount to underbid depends on the bids of the other bidders.

Proposition 2.1. *First-price auction*

- *can be implemented in polynomial time.*

Second-price Auction. Second-price auction is also called Vickrey auction. Consider a single item that is auctioned for sale among n bidders. In a second-price auction the allocation rule is such that the winner is bidder $i \in N$ with the highest bid b_i among all the bidders. The payment rule is such that the winner i pays the second highest bid value for getting the item, *i.e.*, $\pi_i(b) = \max_{j \neq i} b_j$, and all the other bidders pay zero.

In a second-price auction every bidder has a dominant strategy, which is setting their bid b_i equal to their valuation v_i. Moreover, every truthtelling bidder is guaranteed non-negative utility. The following proposition states the main properties of a second-price auction.

Proposition 2.2. *Second-price auction*

- *is dominant-strategy incentive-compatible (DSIC)*
- *maximizes social welfare, if bidders bid truthfully*
- *can be implemented in polynomial time.*

Vickrey-Clarke-Groves Mechanism. We denote by K the set of possible allocations and by K_{-i} the set of allocations in which bidder i is not present. We denote by $v_i(k)$ the revenue of bidder i for what they get when the allocation is $k \in K$. A direct-revelation auction mechanism is a Vickrey-Clarke-Groves (VCG) mechanism if

$$y(v) \in \arg\max_{k \in K} \sum_{i \in N} v_i(k)$$

$$\pi_i(v) \coloneqq \max_{k' \in K_{-i}} \sum_{j \in N \setminus \{i\}} v_j(k') - \sum_{j \in N \setminus \{i\}} v_j(k), \quad \forall i \in N.$$

Intuitively, bidder i pays an amount equal to the total damage that i causes to the other bidders, that is the difference between the social welfare of other bidders with and without i's participation. Alternatively, we can

say that the payments are such that each bidder i internalize the externalities caused to the others by i's presence in the mechanism.

Proposition 2.3. *The Vickrey-Clarke-Groves mechanism*

- *is dominant-strategy incentive-compatible (DSIC)*
- *maximizes social welfare, if bidders bid truthfully*

Proposition 2.3 states some desirable properties of VCG mechanisms. While VCG can be infeasible to implement in practice, it is widely used as a benchmark for other practical approches.

2.2.2 Ad Auctions

We denote by $N \coloneqq \{1, \ldots, n\}$ the set of bidders. The advertisers compete for displaying their ads on a set $M \coloneqq \{1, ..., m\}$ of *slots*. Each bidder $i \in N$ has a type, that is a *private valuation* v_i for an advertising slot, which reflects how much they value a click on their ad. Furthermore, they are characterized by a parameter β_i, which is the probability that a user clicks on the specific ad of bidder i, and the *conversion probability* α_i, which is the probability that a user, after clicking the ad, buys the product. Parameter β_i reflects users' interest in bidder i's ad content, while α_i can be interpreted as a measure of their intention to effectively purchase the item. Then, each slot $j \in M$ is associated with a *prominence* parameter $\lambda_j \in [0, 1]$, encoding the probability with which the slot is clicked by a user. The prominence reflects the visibility of a specific solt, which depends on features such as its dimension and positioning on the Web page. In the literature, the *click-through rate* (CTR) is the ratio between the number of clicks on a specific ad and the number of times that the ad has been showed on a Web page. We define with $\mathsf{CTR}_{i,j} \coloneqq \lambda_j \beta_i$ the click-through rate of bidder i's ad in slot j. Moreover, by denoting with G_i bidder i's *gain* per item sold, we define as $v_i^1 \coloneqq \alpha_i G_i$ their expected valuation, given that a user already clicked on the ad. Sometimes, expliciting parameter β_i may cause an overload of notation. As is customary in the literature, the parameter can be easily omitted by using the notation $v_i^2 \coloneqq \beta_i v_i^1$ to express the expected valuation of bidder i w.r.t. the clicks and purchases, once the user observed the corresponding slot. Finally, we provide another compact notation for bidder i's expected valuation, which is $v_i^3 \coloneqq \lambda_{f(i)} \beta_i v_i^1$, where $f(i) = j$ if the allocation rule assigns slot j to bidder i. In this case, the expectation is w.r.t. users' observation of the slot, clicks on the ad and purchases.

Each agent $i \in N$ participates in the ad auction with a *bid* b_i representing how much they are willing to pay at most for a click on their ad. We denote

by $b = (b_i)_{i \in N}$ the bid profile composed of all the agents' bids. W.l.o.g., we assume that the slots are ordered in decreasing value of prominence, so that $\lambda_1 \geq \ldots \geq \lambda_m$. Moreover, for the ease of presentation, we let $\lambda_{m+1} = \ldots = \lambda_{n+1} = 0$.

The auction goes on as follows. First, bidders select a bidding strategy b. Then, bidders individually report their bids to the auction mechanism. Finally, the mechanism allocates an ad to each slot and defines a *payment* $\pi_i(b)$ for each bidder $i \in N$.

Given a bid profile b, we denote bidder i's *utility* by $u_i(b) \coloneqq v_i(b) - \pi_i(b)$, when i's ad is assigned to a slot, otherwise the utility is zero. Finally, we define the *social welfare* of the allocation as $\mathrm{SW} = \sum_i v_i(b)$, while its *revenue* is defined as $\mathrm{Rev} = \sum_i \pi_i$, where the sums are taken on bidders i whose ad has been assigned to a slot. Analogously to bidders' valuations, also bids, payments, utilities and SWs can be expressed in expectation. The choice of notation depends on what is more suitable to model a specific scenario. In Table 2.3 we summarize the notation adopted in the following chapters, clarifying when we consider expected values. As we did for bidders' valuations, we use a superscript equal to

- 1, when the expectation is taken w.r.t. the purchases
- 2, when the expectation is taken w.r.t. the purchases and clicks on the ad
- 3, when the expectation is taken w.r.t. the purchases, clicks on the ad and observation of the slot.

Finally, we summarize notable properties of ad auctions mechanisms (see Mas-Colell et al. (1995) for their general definitions). An ad auction mechanism is

- *individually rational*, if for every bidder i, the assigned payment π_i is non-larger than the bid value b_i.
- *weakly budget-balanced* if the sum of payments is always non-negative, *i.e.*, $\mathrm{Rev} = \sum_{i \in N} \pi_i(b) \geq 0$.
- *truthful* if for every bidder i it is a dominant strategy to report their type v_i to the mechanism, *i.e.*, the utility u_i that bidder i achieves by reporting v_i is at least as large as with every alternative input, regardless of other bidders' actions.

Chapter 5	
λ_j	Prominence of slot j
α_i	Bidder i's conversion probability
$q_i(p_i, p_{\min})$	Bidder i's quality function. Given prices p_i, $p_{\min}$ and an allocation, it coincides with bidder i's click probability β_i. In this chapter we call it *quality* and depends on the specific setting under study
p_i, c_i	Selling price and supply cost of bidder i's product, respectively. The gain is expressed as $G_i \coloneqq p_i - c_i$
f	Allocation function, defined as $f\colon N \to M \cup \{\perp\}$
v_i	Expected valuation of bidder i, when assigned to slot j, defined as $v_i \coloneqq v_i^3 = \lambda_j q_i(p_i, p_{\min})\alpha_i(p_i - c_i)$.
b_i	Bidder i's bid. When truthful, it is such that $b_i \coloneqq b_i^1 = \alpha_i(p_i - c_i)$
$\pi_i(b)$	Bidder i's expected payment when assigned to slot j, defined as $\pi_i(b) \coloneqq \pi_i^3(b)$
Chapter 6	
λ_j	Prominence of slot j
v_i	Expected valuation of bidder i, defined as $v_i \coloneqq v_i^2$
$\pi_i(b)$	Bidder i's expected payment when assigned to slot j, defined as $\pi_i(b) \coloneqq \pi_i^3(b)$
b_i	Bidder i's bid, defined as $b_i \coloneqq b_i^2$.
Chapter 8	
v_j	Bidder's expected valuation for subcampaign j, defined as $v_j \coloneqq v_j^1$
$n_j(x_{j,t})$	Expected number of clicks for subcampaign j, given action, *i.e.* a bid, $x_{j,t}$ at time t
$c_j(x_{j,t})$	Expected cost for subcampaign j, given action, *i.e.* a bid, $x_{j,t}$ at time t

Table 2.3: *Detailed summary of the ad-auction notation and clarifications on its use in the each chapter.*

Next, we review GSP and VCG mechanisms applied to ad auctions (see the book by Nisan and Ronen [2001] for their general description). Consider a bid profile b and a valuation profile v. Assuming w.l.o.g. that $b_1 \geq \ldots \geq b_n$ (by re-labeling bidders accordingly), both mechanisms orderly assign the first m agents, who are those with the highest bids, to the first m slots, which are those with the highest prominences. Moreover, the mechanisms assign the following expected payments.

- *GSP mechanism*: $\pi_i^{\text{GSP}}(b) := \lambda_i b_{i+1}$ for each agent $i \in [m]$, and $\pi_i^{\text{GSP}}(b) = 0$ for all the other agents.
- *VCG mechanism*: $\pi_i^{\text{VCG}}(b) := \sum_{j=i+1}^{m+1} b_j(\lambda_{j-1} - \lambda_j)$ for each agent $i \in [m]$, and $\pi_i^{\text{VCG}}(b) = 0$ for the others.

The VCG payments are such that each bidder is charged the externalities that they impose on other bidders through their participation in the auction. This makes the VCG mechanism *truthful*, which means that it is a dominant strategy for each agent to report their true valuation to the mechanism, namely $b_i = v_i$ for every $i \in N$. This is *not* the case for the GSP mechanism.

2.3 Online Learning Framework

We consider the following online setting. An agent plays a repeated game over T rounds. At each round $t \in [T]$, the agent plays an action $x \in X$ while the environment selects an utility function u. [1] At each round $t \in [T]$, after selecting the action x_t, the agent observes an utility $u_t(x_t)$, where $u_t : X \to [0, 1]$.

We are interested in algorithms computing x_t at each round t. The performance of such algorithms is measured using the *regret* computed with respect to the best fixed action in hindsight. Formally:

$$R^T := \max_{x \in X} \sum_{t=1}^{T} u_t(x) - \mathbb{E}\left[\sum_{t=1}^{T} u_t(x_t)\right],$$

where the expectation is on the randomness of the online algorithm. Ideally, we would like to find an algorithm that generates a sequence $\{x_t\}_{t \in [T]}$ such that the regret is sublinear in T. An algorithm satisfying this property is usually called a *no-regret* algorithm. In the case in which requiring no-regret is too limiting, we use the following relaxed notions of regret. Given an

[1] The set $\{1, \ldots, n\}$ is denoted by $[n]$.

$\alpha \in [0, 1]$, the α-multiplicative-regret of an algorithm is defined as follows:

$$R^T_{\mathrm{M},\alpha} \coloneqq \alpha \max_{x \in X} \sum_{t=1}^{T} u_t(x) - \mathbb{E}\left[\sum_{t=1}^{T} u_t(x_t)\right],$$

while the α-additive-regret of an algorithm is defined as follows:

$$R^T_{\mathrm{A},\alpha} \coloneqq \max_{x \in X} \sum_{t=1}^{T} u_t(x) - \alpha - \mathbb{E}\left[\sum_{t=1}^{T} u_t(x_t)\right].$$

We call an algorithm that has α-multiplicative-regret or α-additive-regret sublinear in T a no-α-multiplicative-regret or no-α-additive-regret algorithm, respectively. The idea of no-α-regret is that the algorithm has no-regret with respect to an approximation of the optimal fixed action.

2.3.1 Multi-Armed Bandit Problems

In a multi-armed bandit problem, the agent chooses an action (*i.e.* pulls an arm) at each time step, and then observes the corresponding reward. Bandit problems are basic instances of sequential decision making with limited information and naturally address the fundamental tradeoff between exploration and exploitation in sequential experiments. Indeed, the agent needs to balance the exploitation of actions that did well in the past and the exploration of actions that might give higher payoffs in the future. Clinical trials were the original motivation of Thompson (see Thompson (1933)) for studying bandit problems. When different treatments are available for a certain disease, one must decide which treatment to use on the next patient. However, modern technologies have created many opportunities for new applications. Nowadays, bandit problems play an important role in several industrial domains. In particular, online services are natural targets for bandit algorithms, because one can benefit from adapting the service to the individual sequence of requests. For instance, given a set of Web advertisements, consider the ad-placement problem which consists of deciding which advertisement to display on the Web page delivered to the next visitor of a website. Similarly, website optimization deals with the problem of sequentially choosing design elements, such as font, images and layout, for the Web page. Here the payoff is associated with user's actions, e.g., clicks or other desired behaviors. Notice that there are important differences with respect to the basic bandit problem. For instance, in ad placement the pool of available ads, which are bandit arms, may change over time, and there might be a limit on the number of times each ad could

be displayed. Many other real-world scenarios as the ones mentioned above, can be efficiently solved by applying techniques from the bandits framework. This fact results in the proliferation of many variants of the standard bandit problem, such as the one we present in Chapter 4. In what follows, we present a well-known bandit algorithm from the literature, which applies to the standard stochastic bandit setting and will be used in the rest of the thesis.

Algorithm 2.1 `UCB1`

1: **for** $t \in \{1, \dots, K\}$ **do** ▷ init phase
2: Pull arm $i_t = t$
3: Observe the reward $r_t^{i_t}$ of the arm pulled at round t
4: **end for**
5: **for** $t \in \{K+1, \dots, T\}$ **do** ▷ loop phase
6: **for** $i \in \{1, \dots, K\}$ **do**
7: $\hat{R}^i_{t-1} \leftarrow \frac{1}{n^i_{t-1}} \sum_{h=1}^{t-1} r_h^i \mathbf{1}_{\{i_h = i\}}$
8: $c^i_{t-1} \leftarrow \overline{R}^i \sqrt{\frac{2 \ln t}{n^i_{t-1}}}$
9: $u^i_{t-1} \leftarrow \hat{R}^i_{t-1} + c^i_{t-1}$
10: **end for**
11: Pull arm $i_t = \operatorname{argmax}_{i \in [K]} u^i_{t-1}$
12: Observe the reward $r_t^{i_t}$ of the arm pulled at round t
13: **end for**

`UCB1` Algorithm. We describe the `UCB1` algorithm, from Auer et al. [2002a], in which the reward $r_t^{i_t}$ provided by pulling arm i_t at round t is observed by the learner at time t. We present the case in which the reward r_t^i is the realization of the random variable R^i with support $[\underline{R}^i, \overline{R}^i]$. We denote its policy by $\mathfrak{U}_{\texttt{UCB1}}$.

The pseudo-code of the `UCB1` algorithm is reported in Algorithm 2.1. During the initialization phase, all the arms are pulled once (Line 2). Subsequently, at each round t, the learner computes the empirical mean of the cumulative rewards $\hat{R}^i_{t-1}$ collected up to round $t-1$ (Line 7), where we denote by $n^i_{t-1} := \frac{1}{n_i} \sum_{h=1}^{t-1} \mathbf{1}_{\{i_h = i\}}$ the number of times the arm i has been pulled up to round $t-1$, and the confidence interval c^i_{t-1} (Line 8). Finally, the learner pulls the arm with the largest upper confidence bound u^i_{t-1} (Line 11), and observes the reward $r_t^{i_t}$ (Line 12).

We provide the following upper bound on the regret of the `UCB1` algorithm (see Auer et al. (2002a)):

Theorem 2.3. *The pseudo-regret of* `UCB1` *after* $T \in \mathbb{N}^*$ *rounds on a MAB problem with* r_t^i *rewards is:*

$$R^T(\mathfrak{U}_{\texttt{UCB1}}) \leq \sum_{i:\mu_i<\mu^*} \frac{8(\overline{R}^i)^2 \ln T}{\Delta_i} + \left(1 + \frac{\pi^2}{3}\right) \sum_{i:\mu_i<\mu^*} \Delta_i.$$

Part I

Expanding Algorithmic Pricing: Temporal Dependency

CHAPTER 3

Online Posted Pricing with Unknown Time-Discounted Valuations

This chapter studies the problem of selling a perishable item whose value diminishes over time. A significant challenge in examining the temporal aspect of this problem is finding a balance between setting high prices to maximize revenue and setting low prices to increase the probability of selling the item. This must be done while considering that the item's value will expire within a finite time horizon. We provide a posted price mechanism to address the problem and we evaluate it by performing a competitive analysis. Selling a single unit of a product does not allow the use learning algorithms, which is the reason behind the different techniques used in Chapter 3 and Chapter 4. While the former aims to sell a single unit of a single item, the latter focuses on selling multiple units of a single item.

In Section 1.1 we introduced the topic of this chapter, which is the problem of designing *posted-price mechanisms* in order to sell a single unit of a single item within a finite period of time. The chapter is structured as follows: Section 3.1 formally defines the model and the most suitable performance measure for this setting which is the *competitive ratio*. In Section 3.2 we focus on the *identical valuation* setting, where all the customers value

the item for the same amount. In this setting, we provide a mechanism $\mathcal{M}_{\text{C}}$ that achieves the best possible competitive ratio, discussing its dependency on the parameters in the case of linear discount. In Section 3.3 we switch to the *random valuation* setting. provide posted price mechanisms for the identical-valuation and the random-valuation setting, respectively. We show that, if we restrict the attention to distributions of valuations with a monotone hazard rate, then the competitive ratio of $\mathcal{M}_{\text{C}}$ is lower bounded by a strictly positive constant that does not depend on the distribution. Moreover, we provide another mechanism, called $\mathcal{M}_{\text{PC}}$, which is defined by a piecewise constant pricing strategy and reaches performances comparable to those obtained with $\mathcal{M}_{\text{C}}$. This mechanism is useful when the seller cannot change the posted price too often. Finally, in Section 3.4, we empirically evaluate the performances of our mechanisms in a number of experimental settings.

3.1 Model

We study a model in which a seller is interested in selling a single unit of an item within a finite time period of length T. The seller implements a *posted-price mechanism* by setting a take-it-or-leave-it price at each time $t \in [0, T]$. We denote by $p : [0, T] \to \mathbb{R}_+$ the *pricing strategy* adopted by the seller, with $p(t)$ being the price offered at time $t \in [0, T]$. The agents (*i.e.*, the buyers) arrive sequentially over time, according to a Poisson process with rate parameter $\lambda > 0$.

We label agents according to their order of arrival (*i.e.*, agent i is the i-th agent arriving in $[0, T]$). Each agent i has a private valuation V_i for the item, drawn from a distribution F with finite support $[v_{\min}, v_{\max}]$, where $v_{\max} > v_{\min} > 0$ denote the maximum and minimum valuation, respectively. In the following, for the ease of presentation, we normalize agents' valuations in the range $[1, h]$, where $h \coloneqq \frac{v_{\max}}{v_{\min}}$. Accordingly, we scale the support of F to $[1, h]$. Then, we denote by f the probability density function of F.

The value of the item for sale decreases over time. In particular, V_i is agent i's initial valuation at time $t = 0$. We model decreasing values by introducing a continuous non-increasing *discount function* $\xi : [0, T] \to [0, 1]$ such that $\xi(0) = 1$ and $\xi(T) = 0$. By letting W_i be the random variable representing the arrival time of agent i, we define the agent i's *discounted valuation* as $D_i \coloneqq V_i\, \xi(W_i)$, which represents how much agent i is willing to pay upon her arrival. As a result, whenever agent i arrives, she buys the item if and only if $D_i \geq p(W_i)$, *i.e.*, her discounted valuation is at least as large as the price offered by the mechanism.

We introduce the following additional notation. We denote by $I_{s,\tau} \coloneqq$

$[s, s+\tau] \subseteq [0,T]$ the time interval of length $\tau \in [0,T]$ starting from time $s \in [0, T-\tau]$. The number of agents arriving in $I_{s,\tau}$ is a random variable denoted by $N_{s,s+\tau}$. Given $\tau \in [0,T]$, the random variables $N_{s,s+\tau}$ are equally distributed for all $s \in [0, T-\tau]$, as the arrivals are generated by a Poisson process. For the sake of presentation, we omit s in $N_{s,s+\tau}$, denoting by N_τ the random variable of the number of agents arriving in any time interval of length τ, which follows a Poisson distribution with parameter $\lambda\tau$. [1] Thus, N_T is the random variable of the total number of agents arriving in the overall time period. In the following, we sometimes focus on the *linear* discount function, denoted as $\xi_{\text{lin}} : [0,T] \to [0,1]$ with $\xi_{\text{lin}}(t) \coloneqq 1 - \frac{t}{T}$. In this case, each agent i's discounted valuation is $D_i \coloneqq V_i \left(1 - \frac{W_i}{T}\right)$.

3.1.1 Performances of Posted-Price Mechanisms

Given a deterministic posted-price mechanism $\mathcal{M}$ defined by a price function $p_{\mathcal{M}} : [0,T] \to \mathbb{R}_+$, we denote by $\mathbb{E}_F[\mathcal{R}(\mathcal{M})]$ the expected revenue that the mechanism provides to the seller. The expectation is calculated with respect to both the Poisson arrivals and the distribution F of agents' initial valuations. We made explicit the dependence on F, as we will frequently refer to it along the chapter.

We adopt the perspective of competitive analysis and measure the performances of a mechanism $\mathcal{M}$ by comparing the seller's expected revenue with that of a benchmark mechanism $\mathcal{M}^\star$, which is optimal having knowledge of the distribution F. Notice that the benchmark has no information on the actual realizations of agents' initial valuations, but only on their distribution, whereas the mechanisms we propose operate having knowledge of their range only.

Our goal is to bound the performances of our mechanisms w.r.t. those of the benchmark $\mathcal{M}^\star$ by looking at the worst case over the set $\mathcal{F}$ of possible distributions F, *i.e.*, all those with support $[1,h]$. This is captured by the following:

Definition 3.1. *The* competitive ratio *of a deterministic posted-price mechanism $\mathcal{M}$ is defined as:*

$$\rho(\mathcal{M}) \coloneqq \min_{F \in \mathcal{F}} \rho_F(\mathcal{M}), \quad \textit{where } \rho_F(\mathcal{M}) \coloneqq \frac{\mathbb{E}_F[\mathcal{R}(\mathcal{M})]}{\mathbb{E}_F[\mathcal{R}(\mathcal{M}^\star)]}.$$

Moreover, we say that a mechanism is optimal *when its competitive ratio is the highest possible among all the deterministic posted-price mechanisms.*

[1] By definition of Poisson distribution, $\mathbb{P}\{N_\tau = j\} = \frac{(\lambda\tau)^j e^{-\lambda\tau}}{j!}$.

Notice that $\rho(\mathcal{M}) \in [0,1]$ and, for every possible distribution $F \in \mathcal{F}$, we are guaranteed that the seller's expected revenue $\mathbb{E}_F[\mathcal{R}(\mathcal{M})]$ provided by mechanisms $\mathcal{M}$ is at least a fraction $\rho(\mathcal{M})$ of that achieved by $\mathcal{M}^\star$, *i.e.*, it holds $\mathbb{E}_F[\mathcal{R}(\mathcal{M})] \geq \rho(\mathcal{M})\,\mathbb{E}_F[\mathcal{R}(\mathcal{M}^\star)]$.

As previously showed by Babaioff et al. (2017) in similar settings, we can safely restrict our analysis to mechanisms maintaining the bottom price for a non-negligible period of time. Indeed, in the case in which F places all the probability mass on 1, then any mechanism providing a non-null seller's expected revenue must set the minimum price during some time interval, otherwise no agent would buy the item.

Proposition 3.1. *Every deterministic posted-price mechanism $\mathcal{M}$ such that $\rho(\mathcal{M}) > 0$ must set the minimum price $p_{\mathcal{M}}(t) = \xi(t)$ for every t in a time interval of length $\tau > 0$.*

3.2 Identical Valuation Setting

We start studying the *identical valuation* (IV) setting, where all the agents share the same initial valuation $v \in [1,h]$ for the item, *i.e.*, it holds $V_i = v$ and $D_i = v\,\xi(W_i)$ for every agent i. The IV setting is a special case of the general random valuation model where one restricts the attention to distributions F placing all the probability mass on a single valuation in $[1,h]$. In the following, we adjust notation for expected revenues and competitive ratios accordingly, writing $\mathbb{E}_v[\mathcal{R}(\mathcal{M})]$ and $\rho_v(\mathcal{M})$ instead of $\mathbb{E}_F[\mathcal{R}(\mathcal{M})]$ and $\rho_F(\mathcal{M})$.

Our main result (Theorem 3.1) is to provide a deterministic posted-price mechanism, called $\mathcal{M}_{\mathrm{C}}$, which is optimal for the IV setting for every discount function ξ. We also study the specific case of a linear discount function ξ_{lin}, where we design an optimal mechanism $\mathcal{M}_{\mathrm{C,lin}}$ (Theorem 3.2) that enjoys an easily interpretable analytical description.

First, we describe the shape of the benchmark mechanism $\mathcal{M}^\star$ for the IV setting. Indeed, since $\mathcal{M}^\star$ knows the actual initial valuation v, its price function $p_{\mathcal{M}^\star} : [0,T] \to \mathbb{R}_+$ is such that $p_{\mathcal{M}^\star}(t) = v\,\xi(t)$ for $t \in [0,T]$. Therefore, we can compute the expected revenue of $\mathcal{M}^\star$ as follows:

$$\mathbb{E}_v\left[\mathcal{R}(\mathcal{M}^\star)\right] \coloneqq \int_0^T p_{\mathcal{M}^\star}(t)\,\lambda\,e^{-\lambda t}\,\mathrm{d}t = \int_0^T v\,\xi(t)\,\lambda\,e^{-\lambda t}\,\mathrm{d}t = v\,k^\star, \quad (3.1)$$

where $k^\star \coloneqq \int_0^T \xi(t)\,\lambda\, e^{-\lambda t}\,\mathrm{d}t$ does not depend on v, but only on the problem parameters T, λ, and the discount function ξ. Let us remark that the expected revenue of the benchmark $\mathcal{M}^\star$ defined in Equation (3.1) is expressed as a *linear* function of v.

3.2.1 Optimal Mechanism for a General Discount

We start proving two lemmas that highlight two crucial properties which characterize optimal posted-price mechanisms for the IV setting. Lemma 3.1 implies that the pricing strategy of an optimal mechanism must be such that the undiscounted price defined as $\frac{p(t)}{\xi(t)}$ is non-increasing in t, whereas Lemma 3.2 shows that any mechanism which always provides a constant fraction of the expected revenue of the benchmark, independently of the agents' initial valuation v, is an optimal mechanism.

Lemma 3.1. *In the IV setting, given any deterministic posted-price mechanism $\mathcal{M}$, there always exists a deterministic posted-price mechanism $\mathcal{M}'$ with undiscounted price $\frac{p_{\mathcal{M}'}(t)}{\xi(t)}$ non-increasing in t such that $\mathbb{E}_v[\mathcal{R}(\mathcal{M})] \leq \mathbb{E}_v[\mathcal{R}(\mathcal{M}')]$ for every possible agents' initial valuation $v \in [1, h]$.*

Proof. We only need to prove the result for mechanisms $\mathcal{M}$ whose undiscounted price $\frac{p_{\mathcal{M}}(t)}{\xi(t)}$ is *not* non-increasing in t, otherwise the statement of the lemma is trivially true. The main idea of the proof is to let the time period $[0, T]$ be evenly partitioned into time intervals of length τ such that the undiscounted price function of $\mathcal{M}$ is constant in each interval. This is w.lo.g. if we take $\tau \to 0$. Then, there must be two consecutive time intervals, namely $I_1 \coloneqq I_{s,\tau}$ and $I_2 \coloneqq I_{s+\tau,\tau}$ for some starting time $s \in [0, T-\tau]$, such that there exist $p_1 < p_2 \in [1, h]$ with $\frac{p_{\mathcal{M}}(t)}{\xi(t)} = p_1$ and $\frac{p_{\mathcal{M}}(t)}{\xi(t)} = p_2$ during I_1 and I_2, respectively (otherwise the undiscounted price would be non-increasing). Now, let us define a mechanism $\mathcal{M}'$ whose undiscounted price function is the same as that of $\mathcal{M}$, except for the fact that $\frac{p_{\mathcal{M}'}(t)}{\xi(t)} = p_2$ during I_1 and $\frac{p_{\mathcal{M}'}(t)}{\xi(t)} = p_1$ during I_2 (*i.e.*, intuitively, we exchange the values in the two intervals so as to make the undiscounted price non-increasing in that window of time).

We show that the expected revenue provided by $\mathcal{M}'$ is always greater than or equal to that achieved by $\mathcal{M}$, as long as $\tau \to 0$. In order to compare the expected revenues of the two mechanisms, it is sufficient to focus on the window of time $I_1 \cup I_2$, where their price functions differ. Given p_1 and p_2, we can partition the agents' valuations $v \in [1, h]$ into three different subsets, as follows:

- $v < p_1$, implying that $v\,\xi(t) < p_{\mathcal{M}}(t)$ and $v\,\xi(t) < p_{\mathcal{M}'}(t)$ for every time instant $t \in I_1 \cup I_2$;
- $p_1 \leq v \leq p_2$, implying that $p_{\mathcal{M}}(t) \leq v\,\xi(t) \leq p_{\mathcal{M}'}(t)$ for every time instant $t \in I_1$ and $p_{\mathcal{M}'}(t) \leq v\,\xi(t) \leq p_{\mathcal{M}}(t)$ for every time instant $t \in I_2$;
- $v > p_2$, implying that $v\,\xi(t) > p_{\mathcal{M}}(t)$ and $v\,\xi(t) > p_{\mathcal{M}'}(t)$ for every time instant $t \in I_1 \cup I_2$.

In the first case, $\mathbb{E}_v[\mathcal{R}(\mathcal{M})] - \mathbb{E}_v[\mathcal{R}(\mathcal{M}')] = 0$, since both $\mathcal{M}$ and $\mathcal{M}'$ achieve an expected revenue equal to 0 during the time window $I_1 \cup I_2$, given that the item is never sold in that window (as both $p_{\mathcal{M}}(t)$ and $p_{\mathcal{M}'}(t)$ are always higher than the agents' discounted valuation $v\,\xi(t)$). As for the second case, let us assume $p_1 < v < p_2$ (since the cases $v = p_1$ and $v = p_2$ are analogous). Then, $\mathcal{M}$ can sell the item only during the interval I_1, while $\mathcal{M}'$ can sell the item only during the other interval I_2. Thus, we have the following:

$$\mathbb{E}_v[\mathcal{R}(\mathcal{M})] - \mathbb{E}_v[\mathcal{R}(\mathcal{M}')] = \int_s^{s+\tau} p_1 \xi(t)\,\lambda e^{-\lambda t} dt - \int_{s+\tau}^{s+2\tau} p_1 \xi(t)\,\lambda e^{-\lambda t} dt,$$

which goes to 0 as long as $\tau \to 0$, given that ξ is continuous. Finally, in the third case, we can compute the difference between the expected revenues of the two mechanisms as follows:

$$\begin{aligned}
&\mathbb{E}_v[\mathcal{R}(\mathcal{M}) - \mathbb{E}_v[\mathcal{R}(\mathcal{M}')] = \\
&= \int_s^{s+\tau} p_1 \xi(t)\,\lambda e^{-\lambda t} dt + \int_{s+\tau}^{s+2\tau} p_2 \xi(t)\,\lambda e^{-\lambda t} dt \\
&\quad - \int_s^{s+\tau} p_2 \xi(t)\,\lambda e^{-\lambda t} dt - \int_{s+\tau}^{s+2\tau} p_1 \xi(t)\,\lambda e^{-\lambda t} dt = \\
&= (p_1 - p_2) \int_s^{s+\tau} \xi(t)\,\lambda e^{-\lambda t} dt - (p_1 - p_2) \int_{s+\tau}^{s+2\tau} \xi(t)\,\lambda e^{-\lambda t} dt = \\
&= (p_1 - p_2) \left[\int_s^{s+\tau} \xi(t)\,\lambda e^{-\lambda t} dt - \int_{s+\tau}^{s+2\tau} \xi(t)\,\lambda e^{-\lambda t} dt \right],
\end{aligned}$$

which is less than or equal to 0 as $\tau \to 0$, by continuity of ξ.

By re-iterating the procedure on all the pairs of consecutive infinitesimal intervals (since $\tau \to 0$) defined as I_1 and I_2 (each time using the last mechanism $\mathcal{M}'$ as the new $\mathcal{M}$), we can render the undiscounted price function non-increasing, obtaining a final mechanism $\mathcal{M}'$ such that $\mathbb{E}_v[\mathcal{R}(\mathcal{M})] \leq \mathbb{E}_v[\mathcal{R}(\mathcal{M}')]$ for every possible agents' valuation $v \in [1, h]$. □

Notice that, since ξ is continuous and non-increasing by definition, Lemma 3.1 also shows that there is always an optimal mechanism whose pricing strategy is non-increasing. Moreover, by recalling Proposition 3.1, we can conclude that any optimal mechanism must set the minimum price at the end of the overall time period, *i.e.*, during a time interval $[t_0, T] \subseteq [0, T]$ defined for some $t_0 \in [0, T)$. This result is exploited to prove the following lemma.

Lemma 3.2. *In the IV setting, let $\mathcal{M}$ be a deterministic posted-price mechanism whose pricing strategy $p_{\mathcal{M}}$ satisfies $p_{\mathcal{M}}(t) = \xi(t)$ for $t \in [t_0, T]$ with $t_0 \in [0, T)$. If the ratio $\rho_v(\mathcal{M}) = \frac{\mathbb{E}_v[\mathcal{R}(\mathcal{M})]}{\mathbb{E}_v[\mathcal{R}(\mathcal{M}^\star)]}$ for $\mathcal{M}$ does not depend on the agents' initial valuation v, then $\mathcal{M}$ is an optimal mechanism.*

Proof. By contradiction, suppose that $\mathcal{M}$ is *not* optimal, *i.e.*, there exists another deterministic posted-price mechanism $\mathcal{M}'$ such that $\rho(\mathcal{M}') > \rho(\mathcal{M})$. According to Proposition 3.1 and Lemma 3.1, $\mathcal{M}'$ must be defined by a pricing strategy $p_{\mathcal{M}'}$ such that the undiscounted price $\frac{p_{\mathcal{M}'}(t)}{\xi(t)}$ is non-increasing in t and the minimum price is selected for a time interval $[t'_0, T] \subseteq [0, T]$ having non-zero length (recall that $\rho_v(\mathcal{M}) > 0$ does not depend on v and $\rho(\mathcal{M}) = \min_{v \in [1,h]} \rho_v(\mathcal{M})$).

Case $t'_0 \geq t_0$. Let us consider the valuation $v = 1$. Then, we have that the expected revenue of mechanism $\mathcal{M}$ is $\mathbb{E}_v[\mathcal{R}(\mathcal{M})] = \int_{t_0}^{T} \xi(t)\, \lambda e^{-\lambda t} \mathrm{d}t$ (accounting for the case in which an agent arrives at $t \geq t_0$ and buys the item at price $\xi(t)$), which is greater than or equal to the expected revenue of mechanism $\mathcal{M}'$, defined as $\mathbb{E}_v[\mathcal{R}(\mathcal{M}')] = \int_{t'_0}^{T} \xi(t)\, \lambda e^{-\lambda t} \mathrm{d}t$. Intuitively, $\mathbb{E}_v[\mathcal{R}(\mathcal{M})] \geq \mathbb{E}_v[\mathcal{R}(\mathcal{M}')]$ since $\mathcal{M}'$ posts the minimum price for a period of time shorter than that of $\mathcal{M}$. Therefore, it holds $\rho(\mathcal{M}') \leq \rho_v(\mathcal{M}') \leq \rho_v(\mathcal{M}) \leq \rho(\mathcal{M})$, which is a contradiction.

Case $t'_0 < t_0$. First, suppose that there exists a time instant $t' \in [0, t'_0]$ defined as $t' \coloneqq \sup\{t \in [0, t'_0] \mid p_{\mathcal{M}}(t) < p_{\mathcal{M}'}(t)\}$, *i.e.*, the last time instant in which $p_{\mathcal{M}}(t)$ changes from being less than $p_{\mathcal{M}'}(t)$ to being larger than or equal to $p_{\mathcal{M}'}(t)$. Clearly, it holds $p_{\mathcal{M}}(t) \geq p_{\mathcal{M}'}(t)$ for every $t \in [0, T] : t > t'$. Moreover, let us consider the agents' valuation $v \in [1, h]$ such that $v\,\xi(t') = p_{\mathcal{M}}(t')$ and focus on the case in which $p_{\mathcal{M}}(t) = p_{\mathcal{M}'}(t)$ (as the other cases are analogous). Notice that, for every time instant $t \leq t'$, mechanism $\mathcal{M}'$ cannot sell the item, since, by using Lemma 3.1, we get:

$$v\,\xi(t) < v p_{\mathcal{M}'}(t) \frac{\xi(t')}{p_{\mathcal{M}'}(t')} = v p_{\mathcal{M}'}(t) \frac{\xi(t')}{p_{\mathcal{M}}(t')} \leq v p_{\mathcal{M}'}(t) \frac{\xi(t')}{v\,\xi(t')} \leq p_{\mathcal{M}'}(t).$$

Additionally, with an analogous reasoning we can show that, for all the times $t \in [0, T] : t > t'$, both mechanisms may sell the item, but the price

posted by $\mathcal{M}'$ is always less than or equal to that chosen by $\mathcal{M}$, with a non-empty time interval in which the former is strictly less than the latter (as $t'_0 < t_0$). Thus, in this case, it holds $\rho_v(\mathcal{M}) > \rho_v(\mathcal{M}')$, which implies that $\rho(\mathcal{M}') < \rho(\mathcal{M})$, a contradiction. Finally, it remains to analyze the case in which a time instant t' defined above does *not* exist. Since the undiscounted price functions are non-increasing by Lemma 3.2 and $t'_0 < t_0$, it must be the case that there is no intersection point between the two functions. Hence, it must be $p_{\mathcal{M}}(t) > p_{\mathcal{M}'}(t)$ for all $t \in [0, t_0]$, which implies that $\rho(\mathcal{M}') < \rho(\mathcal{M})$ by taking $v = h$. This leads to a contradiction. □

By Lemma 3.2, in order to find an optimal mechanism for the IV setting, we can restrict the attention to mechanisms $\mathcal{M}$ whose ratios $\rho_v(\mathcal{M})$ do not depend on the initial valuation v. Therefore, since the expected revenue of the benchmark $\mathcal{M}^\star$ is a linear function of v (see Equation (3.1)), we can search for an optimal mechanism among those having an expected revenue which linearly depends on v. This crucial observation allows us to design the optimal mechanism $\mathcal{M}_{\mathrm{c}}$ in Theorem 3.1 by leveraging the condition $\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\mathrm{c}})] = k\,v$ for every $v \in [1, h]$, with k being a suitably defined constant independent of v. The key insight that allows us to derive an expression for $\mathcal{M}_{\mathrm{c}}$ is that we can always find the desired pricing strategy $p_{\mathcal{M}_{\mathrm{c}}}$ among the continuous price functions such that $\frac{p_{\mathcal{M}_{\mathrm{c}}}(t)}{\xi(t)}$ is non-increasing in $t \in [0, T)$. Intuitively, using Lemma 3.1, we can always express the expected revenue $\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\mathrm{c}})]$ as a function of the time $t^* \coloneqq \sup\{t \in [0, t_0] \mid p_{\mathcal{M}_{\mathrm{c}}}(t) > v\,\xi(t)\}$, which is the first time in which $p_{\mathcal{M}_{\mathrm{c}}}$ intersects $v\,\xi(t)$. The reason is that it holds $p_{\mathcal{M}_{\mathrm{c}}}(t) \leq v\,\xi(t)$ if and only if $t \geq t^*$, and, thus, only agents arriving after t^* are willing to buy the item. By using the relation among $\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\mathrm{c}})]$ and t^*, we can find the desired pricing strategy $p_{\mathcal{M}_{\mathrm{c}}}$ as a solution to a suitably defined differential equation. This leads to the following theorem.

Theorem 3.1. *In the IV setting, there exists an optimal deterministic posted-price mechanism $\mathcal{M}_{\mathrm{c}}$ whose pricing strategy $p_{\mathcal{M}_{\mathrm{c}}}$ is defined as follows:*

$$p_{\mathcal{M}_{\mathrm{c}}}(t) \coloneqq \begin{cases} a\, e^{\int b(t)\mathrm{d}t} & \text{if } t \in [0, t_0) \\ \xi(t) & \text{if } t \in [t_0, T] \end{cases},$$

where b is a function such that $b(t) \coloneqq \lambda - \frac{\lambda}{k\zeta(t)} - \frac{\zeta'(t)}{\zeta(t)}$ with $\zeta(t) \coloneqq \frac{1}{\xi(t)}$, whereas a, k, t_0 are suitably defined constants that do not depend on the agents' initial valuation v.

Proof. By Lemma 3.2 and using $\rho_v(\mathcal{M}_{\mathrm{c}}) = \frac{\mathbb{E}_v[\mathcal{R}(\mathcal{M})]}{\mathbb{E}_v[\mathcal{R}(\mathcal{M}^\star)]}$, it is sufficient to search for an optimal mechanism $\mathcal{M}_{\mathrm{c}}$ whose pricing strategy $p_{\mathcal{M}_{\mathrm{c}}}$ is such

that the expected revenue of the mechanism is linearly dependent in v, *i.e.*, for every valuation $v \in [1, h]$, it must be the case that:

$$\mathbb{E}_v \left[\mathcal{R}(\mathcal{M}_{\mathrm{C}})\right] = kv,$$

where $k > 0$ is a suitably defined constant that does depend on v. In the following, for the ease of presentation, we omit the index $\mathcal{M}_{\mathrm{C}}$ from $p_{\mathcal{M}_{\mathrm{C}}}$ as the mechanism is clear from the context.

From Proposition 3.1, there must be a $t_0 \in [0, T)$ such that $p(t) = \xi(t)$ for every $t \in [t_0, T]$, otherwise $\rho_v(\mathcal{M}_{\mathrm{C}}) = 0$ for the valuation $v = 1$. Thus, it remains to define $p(t)$ for $t \in [0, t_0)$.

For any valuation $v \in [1, h]$, by letting $t^* \coloneqq \sup\{t \in [0, t_0] \mid p(t) > v\,\xi(t)\}$, we can express the expected revenue of the mechanism $\mathcal{M}_{\mathrm{C}}$ as a function of t^*. First, notice that, it holds $p(t^*) = v\,\xi(t^*)$. Moreover, by using Lemma 3.1, it must be the case that $p(t) > v\,\xi(t)$ for every $t < t^*$, since:

$$v\,\xi(t) < v\,p(t)\frac{\xi(t^*)}{p(t^*)} = v\,p(t)\frac{\xi(t^*)}{v\,\xi(t^*)} = p(t).$$

As a result, the item is never sold before time t^*, which allows us to write the following:

$$\mathbb{E}_v \left[\mathcal{R}(\mathcal{M}_{\mathrm{C}})\right] = e^{\lambda t^*} \int_{t^*}^{t_0} p(t)\,\lambda e^{-\lambda t} \mathrm{d}t + e^{\lambda t^*} \int_{t_0}^{T} \xi(t)\,\lambda e^{-\lambda t} \mathrm{d}t.$$

Thus, since we want $\mathbb{E}_v \left[\mathcal{R}(\mathcal{M}_{\mathrm{C}})\right] = kv$, by using $v = \frac{p(t^*)}{\xi(t^*)}$ and letting $\zeta(t) \coloneqq \frac{1}{\xi(t)}$, we get:

$$e^{\lambda t^*} \int_{t^*}^{t_0} p(t)\,\lambda e^{-\lambda t} \mathrm{d}t + e^{\lambda t^*} \int_{t_0}^{T} \xi(t)\,\lambda e^{-\lambda t} \mathrm{d}t = k\zeta(t^*) p\left(t^*\right). \tag{3.2}$$

By deriving the left-hand side of Equation (3.2) with respect to t^*, we get:

$$\begin{aligned}
&\frac{\mathrm{d}\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\mathrm{C}})]}{\mathrm{d}t^*} = \\
&= e^{\lambda t^*} \frac{\mathrm{d}G(t^*)}{\mathrm{d}t^*} + \lambda \left[e^{\lambda t^*} \int_{t^*}^{t_0} p(t)\,\lambda e^{-\lambda t} \mathrm{d}t + \lambda e^{\lambda t^*} \int_{t_0}^{T} \xi(t)\,\lambda e^{-\lambda t} \mathrm{d}t \right] = \\
&= -\lambda p(t^*) + \lambda k \zeta(t^*) p(t^*)
\end{aligned}$$

where $G\left(t^*\right) \coloneqq \int_{t^*}^{t_0} p(t) \lambda e^{-\lambda t} \mathrm{d}t = \int_{t_0}^{t^*} -p(t) \lambda e^{-\lambda t} \mathrm{d}t = \int_{t_0}^{t^*} g(t) \mathrm{d}t$, with $g(t) \coloneqq -p(t) \lambda e^{-\lambda t}$. By applying the fundamental theorem of calculus, we have that $\frac{\mathrm{d}G(t^*)}{\mathrm{d}t^*} = g(t^*) = -p(t^*) \lambda e^{-\lambda t^*}$. Thus, the last equality is

readily obtained by noticing that the term in the squared brackets is exactly equal to the expected revenue $\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\text{C}})]$, which, in turn, must be equal to $k\zeta(t^*)p(t^*)$. Furthermore, by deriving the right-hand side of Equation (3.2) with respect to t^*, we get:

$$\frac{\mathrm{d}}{\mathrm{d}t^*}\left[k\zeta(t^*)p(t^*)\right] = k\zeta'(t^*)p(t^*) + k\zeta(t^*)p'(t^*).$$

By equating the derivatives of the two sides of Equation (3.2), we get the following differential equation:

$$p'\left(t^*\right) = \left[\lambda - \frac{\lambda}{k\zeta'\left(t^*\right)} - \frac{\zeta'\left(t^*\right)}{\zeta\left(t^*\right)}\right] p\left(t^*\right) \tag{3.3}$$

By solving Equation (3.3) for $p(t)$, we obtain the function:

$$p(t) = a\, e^{\int\left[\lambda - \frac{\lambda}{k\zeta'(t)} - \frac{\zeta'(t)}{\zeta(t)}\right]dt},$$

and, from the boundaries conditions $p(0) = h$ and $p(t_0) = \xi(t_0)$, we can derive constants a and k. Notice that the condition $p(0) = h$ can be derived from the fact that, if $p(0) < h$, then the expected revenue $\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\text{C}})]$ is the same for all the valuations $v \in [1, h]$ such that $p(0) \leq v \leq h$, which is not possible since we want that $\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\text{C}})]$ linearly depends on v.

We recall that $\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\text{C}})] = kv$ for all $v \in [1, h]$. Thus, we can use this in order to find t_0 as a function of the problem parameters λ, T, h, and function ξ. Using $v = 1$, we get:

$$\int_0^{T-t_0} \xi(t)\lambda e^{-\lambda t} dt = k, \tag{3.4}$$

which gives t_0 after replacing k with the expression we got from the boundaries conditions. □

As a byproduct of the proof of Theorem 3.1, we also get an expression for the competitive ratio of the mechanism $\mathcal{M}_{\text{C}}$, as stated by the following corollary.

Corollary 3.1. *In the IV setting, mechanism $\mathcal{M}_{\text{C}}$ achieves:*

$$\rho(\mathcal{M}_{\text{C}}) = \frac{\int_0^{T-t_0} \xi(t)\,\lambda\, e^{-\lambda t}\,\mathrm{d}t}{\int_0^{T} \xi(t)\,\lambda\, e^{-\lambda t}\,\mathrm{d}t}.$$

Proof. Let us recall that, from the proof of Theorem 3.1, $\mathcal{M}_{\mathrm{c}}$ is characterized by the same ratio $\rho_v(\mathcal{M}_{\mathrm{c}})$ for all $v \in [1, h]$. Hence, we can calculate the competitive ratio by taking $v = 1$:

$$\rho(\mathcal{M}_{\mathrm{c}}) = \frac{\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\mathrm{c}})]}{\mathbb{E}_v[\mathcal{R}(\mathcal{M}^\star)]} = \frac{k}{k^\star} = \frac{\int_0^{T-t_0} \xi(t)\lambda e^{-\lambda t}\mathrm{d}t}{\int_0^T \xi(t)\lambda e^{-\lambda t}\mathrm{d}t}, \tag{3.5}$$

where we used Equation (3.1) and Equation (3.4) from the proof of Theorem 3.1. □

3.2.2 Optimal Mechanism for a Linear Discount

The pricing strategy $p_{\mathcal{M}_{\mathrm{c}}}$ of the optimal mechanism defined in Theorem 3.1 still depends on some parameters, namely a, k, and t_0, which do not admit an easy analytical formula for a general discount function ξ. Nevertheless, they can be expressed analytically if we restrict the attention to functions ξ having a particular shape. In the following Theorem 3.2 and Corollary 3.2, we analyze the case of a linear discount function ξ_{lin}, defining an optimal mechanism $\mathcal{M}_{\mathrm{c,lin}}$ for such setting.

Theorem 3.2. *In the IV setting with linear discount function ξ_{lin}, there exists an optimal deterministic posted-price mechanism $\mathcal{M}_{\mathrm{c,lin}}$ whose pricing strategy $p_{\mathcal{M}_{\mathrm{c,lin}}}$ is defined as:*

$$p_{\mathcal{M}_{\mathrm{c,lin}}}(t) \coloneqq \begin{cases} h\,\left(1-\frac{t}{T}\right)\, e^{\lambda\left(1-\frac{1}{k}\right)t+\frac{\lambda}{2kT}t^2} & \text{if } t \in [0, t_0) \\ 1-\frac{t}{T} & \text{if } t \in [t_0, T] \end{cases},$$

where $k \coloneqq \lambda\, t_0\, \frac{2\,T-t_0}{2\,T(\lambda\, t_0+\ln h)}$ and the time $t_0 \in [0, T)$ is defined as the unique positive real root of the following equation: $1-\frac{1}{\lambda T}\left(1+\lambda\, t_0 - e^{-\lambda(T-t_0)}\right) = k$.

Proof. We follow the line of the proof of Theorem 3.1, *i.e.*, we look for a mechanism $\mathcal{M}_{\mathrm{c}}$ such that $\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\mathrm{c}})] = kv$ for every $v \in [1, h]$, where $k > 0$ is suitably defined constant that does not depend on v. For the ease of presentation, we omit the subscript $\mathcal{M}_{\mathrm{c}}$ from the pricing strategy $p_{\mathcal{M}_{\mathrm{c}}}$.

Let us fix $v \in [1, h]$. By defining t^* as in the proof of Theorem 3.1, since in this case the discount is $\xi_{\text{lin}}(t) = 1 - \frac{t}{T}$ for $t \in [0, T]$, we have $p(t^*) = v\,\left(1-\frac{t^*}{T}\right)$, which allows us to write the following:

$$e^{\lambda t^*}\int_{t^*}^{t_0} p(t)\,\lambda e^{-\lambda t}\mathrm{d}t + e^{\lambda t^*}\int_{t_0}^{T}\left(1-\frac{t}{T}\right)\lambda e^{-\lambda t}\mathrm{d}t = k\frac{T}{T-t^*}p\left(t^*\right), \tag{3.6}$$

where the left-hand side is the expected revenue $\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\mathrm{C}})]$ and the right-hand side is kv. By deriving with respect to t^* the left-hand side of the Equation (3.6), we get:

$$\frac{\mathrm{d}\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\mathrm{C}})]}{\mathrm{d}t^*} =$$
$$= e^{\lambda t^*}\frac{\mathrm{d}G(t^*)}{\mathrm{d}t^*} + \lambda\left[e^{\lambda t^*}\int_{t^*}^{t_0} p(t)\lambda e^{-\lambda t}\mathrm{d}t + \lambda e^{\lambda t^*}\int_{t_0}^{T}\left(1-\frac{t}{T}\right)\lambda e^{-\lambda t}\mathrm{d}t\right] =$$
$$= -\lambda p(t^*) + \lambda k\frac{T}{T-t^*}p(t^*),$$

where $G\left(t^*\right)$ is defined as in the proof of Theorem 3.1. Now, we derive the right-hand side of Equation (3.6) with respect to t^*:

$$\frac{\mathrm{d}}{\mathrm{d}t^*}\left(\frac{kT}{T-t^*}p\left(t^*\right)\right) = \frac{kT}{T-t^*}p'\left(t^*\right) + \frac{kT}{\left(T-t^*\right)^2}p\left(t^*\right).$$

By equating the derivatives of the two sides of Equation (3.6), we get the following differential equation:

$$p'\left(t^*\right) = \left[\lambda - \frac{\lambda\left(T-t^*\right)}{kT} - \frac{1}{T-t^*}\right]p\left(t^*\right). \tag{3.7}$$

After solving Equation (3.7) for $p(t)$, we obtain the general solution:

$$p(t) = a\, e^{\int\left[\lambda - \frac{\lambda(T-t)}{kT} - \frac{1}{T-t}\right]\mathrm{d}t} = a\, e^{\lambda\left(1-\frac{1}{k}\right)t + \frac{\lambda}{2kT}t^2 + \ln(T-t)},$$

where, using boundary conditions $p(0) = h$ and $p(t_0) = 1 - \frac{t_0}{T}$, we can derive the expressions $a \coloneqq \frac{h}{T}$ and $k \coloneqq \lambda t_0 \frac{2T-t_0}{2T(\lambda t_0 + \ln(h))}$.

Since $\mathbb{E}_v[\mathcal{R}[\mathcal{M}_{\mathrm{C}}]] = kv$ for all $v \in [1, h]$, we can use the equation in order to define t_0 with respect to the problem parameters λ, T and h. For $v = 1$:

$$\int_0^{T-t_0}\left(1-\frac{t}{T}\right)\lambda e^{-\lambda t}\mathrm{d}t = 1 - \frac{1}{\lambda T}\left(1 + \lambda t_0 - e^{-\lambda(T-t_0)}\right) = k, \tag{3.8}$$

and, by replacing k with the expression we got from the boundary conditions, we obtain:

$$1 - \frac{1}{\lambda\, T}\left(1 + \lambda\, t_0 - e^{-\lambda\,(T-t_0)}\right) = \lambda\, t_0 \frac{2\, T - t_0}{2\, T\,(\lambda\, t_0 + \ln(h))} \tag{3.9}$$

Finally, we can define t_0 as the unique positive real root of Equation (3.9). In particular, it is easy to show that Equation (3.9) always admits a positive

	$T \to \infty$	$\lambda \to \infty$	$h \to \infty$
t_0	$\Theta(\sqrt{T})$	$\Theta(\sqrt{T/\lambda})$	$\Theta(T)$
$\rho(\mathcal{M}_{\text{C,lin}})$	$\Theta\left(1-\frac{1}{\sqrt{T}}\right)$	$\Theta\left(1-\frac{1}{\sqrt{\lambda}}\right)$	$\Theta\left(\frac{1}{\log^2(h)}\right)$
$\lim \rho(\mathcal{M}_{\text{C,lin}})$	1	1	0

Table 3.1: *Values of t_0 and $\rho(\mathcal{M}_{\text{C,lin}})$ as T, λ, h go to infinity.*

real root in the range $(0, T)$. Indeed, we call $q(x) = \lambda x \frac{2T-x}{2T(\lambda x + \ln(h))} - 1 + \frac{1}{\lambda T}\left(1 + \lambda x - e^{-\lambda(T-x)}\right)$. We observe that $q(x)$ is continuous on the interval $[0, T]$ and that $q(T) > 0$ and $q(0) < 0$, therefore, for Bolzano's theorem, there exists al least a $t_0 \in (0, T)$ such that $q(t_0) = 0$. The uniqueness can be derived as consequence of Lemma 3.2. □

The prices posted by $\mathcal{M}_{\text{C,lin}}$ decrease as a linearly discounted exponential function until $t = t_0$, starting, at time $t = 0$, by setting the price equal to the maximum agents' initial valuations h. Then, during the time interval $[t_0, T]$, the price function linearly decreases and equals zero in $t = T$.

Corollary 3.2. *In the IV setting with linear discount function ξ_{lin}, $\mathcal{M}_{\text{C,lin}}$ achieves a competitive ratio:*

$$\rho(\mathcal{M}_{\text{C,lin}}) = \frac{1 - \frac{1}{\lambda T}\left(1 + \lambda t_0 - e^{-\lambda(T-t_0)}\right)}{1 - \frac{1}{\lambda T}\left(1 - e^{-\lambda T}\right)}.$$

Proof. We can calculate it by taking $v = 1$:

$$\begin{aligned}\rho(\mathcal{M}_{\text{C}}) = \frac{\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\text{C}})]}{\mathbb{E}_v[\mathcal{R}(\mathcal{M}^\star)]} = \frac{k}{k^\star} &= \frac{\int_0^{T-t_0}\left(1-\frac{t}{T}\right)\lambda e^{-\lambda t}\mathrm{d}t}{\int_0^{T}\left(1-\frac{t}{T}\right)\lambda e^{-\lambda t}\mathrm{d}t} \\ &= \frac{1 - \frac{1}{\lambda T}\left(1 + \lambda t_0 - e^{-\lambda(T-t_0)}\right)}{1 - \frac{1}{\lambda T}\left(1 - e^{-\lambda T}\right)},\end{aligned}$$

where we used Equation (3.1) and Equation (3.8) from the proof of Theorem 3.2. □

The asymptotic values of t_0 and $\rho(\mathcal{M}_{\text{C,lin}})$ as T, λ, h go to infinity are in Table 3.1. In particular, $\rho(\mathcal{M}_{\text{C,lin}})$ goes asymptotically to 1 as λ or T increases. This corresponds to having an infinite number of agents and, thus, selling the item with certainty. Instead, $\rho(\mathcal{M}_{\text{C,lin}})$ decreases as h increases, going asymptotically to 0 as $\frac{1}{\log^2(h)}$. The range $[1, h]$ represents the degree

of uncertainty that the mechanism has on the agents' valuation. Therefore, $\rho(\mathcal{M}_{\text{C,lin}})$ decreases as the uncertainty increases and it cannot be lower bounded by any strictly positive constant if no finite upper bound on h is known (*i.e.*, when $h \to +\infty$). However, the dependency of $\rho(\mathcal{M}_{\text{C,lin}})$ on the degree of uncertainty is logarithmic. Instead, notice that a trivial mechanism setting the price equal to $\xi_{\text{lin}}(t)$ for $t \in [0, T]$ would have a competitive ratio of $\frac{1}{h}$, which depends linearly on the degree of uncertainty. More details on the analysis of t_0 and $\rho(\mathcal{M}_{\text{C,lin}})$ as T, λ, h vary are provided in the following paragraph.

Analysis of the Competitive Ratio $\rho(\mathcal{M}_{\text{C,lin}})$ From Corollary 3.2, we know that, in the IV setting with linear discount function ξ_{lin}, mechanism $\mathcal{M}_{\text{C,lin}}$ achieves a competitive ratio:

$$\rho(\mathcal{M}_{\text{C,lin}}) = \frac{1 - \frac{1}{\lambda T}\left(1 + \lambda t_0 - e^{-\lambda(T-t_0)}\right)}{1 - \frac{1}{\lambda T}\left(1 - e^{-\lambda T}\right)},$$

where $t_0 \in [0, T)$ is defined in Theorem 3.2 as the unique positive real root of the equation:

$$\lambda\, t_0 \,\frac{2\,T - t_0}{2\,T\,(\lambda\, t_0 + \ln h)} = 1 - \frac{1}{\lambda T}\left(1 + \lambda\, t_0 - e^{-\lambda(T-t_0)}\right). \tag{3.10}$$

We analyze the behavior of the competitive ratio $\rho(\mathcal{M}_{\text{C,lin}})$ when the problem parameters h, λ and T vary. We summarize our results in Table 3.1. By using Equation (3.10), we can conclude that t_0 is asymptotically equivalent to $\sqrt{T}$ when $T \to \infty$. Then, the limit of the competitive ratio is:

$$\lim_{T\to\infty} \rho(\mathcal{M}_{\text{C,lin}}) = \lim_{T\to\infty} 1 - \frac{t_0}{T} = \lim_{T\to\infty} 1 - \frac{1}{\sqrt{T}} = 1.$$

Similarly, t_0 is asymptotically equivalent to $\sqrt{\frac{T}{\lambda}}$ when $\lambda \to \infty$. Thus:

$$\lim_{\lambda\to\infty} \rho(\mathcal{M}_{\text{C,lin}}) = \lim_{\lambda\to\infty} 1 - \frac{t_0}{T} = \lim_{\lambda\to\infty} 1 - \frac{1}{\sqrt{T\lambda}} = 1.$$

Moreover, it is easy to see that $t_0 \to T$ when $h \to \infty$. Indeed, in this case we have that t_0 is the unique positive real root of the equation:

$$1 - \frac{1}{\lambda T}\left(1 + \lambda\, t_0 - e^{-\lambda(T-t_0)}\right) = 0.$$

Since $t_0 \to T$, the limit of the competitive ratio is:

$$\lim_{h\to\infty} \rho(\mathcal{M}_{\text{C,lin}}) = \frac{1 - \frac{1}{\lambda T}\left(1 + \lambda T - e^{-\lambda(T-T)}\right)}{1 - \frac{1}{\lambda T}\left(1 - e^{-\lambda T}\right)} = 0.$$

Therefore, having a valuation function with finite support is fundamental in order to achieve a certain fraction of the expected revenue of an optimal mechanism. There are no guarantees for valuation functions with unbounded support. In the following we analyze how $\rho(\mathcal{M}_{\text{C,lin}})$ goes to 0. We first observe that $\rho(\mathcal{M}_{\text{C,lin}})$ is proportional to $\frac{\lambda(T-t_0)^2}{2T} + o(T-t_0)^2$ when $h \to \infty$. Indeed, the numerator of $\rho(\mathcal{M}_{\text{C,lin}})$ depends on h through t_0:

$$1-\frac{1}{\lambda T}\left(1+\lambda t_0 - e^{-\lambda(T-t_0)}\right) = \frac{z}{T} - \frac{1}{\lambda T}(1-e^{-\lambda z}) = \frac{\lambda(T-t_0)^2}{2T} + o(T-t_0)^2,$$

where $z \coloneqq T - t_0 \to 0$ as $h \to \infty$, and the Taylor series $e^{-\lambda z} = 1 - \lambda z + \frac{\lambda^2 z^2}{2} + o(z^2)$ is used to expand function $e^{-\lambda z}$ at $t_0 = T$. Notice that the competitive ratio is decreasing in t_0. We compute $\bar{t}_0$, which is an upper bound for t_0, by solving Equation (3.10) with the exponential term $e^{-\lambda(T-t_0)}$ substituted by parameter a ε. We impose ε and $e^{-\lambda(T-t_0)}$ to have the same domain, hence $\varepsilon \in (0,1)$. Thus, we obtain the following equation:

$$\lambda\, x\, \frac{2\,T - x}{2\,T\,(\lambda\, x + \ln h)} = 1 - \frac{1}{\lambda T}\,(1 + \lambda\, x - \varepsilon)\,. \tag{3.11}$$

The solution

$$x = \frac{\sqrt{2\lambda T \ln(h) + \ln^2 h + \varepsilon^2 - 2\varepsilon + 1} - \ln h + \varepsilon - 1}{\lambda}$$

is increasing in ε, being its first partial derivative positive for all $\varepsilon \in (0,1)$:

$$\frac{\partial x}{\partial \varepsilon} = \frac{\varepsilon - 1}{\lambda\sqrt{2\lambda T \ln(h) + \ln^2 h + (\varepsilon-1)^2}} + \frac{1}{\lambda}.$$

Hence, by setting $\varepsilon = 1$, we get the following upper bound on t_0:

$$\bar{t}_0 \coloneqq \frac{-\ln h + \sqrt{2\lambda T \ln(h) + \ln^2 h}}{\lambda} = \frac{2T\ln(h)}{\sqrt{2\lambda T \ln(h) + \ln^2 h} + \ln h}.$$

Notice that $T - \bar{t}_0$ is a lower bound for $T - t_0$. By asymptotic analysis, as $h \to \infty$ we have:

$$T - \bar{t}_0 = \frac{2\lambda T^2 \ln h}{2\ln^2 h + 2\lambda T \ln h + 2\ln h\sqrt{2\lambda T \ln h + \ln^2 h}} \sim C_1 \frac{1}{\ln(h)}$$

where C_1 is constant with respect to h and depends on parameters λ and T. Hence, as $h \to \infty$:

$$\rho(\mathcal{M}_{\text{C,lin}}) \sim \frac{\lambda(T-t_0)^2}{2T - \frac{2}{\lambda}(1-e^{-\lambda T})} \geq \frac{\lambda(T-\bar{t}_0)^2}{2T - \frac{2}{\lambda}(1-e^{-\lambda T})} \sim C_2 \frac{1}{\ln^2(h)},$$

where C_2 is a constant with respect to h and depends on parameters λ and T. We conclude that, as $h \to \infty$, the competitive ratio $\rho(\mathcal{M}_{\text{C,lin}})$ converges to 0 slower than or the same as the function $\frac{1}{\log^2(h)}$.

3.3 Random Valuation Setting

We now switch to the *random valuation* (RV) setting, where agents' initial valuations V_i are i.i.d. random variables defined by a cumulative distribution function F with support $[1, h]$. We focus on distributions F satisfying the *monotone hazard rate* (MHR) condition. Formally, a distribution F is MHR if the hazard rate $H(x) \coloneqq \frac{f(x)}{1-F(x)}$ is non-decreasing in x. This assumption is common when studying posted-price mechanisms that operate without knowing the shape of the distribution of valuations (see (Babaioff et al., 2015, 2017)) and many distributions used in practice satisfy it (such as, *e.g.*, uniform, normal, and exponential distributions). Moreover, the MHR condition is necessary for proving our main results (Theorems 3.3 and 3.4). Indeed, when the family of possible distributions is unrestricted, one cannot design posted-price mechanisms guaranteeing a constant fraction of the revenue of $\mathcal{M}^\star$ independently of the distribution F, as shown by Babaioff et al. (2017) for the easier setting in which agents do not arrive stochastically.

Auxiliary Definitions and Results We introduce the random variable $X_{\lambda\tau}$ as the maximum initial valuation of agents arriving in an interval of length $\tau \in (0, T]$. Formally:

$$X_{\lambda\tau} \coloneqq \max_{i \in \{1,\dots,N_\tau\}} V_i.\ ^2$$

$X_{\lambda\tau}$ is the first order statistic of N_τ samples drawn from F and, since agents' arrivals are governed by a Poisson process, its cumulative distribution function $F_{X_{\lambda\tau}}$ is defined as:

[2] In the definition of $X_{\lambda\tau}$ and $Y_{s,\lambda\tau}$, overloading the notation, we assume that the agents arriving in the considered time interval of length τ are labeled from 1 to N_τ according to their order. Their actual labels referred to the overall period $[0, T]$ may be different.

$$F_{X_{\lambda\tau}}(x) \coloneqq \sum_{j=1}^{\infty} \mathbb{P}\left\{N_\tau = j\right\} F_{X_{\lambda\tau}|N_\tau=j}(x) = \sum_{j=1}^{\infty} \frac{(\lambda\tau)^j e^{-\lambda\tau}}{j!} \left[F(x)\right]^j = e^{-\lambda\tau(1-F(x))}.$$

We also define $Y_{s,\lambda\tau}$ as the random variable representing the maximum discounted valuation of agents arriving in an interval $I_{s,\tau}$ of length $\tau \in (0, T]$ starting at $s \in [0, T-\tau]$:

$$Y_{s,\lambda\tau} \coloneqq \max_{i\in\{1,\dots,N_\tau\}} D_i.$$

The cumulative distribution function $F_{Y_{s,\lambda\tau}}$ of $Y_{s,\lambda\tau}$ is:

$$F_{Y_{s,\lambda\tau}}(x) \coloneqq \sum_{j=1}^{\infty} \mathbb{P}\left\{N_\tau = j\right\} F_{Y_{s,\lambda\tau}|N_\tau=j}(x) = \sum_{j=1}^{\infty} \frac{(\lambda\tau)^j e^{-\lambda\tau}}{j!} F_{Y_{s,\lambda\tau}|N_\tau=j}(x),$$

where $F_{Y_{s,\lambda\tau}|N_\tau=j}$ is the cumulative distribution function of $Y_{s,\lambda\tau}$ conditioned on the event $N_\tau = j$. Let us remark that, by definition, $F_{Y_{s,\lambda\tau}}$ depends on distribution F. In the following, we also let $Y_{\lambda T} \coloneqq Y_{0,\lambda T}$ be the random variable representing the maximum discounted valuation of agents arriving in the overall time period $[0, T]$. In Appendix A.1.1, for the specific case of a linear discount function, we show how to exploit some useful properties of Poisson processes so as to find an analytical expression for $F_{Y_{\lambda T}}$. In particular, by letting $Z \coloneqq V\,U$, where V and U are independent random variables distributed according to F and $\mathcal{U}(0,1)$, respectively, we obtain:

$$F_{Y_{\lambda T}}(x) \coloneqq \sum_{j=1}^{\infty} \frac{(\lambda T)^j e^{-\lambda T}}{j!} \left[F_Z(x)\right]^j,$$

where

$$F_Z(x) \coloneqq \begin{cases} x \int_1^h \frac{1}{z} f(z)\mathrm{d}z & \text{if } x \in [0,1) \\ F(x) + x \int_x^h \frac{1}{z} f(z)\mathrm{d}z & \text{if } x \in [1,h] \end{cases}.$$

3.3.1 Bounding Competitive Ratio in the RV Setting

We show that mechanism $\mathcal{M}_{\mathrm{C}}$ (see definition in Theorem 3.1), which is optimal in the IV setting, provides good performances also in the RV setting.

Our main result (Theorem 3.3) is a lower bound on the competitive ratio of the mechanism, which is obtained by showing that $\mathcal{M}_{\mathrm{C}}$ always provides at least a constant fraction of the seller's expected revenue achieved by the benchmark $\mathcal{M}^{\star}$, independently of the distribution of agents' initial valuations F.[3] This is surprising since, differently from $\mathcal{M}^{\star}$, our mechanism works without having knowledge about F (except for its range).

We first need some definitions and lemmas.

Definition 3.2. *Let $I_{s,\tau}$ be any interval of length $\tau \in (0, T]$ starting at $s \in [0, T - \tau]$. Then, the ratio between the prices posted by $\mathcal{M}_{\mathrm{C}}$ at the endpoints of $I_{s,\tau}$ is defined as:*

$$\kappa_{\tau}(s) \coloneqq \frac{p_{\mathcal{M}_{\mathrm{C}}}(s)}{p_{\mathcal{M}_{\mathrm{C}}}(s+\tau)}.$$

Intuitively, $\kappa_{\tau}(s)$ bounds the slope of the price function of $\mathcal{M}_{\mathrm{C}}$ in the time interval $I_{s,\tau}$, which depends on both the starting time s and the length τ of the interval. Moreover, notice that $\kappa_{\tau}(s) \geq 1$ since $p_{\mathcal{M}_{\mathrm{C}}}$ is non-increasing by Lemma 3.1. Next, we introduce an upper bound on the price ratios of all the time intervals of length τ, which is useful in deriving our main result.

Definition 3.3. *The maximum price ratio of $\mathcal{M}_{\mathrm{C}}$ over intervals of length $\tau \in (0, T]$ is denoted by:*

$$\kappa_{\tau} \coloneqq \max_{s \in [0, T-\tau]} \kappa_{\tau}(s).$$

The following lemma establishes a relation between the price function $p_{\mathcal{M}_{\mathrm{C}}}$ of $\mathcal{M}_{\mathrm{C}}$ and the expected value of the random variable $X_{\lambda T}$ representing the maximum initial valuation of agents arriving in the overall time period. This is crucial to prove Theorem 3.3.

Lemma 3.3. *In the RV setting with agents' initial valuations drawn from a distribution F, given $\tau \in (0, T]$ and $0 < \epsilon < 1$, there exists at least an interval $I_{s,\tau}$ of length $\tau \in (0, T]$ starting at $s \in [0, T - \tau]$ such that the prices $p_{\mathcal{M}_{\mathrm{C}}}(t)$ posted by mechanism $\mathcal{M}_{\mathrm{C}}$ during the time instants $t \in I_{s,\tau}$ lie in the range $\left[\frac{\mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)}{\kappa_{\tau}}, \mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)\right]$.*

Proof. Given how the function $p_{\mathcal{M}_{\mathrm{C}}}$ is defined, we can always define a time interval $I_{s,\tau}$ as desired by selecting its starting time $s \in [0, T - \tau]$ in such a

[3] To prove the lower bound, we follow an approach similar to that used by Babaioff et al. (2017) to bound the competitive ratio of their *Equal-Sample-of-Every-Scale* mechanism. However, our setting introduces additional challenges, since the agents' arrivals are stochastic and the valuations are discounted. Thus, our proofs require different techniques w.r.t. those of Babaioff et al. (2017).

way that $p_{\mathcal{M}_c}(s) = \mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)$. From Definition 3.3 we know that $\kappa_\tau(s) \leq \kappa_\tau$. Hence,

$$p_{\mathcal{M}_c}(s+\tau) = \frac{p_{\mathcal{M}_c}(s)}{\kappa_\tau(s)} = \\ = \frac{\mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)}{\kappa_\tau(s)} \geq \frac{\mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)}{\kappa_\tau}.$$

Since $p_{\mathcal{M}_c}$ is non-increasing by Lemma 3.1, for every $t \in I_{s,\tau}$ we have:

$$p_{\mathcal{M}_c}(t) \in [p_{\mathcal{M}_c}(s), p_{\mathcal{M}_c}(s+\tau)] \subseteq \\ \subseteq \left[\frac{\mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)}{\kappa_\tau}, \mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)\right].$$

Notice that the inequality involving $p_{\mathcal{M}_c}(s+\tau)$ holds with equality if $s \in \arg\max_{s\in[0,T-\tau]} \kappa_\tau(s)$. If this is the case, then there exists a unique interval verifying the statement. □

The following two lemmas are the final pieces that we need to prove Theorem 3.3. Lemma 3.4 establishes that, if the distribution F is MHR, then the same holds for the distribution $F_{X_{\lambda\tau}}$ of $X_{\lambda\tau}$. Lemma 3.5, given two intervals of length τ and τ' with $\tau \leq \tau'$, provides a lower bound on the expected value of $X_{\lambda\tau}$ which depends on the expected value of $X_{\lambda\tau'}$ and the logarithms of the expected number of agents' arrivals in the two intervals, respectively $\lambda\tau$ and $\lambda\tau'$.

Lemma 3.4. *$F_{X_{\lambda\tau}}$ has non-decreasing monotone hazard rate.*

Proof. Let us recall that the cumulative distribution function of $X_{\lambda\tau}$ is such that:

$$F_{X_{\lambda\tau}}(x) = e^{-\lambda\tau(1-F(x))}.$$

We compute the hazard rate of $F_{X_{\lambda\tau}}$ and show it is non-decreasing, as follows:

$$H_{X_{\lambda\tau}}(x) = \frac{f_{X_{\lambda\tau}}(x)}{1-F_{X_{\lambda\tau}}(x)} = \frac{\frac{\mathrm{d}}{\mathrm{d}x}F_{X_{\lambda\tau}}(x)}{1-F_{X_{\lambda\tau}}(x)} = \frac{\lambda\tau f(x)e^{-\lambda\tau(1-F(x))}}{1-e^{-\lambda\tau(1-F(x))}} \\ = \frac{\lambda\tau f(x)}{e^{\lambda\tau(1-F(x))}-1} = \lambda\tau\frac{f(x)}{1-F(x)}\frac{1-F(x)}{e^{\lambda\tau(1-F(x))}-1} \\ = \lambda\tau H(x)\frac{1-F(x)}{e^{\lambda\tau(1-F(x))}-1}.$$

Since F is MHR, the hazard rate $H(x)$ is non-decreasing. Notice that $F(x)$ is non-decreasing, and, thus, $1-F(x)$ in non-increasing. As a result,

proving that $\frac{1-F(x)}{e^{\lambda\tau(1-F(x))}-1}$ is non-decreasing in x is equivalent to show that $g(y) \coloneqq \frac{y}{e^{\lambda\tau y}-1}$ is non-increasing in y. We study the first derivative of $g(y)$:

$$\frac{\mathrm{d}}{\mathrm{d}y}g(y) = \frac{e^{\lambda\tau y}(1-\lambda\tau y)-1}{(e^{\lambda\tau y}-1)^2} \leq 0 \quad \text{for all } y \in [0,1].$$

This implies that $g(y)$ is non-increasing in y; hence, $\frac{1-F(x)}{e^{\lambda\tau(1-F(x))}-1}$ is non-decreasing in x. We conclude that $H_{X_{\lambda\tau}}(x)$ is monotone non-decreasing. □

In order to prove Lemma 3.5, we first state the following variant of the Chebyshev inequality Mitrinovic et al. (2013), where the adopted notation is specific for the proposition.

Proposition 3.2 (Mitrinovic et al. (2013))**.** *Suppose function $h(x)$ is positive and non-decreasing on $[a,b]$, function $g(x)$ is non-decreasing on $[a,b]$, and function $f(x)$ is continuous on $[a,b]$, then the following inequality holds:*

$$\frac{\int_a^b h(x)f(x)g(x)\mathrm{d}x}{\int_a^b h(x)f(x)\mathrm{d}x} \geq \frac{\int_a^b f(x)g(x)\mathrm{d}x}{\int_a^b f(x)\mathrm{d}x}.$$

Lemma 3.5. *For every $\tau, \tau' \in (0,T]$ with $\tau \leq \tau'$, it holds:*

$$\frac{\mathbb{E}[X_{\lambda\tau}]}{\mathbb{E}[X_{\lambda\tau'}]} \geq \frac{\ln(\lambda\tau)}{\ln(\lambda\tau')}.$$

Proof. Recall that $F_{X_{\lambda\tau}}(x) = e^{-\lambda\tau(1-F(x))}$. Then, we can write the following:

$$\begin{aligned}
\mathbb{E}[X_{\lambda\tau}] &= \int_0^\infty x f_{X_{\lambda\tau}}(x)\mathrm{d}x = \int_0^\infty 1 - F_{X_{\lambda\tau}}(x)\mathrm{d}x = \\
&= \int_0^\infty 1 - e^{-\lambda\tau(1-F(x))}\mathrm{d}x = \\
&= \int_0^\infty \frac{1-F(x)}{f(x)}\frac{1-e^{-\lambda\tau(1-F(x))}}{1-F(x)}\mathrm{d}F(x) = \\
&= \int_0^1 \frac{1}{H(F^{-1}(1-\eta))}\frac{1-e^{-\lambda\tau\eta}}{\eta}\mathrm{d}\eta.
\end{aligned}$$

Now, we apply Lemma 3.2. F having non-decreasing monotone hazard rate implies that $h(\eta) \coloneqq \frac{1}{H(F^{-1}(1-\eta))}$ is a non-decreasing function of η.

Hence, $h(k)$ is non-decreasing and positive on $[0,1]$. $g(\eta) \coloneqq \frac{1-e^{-\lambda\tau\eta}}{1-e^{-\lambda\tau'\eta}}$ is non-decreasing on $[0,1]$ and $f(\eta) \coloneqq \frac{1-e^{-\lambda\tau'\eta}}{\eta}$ is continuous on $[0,1]$. Thus,

$$\begin{aligned}\frac{\mathbb{E}[X_{\lambda\tau}]}{\mathbb{E}[X_{\lambda\tau'}]} &= \frac{\int_0^1 \frac{1}{H(F^{-1}(1-\eta))}\frac{1-e^{-\lambda\tau\eta}}{\eta}\mathrm{d}\eta}{\int_0^1 \frac{1}{H(F^{-1}(1-\eta))}\frac{1-e^{-\lambda\tau'\eta}}{\eta}\mathrm{d}\eta} = \\ &= \frac{\int_0^1 \frac{1}{H(F^{-1}(1-\eta))}\frac{1-e^{-\lambda\tau'\eta}}{\eta}\frac{1-e^{-\lambda\tau\eta}}{1-e^{-\lambda\tau'\eta}}\mathrm{d}\eta}{\int_0^1 \frac{1}{H(F^{-1}(1-\eta))}\frac{1-e^{-\lambda\tau'\eta}}{\eta}\mathrm{d}\eta} \geq \\ &\geq \frac{\int_0^1 \frac{1-e^{-\lambda\tau\eta}}{\eta}\mathrm{d}\eta}{\int_0^1 \frac{1-e^{-\lambda\tau'\eta}}{\eta}\mathrm{d}\eta} = \frac{\int_0^{\lambda\tau} \frac{1-e^{-t}}{t}\mathrm{d}t}{\int_0^{\lambda\tau'} \frac{1-e^{-t}}{t}\mathrm{d}t} = \frac{Ein(\lambda\tau)}{Ein(\lambda\tau')} = \\ &= \frac{\gamma - Ei(-\lambda\tau) + \ln(\lambda\tau)}{\gamma - Ei(-\lambda\tau') + \ln(\lambda\tau')} \geq \frac{\ln(\lambda\tau)}{\ln(\lambda\tau')},\end{aligned}$$

where $Ein(x) \coloneqq \int_0^x \frac{1-e^{-t}}{t}\mathrm{d}t$ is the entire exponential integral function, $Ei(x) \coloneqq \int_{-\infty}^x \frac{e^t}{t}\mathrm{d}t$ is the exponential integral function, and $\gamma \approx 0.577$ is the Euler's constant. □

Theorem 3.3. *Consider the RV setting with $\lambda\tau = (\lambda T)^{1-\epsilon} \geq 1 - \ln(e-1)$ for some $\tau \in (0,T]$ and $0 < \epsilon < 1$. Then, restricted to the set $\mathcal{F}$ of distributions F satisfying the MHR condition, mechanism $\mathcal{M}_{\mathrm{C}}$ has a competitive ratio that can be lower bounded as follows:*

$$\rho(\mathcal{M}_{\mathrm{C}}) \geq \frac{\xi(t_0 + T^{1-\epsilon}\lambda^{-\epsilon})(1-\epsilon)}{\kappa_\tau e}.$$

Proof. By hypothesis, we have $\lambda\tau = (\lambda T)^{1-\epsilon}$ for some $\tau \in (0,T]$ and $0 < \epsilon < 1$, which implies that $1 - \epsilon = \frac{\ln(\lambda\tau)}{\ln(\lambda T)}$. Moreover, let us fix a distribution F satisfying the MHR condition. From Lemma 3.3, there exists a time interval $I_{s,\tau}$ with starting time $s \in [0, T-\tau]$ such that $p_{\mathcal{M}_{\mathrm{C}}}(t) \in \left[\frac{\mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)}{\kappa}, \mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)\right]$ for every $t \in I_{s,\tau}$. We distinguish two cases, depending on whether the starting time of the interval is before or after the time t_0 characterizing mechanism $\mathcal{M}_{\mathrm{C}}$ (as defined in Theorem 3.1).

Case $s < t_0$. By using the fact that the seller's expected revenue for the overall time period is at least that achieved during the interval $I_{s,\tau}$, we have:

$$\mathbb{E}_F[\mathcal{R}(\mathcal{M}_{\mathrm{C}})] \geq p_{\mathcal{M}_{\mathrm{C}}}(s+\tau)\mathbb{P}\left\{Y_{\lambda\tau} \geq p_{\mathcal{M}_{\mathrm{C}}}(s)\right\}$$

$$\geq p_{\mathcal{M}_c}(s+\tau)\mathbb{P}\left\{X_{\lambda\tau}\xi(s+\tau) \geq \mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)\right\} \tag{3.12}$$

$$= p_{\mathcal{M}_c}(s+\tau)\mathbb{P}\left\{X_{\lambda\tau} \geq \mathbb{E}[X_{\lambda T}](1-\epsilon)\right\}$$

$$= p_{\mathcal{M}_c}(s+\tau)\mathbb{P}\left\{X_{\lambda\tau} \geq \mathbb{E}[X_{\lambda T}]\frac{\ln(\lambda\tau)}{\ln(\lambda T)}\right\}$$

$$\geq p_{\mathcal{M}_c}(s+\tau)\mathbb{P}\left\{X_{\lambda\tau} \geq \mathbb{E}[X_{\lambda\tau}]\right\} \tag{3.13}$$

$$\geq \frac{p_{\mathcal{M}_c}(s+\tau)}{e} \tag{3.14}$$

$$\geq \frac{\mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)}{\kappa_\tau e}$$

$$\geq \frac{\mathbb{E}[X_{\lambda T}]\xi(t_0+\tau)(1-\epsilon)}{\kappa_\tau e}.$$

Equation (3.12) holds since $X_{\lambda\tau}\xi(s+\tau)$ is a random variable representing the maximum initial valuation of agents arriving in a time interval of length τ weighted by the maximum possible discount, and, thus, it is always smaller than or equal to $Y_{\lambda\tau}$. Equation (3.13) follows from Lemma 3.5. Equation (3.14) follows from a result by Barlow and Marshall (1964), which implies that, for any MHR distribution, the probability of exceeding its expectation is at least $\frac{1}{e}$.

Case $s \geq t_0$. In this case, we can lower bound the seller's expected revenue for the overall time period with that obtained during the the interval $I_{t_0,\tau}$, as follows:

$$\begin{aligned}\mathbb{E}_F[\mathcal{R}(\mathcal{M}_c)] &\geq p_{\mathcal{M}_c}(t_0+\tau)\left(1-e^{-\lambda\tau}\right)\\ &\geq \xi(t_0+\tau)\frac{1}{e}\\ &\geq \frac{\mathbb{E}[X_{\lambda T}]\xi(t_0+\tau)(1-\epsilon)}{\kappa_\tau e},\end{aligned}$$

where for the first inequality we used the fact that the expected revenue in $I_{t_0,\tau}$ is at least the lowest price posted during the interval times the probability that at least one agent arrives in $I_{t_0,\tau}$, the second inequality holds since $\left(1-e^{-\lambda\tau}\right) \geq \frac{1}{e}$ when $\lambda\tau \geq 1-\ln(e-1) \simeq 0,46$, while the last inequality follows from the fact that $s \geq t_0$. Indeed, by Lemma 3.3, we can write the following:

$$p_{\mathcal{M}_c}(s+\tau) = \xi(s+\tau) \geq \frac{\mathbb{E}[X_{\lambda T}]\xi(s+\tau)(1-\epsilon)}{\kappa_\tau},$$

which implies that $\frac{\mathbb{E}[X_{\lambda T}](1-\epsilon)}{\kappa_\tau} \leq 1$.

We can now compute a lower bound on the ratio $\rho_F(\mathcal{M}_{\mathrm{C}})$ of mechanism $\mathcal{M}_{\mathrm{C}}$, as follows:

$$\begin{aligned}\rho_F(\mathcal{M}_{\mathrm{C}}) &= \frac{\mathbb{E}_F[\mathcal{R}(\mathcal{M}_{\mathrm{C}})]}{\mathbb{E}_F[\mathcal{R}(\mathcal{M}^\star)]} \\ &\geq \frac{\mathbb{E}_F[\mathcal{R}(\mathcal{M}_{\mathrm{C}})]}{\mathbb{E}[Y_{\lambda T}]} \\ &\geq \frac{\mathbb{E}[X_{\lambda T}]}{\mathbb{E}[Y_{\lambda T}]} \frac{\xi(t_0+\tau)(1-\epsilon)}{\kappa_\tau e} \\ &\geq \frac{\xi(t_0+\tau)(1-\epsilon)}{\kappa_\tau e}\end{aligned}$$

where the first inequality holds since $\mathbb{E}[Y_{\lambda T}]$ is the expected revenue of a mechanism that knows the actual realization of agents' initial valuations and arrival times, *i.e.*, the realization of variable $Y_{\lambda T}$.- This mechanism achieves an expected revenue greater than or equal to that obtained by the benchmark $\mathcal{M}^\star$, since the latter only knows the distribution of valuations. As for the second inequality, it is easy to see that $\frac{\mathbb{E}[X_{\lambda T}]}{\mathbb{E}[Y_{\lambda T}]} \geq 1$. Finally, by recalling the condition $\lambda\tau = (\lambda T)^{1-\epsilon}$, we have $\tau = T^{1-\epsilon}\lambda^{-\epsilon}$, which allows us to write the following bound:

$$\rho(\mathcal{M}_{\mathrm{C}}) \geq \frac{\xi(t_0 + T^{1-\epsilon}\lambda^{-\epsilon})(1-\epsilon)}{\kappa_\tau e}.$$

This concludes the proof. □

The idea of the proof is to use $\rho_F(\mathcal{M}_{\mathrm{C}}) \geq \frac{\mathbb{E}_F[\mathcal{R}(\mathcal{M}_{\mathrm{C}})]}{\mathbb{E}[Y_{\lambda T}]}$, following from the fact that $\mathbb{E}_F[\mathcal{R}(\mathcal{M}^\star)]$ cannot be larger than $\mathbb{E}[Y_{\lambda T}]$, which is the expected revenue achieved by an optimal mechanism that knows the realization of agents' initial valuations and arrivals. Then, $\mathbb{E}_F[\mathcal{R}(\mathcal{M}_{\mathrm{C}})]$ is lower bounded by the revenue that $\mathcal{M}_{\mathrm{C}}$ achieves in a suitably defined interval $I_{s,\tau}$, whose existence is guaranteed by Lemma 3.3. Moreover, Lemmas 3.3, 3.4, and 3.5, together with the properties of MHR distributions, allow us to write $\mathbb{E}_F[\mathcal{R}(\mathcal{M}_{\mathrm{C}})] \geq \frac{\mathbb{E}[X_{\lambda T}]\xi(t_0+\tau)(1-\epsilon)}{\kappa_\tau e}$, giving the result as $\mathbb{E}[Y_{\lambda T}] \leq \mathbb{E}[X_{\lambda T}]$.

3.3.2 A Mechanism with a Piecewise Constant Price

We introduce a new mechanism $\mathcal{M}_{\mathrm{PC}}$ whose pricing strategy $p_{\mathcal{M}_{\mathrm{PC}}}$ is a piecewise constant function. This turns out to be useful in all the situations in which the seller is constrained not to change the posted price too often,

e.g., when the mechanism is required to set prices for time intervals having a given minimum length. Our main result (Theorem 3.4) is a lower bound on the competitive ratio of $\mathcal{M}_{\text{PC}}$ in the RV setting, which is comparable to that obtained for $\mathcal{M}_{\text{C}}$ in Theorem 3.3. Thus, we show that, even in presence of constraints on the allowed pricing strategies, we are still able to design mechanisms with good performances in terms of competitive ratio. Clearly, $\mathcal{M}_{\text{PC}}$ depends on the minimum length requirement, which influences the resulting lower bound. In particular, $\mathcal{M}_{\text{PC}}$ is tuned by a parameter δ related to the number of time intervals in which the price must be constant.

Mechanism $\mathcal{M}_{\text{PC}}$ works by evenly partitioning the time interval $[0, t_0]$ into $\lceil \log_\delta h \rceil$ sub-intervals of length τ, where $\delta \in (1, h]$ and $t_0 \in [0, T]$ are suitably defined parameters. Then, the remaining time $[t_0, T]$ is organized in other sub-intervals of length τ. As a result, $[0, T]$ is partitioned into $\left\lceil \frac{T}{\tau} \right\rceil$ sub-intervals, which, overloading notation, we denote by $I_i \coloneqq [(i-1)\tau, \min\{i\tau, T\}]$ for $i = 1, \ldots, \left\lceil \frac{T}{\tau} \right\rceil$. Notice that $\tau = \frac{t_0}{\lceil \log_\delta h \rceil}$, and, thus, parameters t_0 and δ can be tuned to match the required minimum length τ. The pricing strategy $p_{\mathcal{M}_{\text{PC}}}$ of $\mathcal{M}_{\text{PC}}$ is defined in such a way that the price is constant in each interval I_i. By letting $p_{\mathcal{M}_{\text{PC}}}(I_i)$ be the price posted during I_i, we define the function $p_{\mathcal{M}_{\text{PC}}}$ as follows: [4]

$$p_{\mathcal{M}_{\text{PC}}}(I_i) \coloneqq \begin{cases} \frac{h}{\delta^i}\xi(i\tau) & \text{if } i = 1, \ldots, \lfloor \log_\delta h \rfloor \\ \xi(i\tau) & \text{if } i = \lceil \log_\delta h \rceil, \ldots, \left\lceil \frac{T}{\tau} \right\rceil - 1. \\ \xi((i-1)\tau) & \text{if } i = \left\lceil \frac{T}{\tau} \right\rceil \end{cases}$$

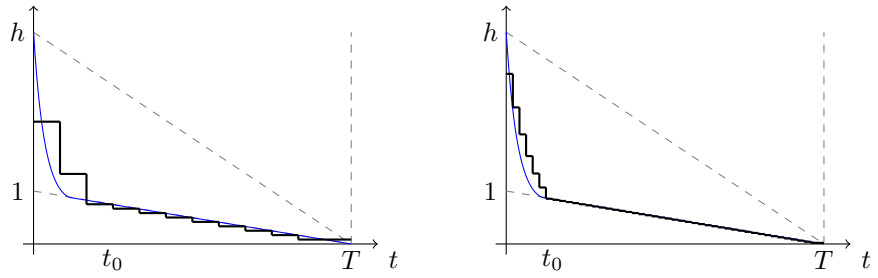
We compare in Figure 3.1 the prices of $\mathcal{M}_{\text{C,lin}}$ and $\mathcal{M}_{\text{PC,lin}}$ (*i.e.*, $\mathcal{M}_{\text{PC}}$ with a linear discount) in a specific setting for two values of τ. Notice that $\mathcal{M}_{\text{PC}}$ can be thought of as an extension of the *Equal-Sample-of-Every-Scale* (ESoES) mechanism by Babaioff et al. (2017) to the more general setting in which agents arrive stochastically according to a Poisson process and agents' valuations are discounted over time.

Before proving our main result, we need the following lemma, which is the analogous of Lemma 3.3 working for mechanism $\mathcal{M}_{\text{PC}}$ instead of $\mathcal{M}_{\text{C}}$.

Lemma 3.6. *In the RV setting with agents' initial valuations drawn from a distribution F, given $0 < \epsilon < 1$, there exists $i = 1, \ldots, \lceil \log_\delta h \rceil$ such that the price $p_{\mathcal{M}_{\text{PC}}}(I_i)$ posted by $\mathcal{M}_{\text{PC}}$ during the interval I_i lies in the range $\left[\frac{\nu}{\delta}\xi(i\tau), \nu\xi(i\tau)\right]$, where $\nu \coloneqq \max\{1, \mathbb{E}[X_{\lambda T}](1-\epsilon)\}$.*

Proof. For ease of presentation, in the rest of this proof, we define $\tilde{I}_i \coloneqq \left[\frac{\nu}{\delta}\xi(i\tau), \nu\xi(i\tau)\right]$ for any $i = 1, \ldots, \lceil \log_\delta h \rceil$. By contradiction, suppose that

[4] Whenever T is not divisible by τ, then the last time interval is shorter than τ. Thus, in order to satisfy the minimum length constraint, we set its price equal to the one in the preceeding interval.



(a) $\tau = 1$ **(b)** $\tau = 0.25$

Figure 3.1: *Prices of mechanisms $\mathcal{M}_{\text{C,lin}}$ (blue) and $\mathcal{M}_{\text{PC,lin}}$ (black) when $h = 2.8, \lambda = 10$, and $T = 12$.*

there is no $i = 1, \dots, \lceil \log_\delta h \rceil$ such that $p_{\mathcal{M}_{\text{PC}}}(I_i) \in \tilde{I}_i$. Notice that ν is a lower bound on $\mathbb{E}[X_{\lambda T}]$ and belongs to the range $\in [1, h)$.

We reach a contradiction by employing an iterated reasoning. As a first step, we observe that either $\nu \in \left(\frac{h}{\delta}, h\right)$ or $\nu \in \left[1, \frac{h}{\delta}\right]$. If $\nu \in \left(\frac{h}{\delta}, h\right)$, then $p_{\mathcal{M}_{\text{PC}}}(I_1) = \frac{h}{\delta}\xi(\tau)$ is in the range $\tilde{I}_1 = \left[\frac{\nu}{\delta}\xi(\tau), \nu\xi(\tau)\right]$. Hence, it must hold $\nu \in [1, \frac{h}{\delta}]$. Then, as a second step, we can conclude that either $\nu \in \left(\frac{h}{\delta^2}, \frac{h}{\delta}\right]$ or $\nu \in \left[1, \frac{h}{\delta^2}\right]$. If $\nu \in \left(\frac{h}{\delta^2}, \frac{h}{\delta}\right]$, then $p_{\mathcal{M}_{\text{PC}}}(I_2) = \frac{h}{\delta^2}\xi(2\tau)$ is in the range $\tilde{I}_2 = \left[\frac{\nu}{\delta}\xi(2\tau), \nu\xi(2\tau)\right]$. Hence, it must hold $\nu \in [1, \frac{h}{\delta^2}]$. By iterating the reasoning until the $\lfloor \log_\delta h \rfloor$-th step, we obtain that either $\nu \in \left(\frac{h}{\delta^{\lfloor \log_\delta h \rfloor}}, \frac{h}{\delta^{\lfloor \log_\delta h \rfloor - 1}}\right]$ or $\nu \in \left[1, \frac{h}{\delta^{\lfloor \log_\delta h \rfloor}}\right]$.

Let us first consider the case in which it holds $\lfloor \log_\delta h \rfloor \neq \lceil \log_\delta h \rceil$. If $\nu \in \left(\frac{h}{\delta^{\lfloor \log_\delta h \rfloor}}, \frac{h}{\delta^{\lfloor \log_\delta h \rfloor - 1}}\right]$, then $p_{\mathcal{M}_{\text{PC}}}\left(I_{\lfloor \log_\delta h \rfloor}\right) \in \tilde{I}_{\lfloor \log_\delta h \rfloor}$ since:

$$\frac{h}{\delta^{\lfloor \log_\delta h \rfloor}}\xi(\lfloor \log_\delta h \rfloor \tau) \in \left[\frac{\nu}{\delta}\xi(\lfloor \log_\delta h \rfloor \tau), \nu\xi(\lfloor \log_\delta h \rfloor \tau)\right].$$

Hence, it must hold $\nu \in \left[1, \frac{h}{\delta^{\lfloor \log_\delta h \rfloor}}\right]$. Then, $p_{\mathcal{M}_{\text{PC}}}\left(I_{\lceil \log_\delta h \rceil}\right) = \xi(\lceil \log_\delta h \rceil \tau)$ belongs to the range $\tilde{I}_{\lceil \log_\delta h \rceil} = \left[\frac{\nu}{\delta}\xi(\lceil \log_\delta h \rceil \tau), \nu\xi(\lceil \log_\delta h \rceil \tau)\right]$, which leads to a contradiction.

Now, suppose that $\lfloor \log_\delta h \rfloor = \lceil \log_\delta h \rceil = \log_\delta h$. Then, in the $\lfloor \log_\delta h \rfloor$-th step of the iterated reasoning, we can conclude that $\nu \in \left[1, \frac{h}{\delta^{\log_\delta h - 1}}\right]$ and $p_{\mathcal{M}_{\text{PC}}}\left(I_{\log_\delta h}\right) = \xi((\log_\delta h)\tau)$ is in the range

$$\tilde{I}_{\log_\delta h} = \left[\frac{\nu}{\delta}\xi((\log_\delta h)\tau), \nu\xi((\log_\delta h)\tau)\right],$$

which leads to the final contradiction. □

Now, we provide our main result. The idea behind its proof is similar to the one used for Theorem 3.3.

Theorem 3.4. *Consider the RV setting with* $\lambda\tau = (\lambda T)^{1-\epsilon} \geq 1 - \ln(e-1)$ *for some* $\tau \in (0, T]$ *and* $0 < \epsilon < 1$*. Then, restricted to the set* $\mathcal{F}$ *of distributions* F *satisfying the MHR condition, mechanism* $\mathcal{M}_{\text{PC}}$ *has a competitive ratio that can be lower bounded as follows:*

$$\rho(\mathcal{M}_{\text{PC}}) \geq \frac{\xi((\lceil \log_\delta h \rceil + 1)T^{1-\epsilon}\lambda^{-\epsilon})(1-\epsilon)}{\delta e}.$$

Proof. By hypothesis we have $\lambda\tau = (\lambda T)^\epsilon$ for $\tau = \frac{t_0}{\lceil \log_\delta h \rceil} \in (0, T]$. We distinguish two cases, depending on whether $\mathbb{E}[X_{\lambda T}](1-\epsilon)$ is greater or lower than one. Note that $\mathbb{E}[X_{\lambda T}](1-\epsilon)$ is a lower bound for $\mathbb{E}[X_{\lambda T}]$ and that one is the minimum value that $\mathbb{E}[X_{\lambda T}]$ can assume. In particular $\mathbb{E}[X_{\lambda T}] = 1$ when F is the point distribution such that $P(V_i \leq 1) = P(V_i = 1) = 1$.

Case $\mathbb{E}[X_{\lambda T}](1-\epsilon) >= 1$**.** For Lemma 3.6, there exists an $i \in \{1, \ldots, \lceil \log_\delta h \rceil\}$ such that the price $p_i^* = p_{\mathcal{M}_{\text{PC}}}(I_i)$ lies in the range $\tilde{I}_i = \left[\frac{\mathbb{E}[X_{\lambda T}]\xi(i\tau)(1-\epsilon)}{\delta}, \mathbb{E}[X_{\lambda T}]\xi(i\tau)(1-\epsilon)\right]$. By using the fact that the seller's expected revenue for the overall time period is at least that achieved during the interval I_i, we have:

$$\begin{aligned}
\mathbb{E}[\mathcal{R}(\mathcal{M}_{\text{PC}})] &\geq p_i^* \mathbb{P}(Y_{\lambda\tau,i} \geq p_i^*) \\
&\geq p_i^* \mathbb{P}\Big(X_{\lambda\tau}\xi(i\tau) \geq \mathbb{E}[X_{\lambda T}]\xi(i\tau)(1-\epsilon)\Big) && (3.15) \\
&= p_i^* \mathbb{P}(X_{\lambda\tau} \geq \mathbb{E}[X_{\lambda T}](1-\epsilon)) \\
&= p_i^* \mathbb{P}\left(X_{\lambda\tau} \geq \mathbb{E}[X_{\lambda T}]\frac{\ln(\lambda\tau)}{\ln(\lambda T)}\right) \\
&\geq p_i^* \mathbb{P}(X_{\lambda\tau} \geq \mathbb{E}[X_{\lambda\tau}]) && (3.16) \\
&\geq \frac{p_i^*}{e} && (3.17) \\
&\geq \frac{\mathbb{E}[X_{\lambda T}]\xi(i\tau)(1-\epsilon)}{\delta e} \\
&\geq \frac{\mathbb{E}[X_{\lambda T}]\xi(\lceil \log_\delta h \rceil \tau)(1-\epsilon)}{\delta e} && (3.18) \\
&\geq \frac{\mathbb{E}[X_{\lambda T}]\xi((\lceil \log_\delta h \rceil + 1)\tau)(1-\epsilon)}{\delta e}
\end{aligned}$$

Equation (3.15) holds since $X_{\lambda\tau}\xi(i\tau)$ is a random variable representing the maximum initial valuation of agents arriving in a time interval of length

τ weighted by the maximum possible discount, thus it is always smaller than or equal to $Y_{\lambda\tau,i}$. Equation (3.16) follows from Lemma 3.5. Equation (3.17) follows from a result by Barlow and Marshall (1964), which implies that, for any MHR distribution, the probability of exceeding its expectation is at least $\frac{1}{e}$.

Case $\mathbb{E}[X_{\lambda T}](1-\epsilon) < 1$. In this case we can lower bound the seller's expected revenue for the overall time period with that obtained during the interval $I_{\lceil \log_\delta h \rceil + 1}$, as follows:

$$\begin{aligned}
\mathbb{E}_F[\mathcal{R}(\mathcal{M}_{\text{PC}})] &\geq p^*_{\lceil \log_\delta h \rceil + 1} \left(1 - e^{-\lambda\tau}\right) \\
&\geq \xi((\lceil \log_\delta h \rceil + 1)\tau)\frac{1}{e} \\
&\geq \frac{\mathbb{E}[X_{\lambda T}]\xi((\lceil \log_\delta h \rceil + 1)\tau)(1-\epsilon)}{\delta e},
\end{aligned}$$

where for the first inequality we used the fact that the expected revenue in $I_{\lceil \log_\delta h \rceil + 1}$ is the price posted during the interval times the probability that at least one agent arrives in $I_{\lceil \log_\delta h \rceil + 1}$, the second inequality holds since $\left(1 - e^{-\lambda\tau}\right) \geq \frac{1}{e}$ when $\lambda\tau \geq 1 - \ln(e-1) \simeq 0,46$, while the last inequality follows from the fact that $\mathbb{E}[X_{\lambda T}](1-\epsilon) < 1$ and $\delta \geq 1$.

We can now compute a lower bound on the ratio of the mechanism $\rho_F(\mathcal{M}_{\text{PC}})$, as follows:

$$\begin{aligned}
\rho_F(\mathcal{M}_{\text{PC}}) &= \frac{\mathbb{E}_F[\mathcal{R}(\mathcal{M}_{\text{PC}})]}{\mathbb{E}_F[\mathcal{R}(\mathcal{M}^\star)]} \\
&\geq \frac{\mathbb{E}_F[\mathcal{R}(\mathcal{M}_{\text{PC}})]}{\mathbb{E}[Y_{\lambda T}]} \\
&\geq \frac{\mathbb{E}[X_{\lambda T}]}{\mathbb{E}[Y_{\lambda T}]} \frac{\xi((\lceil \log_\delta h \rceil + 1)\tau)(1-\epsilon)}{\delta e} \\
&\geq \frac{\xi((\lceil \log_\delta h \rceil + 1)\tau)(1-\epsilon)}{\delta e}
\end{aligned}$$

where it is easy to see that $\frac{\mathbb{E}[X_{\lambda T}]}{\mathbb{E}[Y_{\lambda T}]} \geq 1$. By recalling the condition $\lambda\tau = (\lambda T)^{1-\epsilon}$, we have $\tau = T^{1-\epsilon}\lambda^{-\epsilon}$, which allows us to write the following bound:

$$\rho_F(\mathcal{M}_{\text{PC}}) \geq \frac{\xi((\lceil \log_\delta h \rceil + 1)T^{1-\epsilon}\lambda^{-\epsilon})(1-\epsilon)}{\delta e}.$$

This concludes the proof. □

3.4 Examples of Mechanisms

In order to ease the reader in the understanding of our mechanisms, we provide their graphical representation for the case of a linear discount function $\xi_{\text{lin}}(t) \coloneqq 1 - \frac{t}{T}$. In particular, we focus on mechanisms $\mathcal{M}_{\text{C,lin}}$ and $\mathcal{M}_{\text{PC,lin}}$, where the latter is the linear-discount version of the general-discount mechanism $\mathcal{M}_{\text{PC}}$. The price function $p_{\mathcal{M}_{\text{PC,lin}}}$ of $\mathcal{M}_{\text{PC,lin}}$ can be easily obtained from that of $\mathcal{M}_{\text{PC}}$ by using the specific definition of the discount function. We report it below for completeness.

$$p_{\mathcal{M}_{\text{PC,lin}}}(I_i) \coloneqq \begin{cases} \frac{h}{\delta^i}\left(1 - \frac{i\tau}{T}\right) & \text{if } i = 1, \dots, \lfloor \log_\delta h \rfloor \\ 1 - \frac{i\tau}{T} & \text{if } i = \lceil \log_\delta h \rceil, \dots, \left\lceil \frac{T}{\tau} \right\rceil - 1 \\ 1 - \frac{(i-1)\tau}{T} & \text{if } i = \left\lceil \frac{T}{\tau} \right\rceil \end{cases} .$$

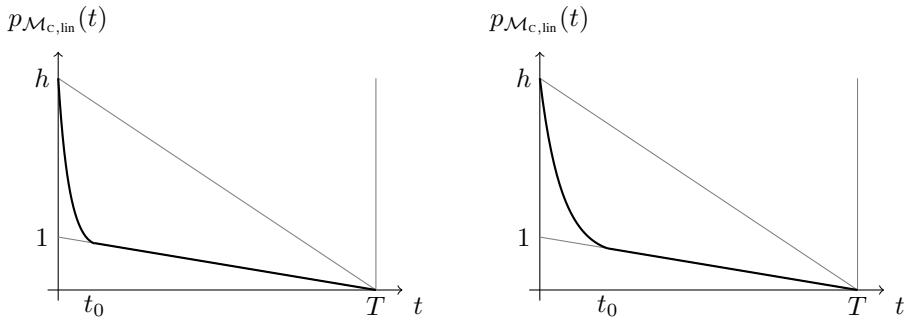
We tune the parameters h, λ, and T so as to simulate real-world scenarios representing the long-term rental of a single room. In particular, we fix the parameter values by analyzing data from a real-world co-living company operating on the web, counting over 7000 rooms. [5] In this scenario, the goal is to rent a single room to students for a fixed period of one year. We set $T = 12$, assuming that each time interval of length 1 corresponds to a period of one month, and we fix the starting time $t = 0$ as the time in which the contract of the previous tenant ends. Therefore, the room value is discounted over time as an effect of the ever shorter period of stay of the future tenant. We also set $h = 2.8$, which means that the highest valuation for the room is around three times the lowest one.

Figure 3.2 shows how the shape of mechanism $\mathcal{M}_{\text{C,lin}}$ changes by varying the arrival rate λ, which is the expected number of agents arriving in a time interval of one month. We observe that the price function decreases as a linearly discounted exponential function in the time interval $[0, t_0]$, and, then, as a linear function in $[t_0, T]$. Notice that, by comparing Figure 3.2a and Figure 3.2b, it is easy to see that the time instant t_0 gets closer to zero as the arrival rate λ increases. This can be explained by recalling that the mechanism has to deal with the trade-off between setting high prices so as to achieve high revenues and posting lower prices in order to increase the probability of selling the item. In the first period of time, the seller posts high prices hoping for the arrival of an agent having an high valuation. This phase cannot be too long, otherwise the item risks to remain unsold, and, on the other hand, it cannot even be too short, otherwise the probability of encountering such an high-valuation agent becomes too small. Therefore, when the arrival rate decreases, the high-price phase must be enlarged in

[5] We cannot disclose the name of the company for privacy reasons.

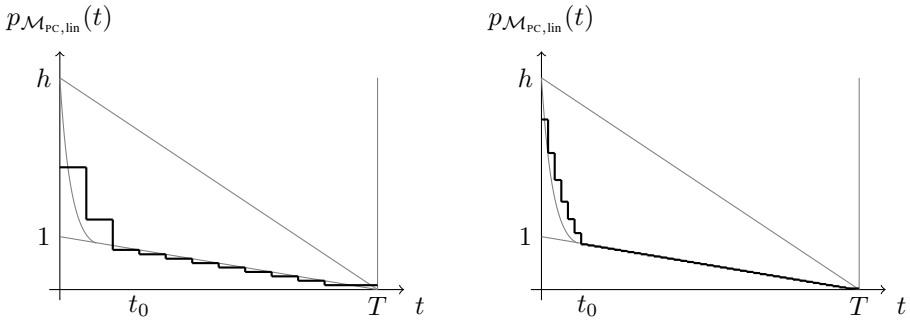
order to still have some chance of concluding the purchase for an high price (Figure 3.2b), while, if λ increases, it suffices to post high prices for a shorter time period (Figure 3.2a).

Figure 3.3 represents the behavior of mechanism $\mathcal{M}_{\text{PC,lin}}$ when we impose different constraints on the minimum time in which the price must be constant. In particular, Figure 3.3a and Figure 3.3b show the shape of $\mathcal{M}_{\text{PC,lin}}$ when the posted price does not change for time intervals of length τ equal to one month (*i.e.*, $\tau = 1$) and one week (*i.e.*, $\tau = 0.25$), respectively.



(a) $\mathcal{M}_{\text{C,lin}}$*:* $h = 2.8, \lambda = 10, T = 12$ **(b)** $\mathcal{M}_{\text{C,lin}}$*:* $h = 2.8, \lambda = 2, T = 12$

Figure 3.2: *Mechanism* $\mathcal{M}_{\text{C,lin}}$ *with different rate parameters* λ*.*



(a) $\mathcal{M}_{\text{PC,lin}}$*:* $\tau = 1, h = 2.8, \lambda = 10, T = 12$ **(b)** $\mathcal{M}_{\text{PC,lin}}$*:* $\tau = 0.25, h = 2.8, \lambda = 10, T = 12$

Figure 3.3: *Mechanism* $\mathcal{M}_{\text{PC,lin}}$ *with different constraints on the minimum time in which the price must be constant.*

3.5 Empirical Evaluation

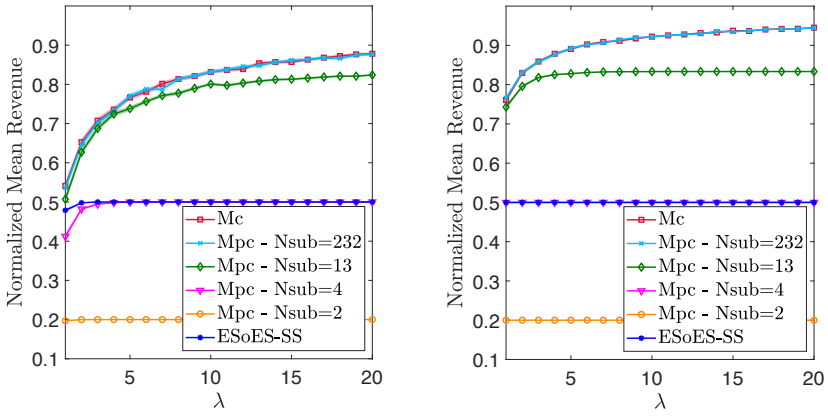
We evaluate mechanisms $\mathcal{M}_{\text{C}}$, $\mathcal{M}_{\text{PC}}$, and a natural adaption of the ESoES mechanism by Babaioff et al. (2017) to *stochastic settings* with no time

discounting (called ESoES-SS). The pricing strategy of ESoES-SS is defined as follows. First, we compute the prices of ESoES by setting the number of agents equal to the expected number λT of agents arriving in $[0, T]$ according to a Poisson process of parameter λ. Then, ESoES-SS proposes the price that ESoES would propose to the i-th agent arrived if $i \leq \lambda T$ and 1 otherwise.

We use the following parameters values for the experiments: $\lambda \in \{1, \dots, 20\}, T \in \{10, 20, 50, 100\}$, and $h \in \{2, \dots, 20\}$. The following results do not consider time discounting so as to have a fair comparison between our mechanisms and ESoES-SS. Further results with a linear discount function are provided in Appendix A.

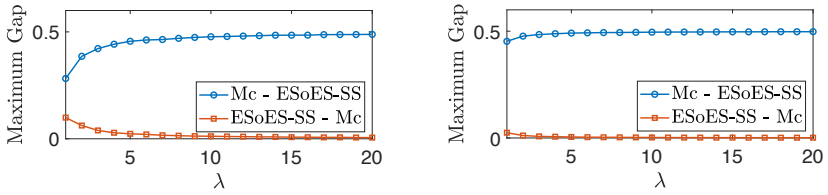
Result #1 We study a RV setting with a uniform probability distribution over $[1, h]$. For every combination of values of λ, T, h, we run 1000 Monte Carlo simulations, evaluating the revenue provided by mechanisms ESoES-SS, $\mathcal{M}_{\text{C}}$, and $\mathcal{M}_{\text{PC}}$. In particular, we analyze some variants of mechanisms $\mathcal{M}_{\text{PC}}$ differing for the number of subintervals (*i.e.*, $Nsub$) in which $[0, t_0]$ is partitioned. Furthermore, we normalize the revenue provided by the mechanisms in each simulation with respect to h. We report the results in Figure 3.4 for $T = 10$ and $T = 50$, when $h = 10$. The results obtained for different values of h are similar. $\mathcal{M}_{\text{C}}$ and $\mathcal{M}_{\text{PC}}$ with $Nsub = 232$ have overlapping performances that beat those of the other mechanisms. $\mathcal{M}_{\text{PC}}$ with $Nsub = 13$ has a performance close to that of the previous two mechanisms, showing that mechanism $\mathcal{M}_{\text{PC}}$ provides good performances even with few subintervals. $\mathcal{M}_{\text{PC}}$ with $Nsub = 4$ and ESoES-SS have almost overlapping performances, showing that very few subintervals are sufficient to $\mathcal{M}_{\text{PC}}$ to match the performances of ESoES-SS. The worst mechanism is $\mathcal{M}_{\text{PC}}$ with $Nsub = 2$. The loss of ESoES-SS w.r.t. $\mathcal{M}_{\text{C}}$ averaged over the values of λ is about $0.3\,h$ when $T = 10$, and $0.4\,h$ when $T = 50$. Surprisingly, the performances of ESoES-SS seem to do not strictly depend on λ and T.

Result #2 We study an IV setting. For every combination of values of λ, T, h, and for every $v \in \{1.0, 1.5, 2.0, \dots, h\}$, we run 1000 Monte Carlo simulations, evaluating the normalized revenue provided by mechanisms ESoES-SS and $\mathcal{M}_{\text{C}}$. For every combination of values of λ, T, h, we calculate $\max_v \frac{\mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\text{C}})] - \mathbb{E}_v[\mathcal{R}(\text{ESoES-SS})]}{h}$, corresponding to the maximum normalized loss of ESoES-SS w.r.t. $\mathcal{M}_{\text{C}}$ over all valuations v, and, then, we calculate $\max_v \frac{\mathbb{E}_v[\mathcal{R}(\text{ESoES-SS})] - \mathbb{E}_v[\mathcal{R}(\mathcal{M}_{\text{C}})]}{h}$, corresponding to the maximum normalized loss of $\mathcal{M}_{\text{C}}$ w.r.t. ESoES-SS over all valuations v. These two indexes



(a) $T = 10$ **(b)** $T = 50$

Figure 3.4: *Average normalized revenue of $\mathcal{M}_{\text{C}}$, $\mathcal{M}_{\text{PC}}$, ESoES-SS in a RV setting w. uniform distribution ($h = 10$).*

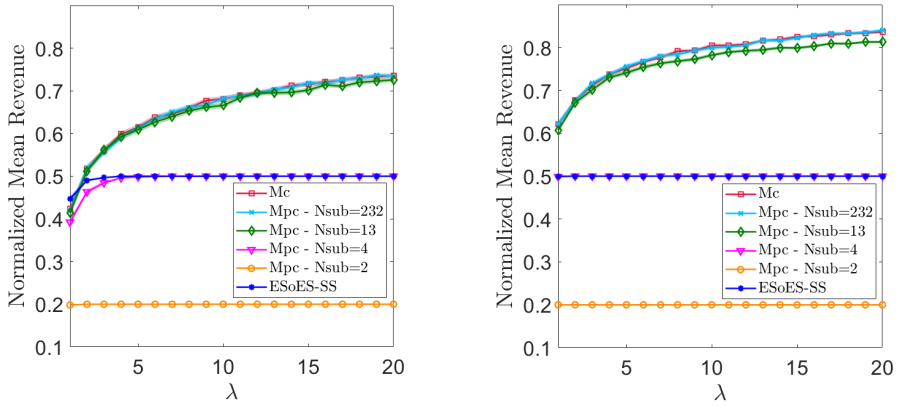


(a) $T = 10$ **(b)** $T = 50$

Figure 3.5: *Maximum difference between the normalized revenues of $\mathcal{M}_{\text{C}}$ and ESoES-SS in an IV setting ($h = 10$).*

are shown in Figure 3.5 for $T = 10$ and $T = 50$, when $h = 10$. The results obtained for different values of h are similar. The loss of ESoES-SS w.r.t. $\mathcal{M}_{\text{C}}$ is always larger than $0.5\,h$ except when both λ and T assume small values, while the loss of $\mathcal{M}_{\text{C}}$ w.r.t. ESoES-SS is negligible. Furthermore, the two losses converge to two constants as λ and T increase. This shows that, even if there are some special settings where ESoES-SS performs better than $\mathcal{M}_{\text{C}}$, the improvement is negligible. Instead, mechanism $\mathcal{M}_{\text{C}}$, which is designed to deal with stochastic arrivals, provides a very significant improvement. In particular, we observe that the difference between the revenue provided by ESoES-SS and that provided by $\mathcal{M}_{\text{C}}$ is maximized for small values of v close to 1, while between $\mathcal{M}_{\text{C}}$ and ESoES-SS for large values of v close to h.

Result #3 We compare $\mathcal{M}_{\text{C}}$, $\mathcal{M}_{\text{PC}}$, and ESoES-SS when the distribution of the agents' valuations is *not* MHR. We perform an experiment similar to that of Result #1. Here, agents' valuations are drawn from a truncated normal distribution with $\mu = \frac{h-1}{2}$, $\sigma^2 = 2$, and support $[1, h]$. Figure 3.6 is similar to Figure 3.4. Observe that, in this setting, the performances of $\mathcal{M}_{\text{PC}}$ with $Nsub = 13$ and $\mathcal{M}_{\text{PC}}$ with $Nsub = 232$ are analogous. This means that, tuning the parameters in a suitable way, we can impose a time constraint with almost no loss in the normalized mean revenue. Moreover, the truncated normal distribution is *not* MHR, hence, all the bounds on the competitive ratio of the mechanism do not hold. Despite this fact, we see that, in this scenario, the behavior of the mean normalized revenue is comparable to that of Result #1. In particular, the loss of ESoES-SS w.r.t. $\mathcal{M}_{\text{C}}$ averaged over the values of λ is about $0.2\,h$ when $T = 10$, and slightly larger when $T = 50$.

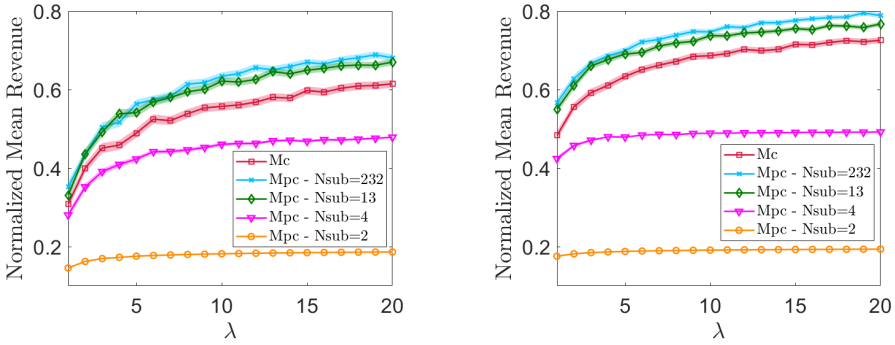


(a) $T = 10$ **(b)** $T = 50$

Figure 3.6: *Average normalized revenue of $\mathcal{M}_{\text{C}}$, $\mathcal{M}_{\text{PC}}$, and ESoES-SS.*

Result #4 We analyze mechanisms $\mathcal{M}_{\text{C}}$ and $\mathcal{M}_{\text{PC}}$ with different $Nsub$ values when the valuations of the agents are linearly discounted. For every λ we run 1000 Monte Carlo simulations, with $h = 10$. Given parameters λ and h, we simulated the arrivals of agents drawn from a uniform distribution with support $[1, h]$ and we computed the revenue of the mechanisms. We normalized the results by h and, for each value of λ, we average by the simulations. Then, for each value of λ, we plot in Figure 3.7 the normalized mean revenues of the mechanisms, for $T = 10$ and $T = 50$. We observe that $\mathcal{M}_{\text{C}}$ is no longer the best mechanism in terms of normalized mean revenue.

The interesting fact is that, a suitably tuned mechanism $\mathcal{M}_{\text{PC}}$ can reach a better average revenue than $\mathcal{M}_{\text{C}}$ in some IV scenarios.



(a) $T = 10$ **(b)** $T = 50$

Figure 3.7: *Average normalized revenue of $\mathcal{M}_{\text{C}}$, $\mathcal{M}_{\text{PC}}$, and ESoES-SS.*

CHAPTER 4

Multi-Armed Bandit Problem with Temporally-Partitioned Rewards: When Partial Feedback Counts

Chapter 4 studies the setting in which the reward derived by selling products is partitioned over multiple time instant after the sale and its temporal structure is unknown to the seller. The proposed price may have delayed effects on the reward, and the seller has to learn which is the revenue-maximizing price. One of the fundamental challenge of such a temporal structure is whether it is possible for the seller to exploit incomplete reward samples to speed up the learning procedure. We remark that the techniques employed to address the problem presented in Chapter 3 are significantly distinct from those implemented in the present chapter. This is due to the different objectives of the two chapters: Chapter 3 aims to sell a single unit of an item, whereas Chapter 4 focuses on selling multiple units of the same item. Selling many units of the same item in a sequential manner allows the seller to employ learning techniques in order to find a good pricing strategy. The guarantees provided by such techniques ensure that the seller does not loose too much during the learning procedure. Conversely, no learning is possible when selling a single unit of product.

Chapter 4. Multi-Armed Bandit Problem with Temporally-Partitioned Rewards: When Partial Feedback Counts

In this chapter, we study a novel bandit setting, namely *Multi-Armed Banditwith Temporally-Partitioned Rewards* (TP-MAB), in which the stochastic reward associated with the pull of an arm is partitioned over a finite number of consecutive rounds following the pull. This setting, unexplored so far to the best of our knowledge, is a natural extension of delayed-feedback bandits to the case in which rewards may be dilated over a finite-time span after the pull instead of being fully disclosed in a single, potentially delayed round. An introduction to the TP-MAB framework is provided in Section 1.1, while a further discussion on applications and other related works from the literature is provided in the following paragraphs. Section 4.1 provides a formal model and a lower bound for the general class of TP-MAB problems. Section 4.2 introduces the α-*smoothness* property, which characterizes the structure of the reward, and provide an improved lower bound for the α-smooth problem. In Section 4.3 we design two algorithms to address TP-MAB problems, namely, `TP-UCB-FR` and `TP-UCB-EW`, which exploit the partial information disclosed by the reward collected over time. We show that our algorithms provide better asymptotical regret upper bounds than delayed-feedback bandit algorithms when α-smoothness holds. In Section 4.4 we empirically evaluate their performance across a wide range of settings, both synthetically generated and from a real-world media recommendation problem.

Other Motivating Applications. Sequential decision-making occurs in many real-world scenarios such as clinical trials, recommender systems, web advertising, and e-commerce. A motivating example for TP-MABs is recommending media content and, in particular, song playlists to a class of users (*i.e.*, users sharing similar characteristics). In this setting, each arm corresponds to a playlist. The reward is measured in listening time (proportional to the user's appreciation). The goal is to find the playlist that maximizes the reward. The recommendation system suggests a playlist to a new user at each round, whose appreciation is revealed through multiple steps. In particular, every partial observation corresponds to a song in the playlist, and the associated reward is positive if the user listens to that song and non-positive otherwise. The cumulative reward provided by recommending a playlist to a single user corresponds to the sum of the reward terms from all the playlist songs. Notice that the playlist cannot be trivially modeled as a collection of independent songs, as their order in the playlist affects the user's behavior. In the classical delayed-feedback bandit setting, the feedback on the recommended playlist is obtained only once the user finishes listening to the entire playlist. However, the platform monitors whether

every song is listened to or skipped by the user. Therefore, clues on the performances of the recommended arm can be exploited *before* the user finishes the playlist.

Another scenario captured by the TP-MAB framework is the evaluation of medical treatments taking place over a long period of time. In this setting, the per-round reward corresponds to the patient's state of health at each daily/weekly medical check, and the goal is to find the treatment providing the greatest overall benefit to the patient. In the case of severe pathologies, such as cancer, this type of *partial information* would span several months if not years, providing valuable insights that would be otherwise ignored. Applying a standard delayed-MAB approach to this scenario, *i.e.*, taking decisions only at the end of each treatment cycle, could negatively affect the time required to select an effective medical treatment. In this type of setting, we argue that the partial information provided by patients in periodic medical checks should be used to speed up the learning process. Other examples of real-world scenarios which can be modeled through the TB-MAB setting are provided in Appendix A.2.3.

Other Related Works. In Section 1.1 we introduced part of the literature realting to both dynamic pricing and multi-armed bandit problems. Now, we provide additional works relating to the multi-armed bandit framework, and specifically, to the delayed feedback bandits. To the best of our knowledge, ours is the first work addressing a bandit problem in which the reward from a pull is partitioned across multiple rounds. The most related works concern the Delayed-MAB setting, such as the seminal paper by Joulani et al. [2013], which summarizes the known results on the regret upper bounds of online learning algorithms. They also provide a modification of the well-known UCB1 algorithm from Auer et al. [2002a] for the delayed-feedback setting, called Delayed-UCB1. More recently, a variety of delayed-feedback scenarios were studied investigating directions different from ours, such as linear and contextual (Arya and Yang [2020], Vernade et al. [2020a], Zhou et al. [2019]), non-stationary (Vernade et al. [2020b]) bandits under delayed feedback. Pike-Burke et al. [2018] and Cesa-Bianchi et al. [2018] also analyze the case of delayed, aggregated, and anonymous feedback. For clarity, we remark that, in our work, per-round rewards corresponding to different pulls can be received in the same round, and it is known from which arm they were generated. Many works apply bandits to practical scenarios, *e.g.*, scheduling Cayci et al. (2019), advertising Nuara et al. (2018); Castiglioni et al. (2022d); Nuara et al. (2022), pricing Trovò et al. (2018), and delayed feedback settings Vernade et al. (2017).

Works from the bandit literature, such as the ones by Dudik et al. [2011], Desautels et al. [2014], Neu et al. [2013], rely on known constant delays or maximum delay values. Similarly, in our work, we assume a maximum finite delay equal to $\tau_{\max}$, which is compliant with the real-world scenarios we aim at modeling, *e.g.*, in the above example of playlist recommendations, an infinite $\tau_{\max}$ would correspond to a playlist of an infinite number of songs. According to the terminology used in the delayed-MAB literature, our setting is *uncensored*, meaning that the reward provided by a given action is eventually observed after a finite maximum delay. Conversely, many works in the field, such as, *e.g.*, Manegueu et al. [2020] and Vernade et al. [2017], deals with random delays from an unbounded distribution with finite expectation.

4.1 Model

Consider a MAB problem with $K \in \mathbb{N}^*$ arms, over a time horizon of $T \in \mathbb{N}^*$ rounds. At every round $t \in [T]$, the learner pulls an arm $i \in \mathcal{A} = [K]$ and, from the pull of that arm, gets a *per-round reward* $w^i_{t,m-t+1}$ at every round $m \in \{t, \dots, t + \tau_{\max} - 1\}$, where $\tau_{\max} \in \mathbb{N}^*$ is the time span over which the reward is partitioned.[1] In particular, $\tau_{\max} - 1$ is the maximum delay affecting the observation of a per-round reward, whose value is known to the learner. Therefore, at round $t + \tau_{\max} - 1$, the cumulative reward from pulling arm i at round t is completely collected by the learner. Furthermore, we denote by $\boldsymbol{w}^i_t = (w^i_{t,1}, \dots, w^i_{t,\tau_{\max}})$ the vector of per-round rewards collected from pulling arm i at round t. For every $j \in [\tau_{\max}]$, the per-round reward $w^i_{t,j}$ is a realization of a random variable $W^i_{t,j}$ with support $[\underline{W}^i_j, \overline{W}^i_j]$. The cumulative reward collected from pulling arm i at round t is denoted by r^i_t, and it is the realization of the random variable $R^i_t := \sum_{j=1}^{\tau_{\max}} W^i_{t,j}$, with support $[\underline{R}^i, \overline{R}^i]$, where $\underline{R}^i := \sum_{j=1}^{\tau_{\max}} \underline{W}^i_j$, and $\overline{R}^i := \sum_{j=1}^{\tau_{\max}} \overline{W}^i_j$. For every $i \in \mathcal{A}$ and $t \in [T]$, we assume that the variables R^i_t are independent with mean $\mu_i := \mathbb{E}[R^i_t]$.[2]

A policy $\mathfrak{U}$ is an algorithm that at each round t chooses an arm $i_t \in [K]$. The performance of a policy $\mathfrak{U}$ is evaluated in terms of *pseudo-regret*, defined as the cumulative loss due to playing suboptimal arms during the

[1] We denote by $[n]$ the set $\{1, \dots, n\}$

[2] W.l.o.g., we assume $\underline{X}^i_j = 0, \forall i \in [K], \forall j \in [\tau_{\max}]$.

time horizon T, formally:

$$R^T(\mathfrak{U}) = T\mu^* - \mathbb{E}\left[\sum_{t=1}^{T} \mu_{i_t}\right],$$

where $\mu^* = \max_{i\in\mathcal{A}}\{\mu_i\}$ is the expected reward of the optimal arm i^*, and the expectation is taken w.r.t. the stochasticity of the policy $\mathfrak{U}$. Notice that we adopt the concept of pseudo-regret as for standard bandits, unlike what is done by Vernade et al. [2017], since our choice allows for a direct comparison with the vast prior work on delayed bandits.

In what follows, we cast the playlist recommendation problem, described in the introduction, in the TP-MAB setting.

Example 4.1 (Playlist Recommendation)**.** *At each round t, a new user enters the platform, which provides a playlist suggestion. The different arms i are the available playlists to suggest, each composed of N songs. Songs are characterized by 4 listening levels (from "skipped" to "complete"), each associated with a different Bernoulli random variable representing the corresponding per-round reward. The vector of realized per-round rewards of song $k \in [N]$ is $(w^i_{t,4(k-1)+1}, w^i_{t,4(k-1)+2}, w^i_{t,4(k-1)+3}, w^i_{t,4(k-1)+4})$. Each variable assumes a value of 1 if the user reaches the corresponding level, and a value of 0 if the user stops listening to the song before that level. The cumulative reward R^i_t for pulling arm i at round t is the sum of the rewards from the songs in the playlist, and the time span over which the platform observes the reward is $\tau_{\max} = 4N$.*

We show that the TP-MAB problem has a lower-bound on the regret of the same order of the delayed-feedback bandit problem. The rationale is that no better lower bound is possible as delayed-feedback MABs with a finite delay are a subclass of TP-MABs whose reward vector $\boldsymbol{w}^i_t$ has a single non-zero element for each $i \in \mathcal{A}$ and $t \in [T]$. Most interestingly, the worst-case instance for the regret lower bound in the TP-MAB setting is the delayed-feedback bandit.

Theorem 4.1. *The regret of any uniformly efficient policy $\mathfrak{U}$ applied to the TP-MAB problem is bounded from below by:*

$$\liminf_{T\to+\infty} \frac{R^T(\mathfrak{U})}{\ln T} \geq \sum_{i:\mu_i<\mu^*} \frac{\Delta_i}{KL\left(\frac{\mu_i}{\overline{R}_{\max}}, \frac{\mu^*}{\overline{R}_{\max}}\right)}, \tag{4.1}$$

where $\Delta_i := \mu^* - \mu_i$ is the expected loss suffered by the learner if the arm i is chosen instead of the optimal one i^*, $\overline{R}_{\max} := \max_{i\in[K]} \overline{R}^i$, and $KL(p, q)$

is the Kullback-Leibler divergence between Bernoulli r.v. with means p and q.[3]

Proof. At first, notice that learning the optimal arm in a TP-MAB problem $\mathcal{P}$ for rewards R_t^i taking values over a generic finite domain $[0, \overline{R}^i]$, having range $\overline{R}^i$, is equivalent to the problem of learning in a TP-MAB problem $\mathcal{P}'$ with reward $\frac{R_t^i}{\overline{R}_{\max}}$ having domain $[0, 1]$. Indeed, from a learning perspective, distinguish between two arms in the first setting requires the same sample complexity of distinguish between two arms in the second one. The expected reward of the i-th arm of the $\mathcal{P}'$ problem is $(\mu_i)' = \frac{\mu_i}{\overline{R}_{\max}}$ and the one corresponding to the optimal arm is $(\mu^*)' = \frac{\mu^*}{\overline{R}_{\max}}$.

Let us consider for each problem $\mathcal{P}$ in the class of TP-MAB problems, its corresponding $\mathcal{P}'$ one. For each $\mathcal{P}'$, we build a corresponding Delayed-MAB equivalent problem, by delaying all the intermediate rewards corresponding to a pull at round t to the round $t + \tau_{\max} - 1$. Therefore, using the results on the lower bound of the Delayed-MAB problems provided by Vernade et al. [2017] (Lemma 15) we have that:

$$\lim_{T \to +\infty} \inf \frac{\mathbb{E}[N_i(T)]}{\log(T)} \geq \frac{1}{KL\left(\frac{\mu_i}{\overline{R}_{\max}}, \frac{\mu^*}{\overline{R}_{\max}}\right)}, \tag{4.2}$$

where $\mathbb{E}[N_i(T)]$ is the expected number of times an arm i is selected over a time horizon of T by the policy $\mathfrak{U}$. Due to the equivalence depicted above, this result holds also for the original problems $\mathcal{P}$ in the class of TP-MAB problems. From the fact that $R^T(\mathfrak{U}) = \Delta_i \mathbb{E}[N_i(T)]$ and summing over the suboptimal arms, i.e., $i \neq i^*$, we get the theorem statement. □

Notice that the lower bound holds for general TP-MAB problems. In the following section, we show that focusing on a broad subset of instances of practical interest, we can design algorithms with a better regret upper bound.

4.2 α-Smoothness Property

From Theorem 4.1, we know that we cannot design algorithms with regret upper bounds better than those of the algorithms for the delayed-feedback bandit setting. Nonetheless, in practice, collecting per-round rewards can provide useful information on the cumulative reward of an arm. However,

[3] An uniformly efficient policy chooses the suboptimal arms on average $o(t^a)$ times ($0 < a < 1$) over t rounds.

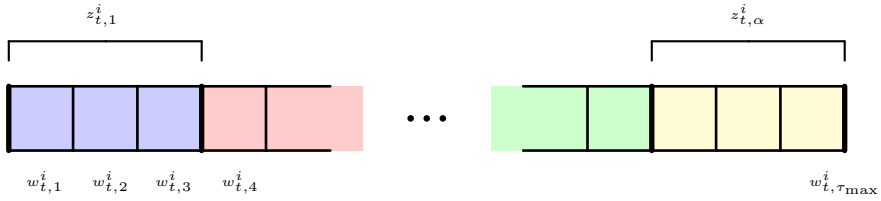


Figure 4.1: *Example of α-smooth reward with $\phi = 3$.*

as already pointed out by Manegueu et al. [2020] for the standard delayed-feedback setting, zero rewards are ambiguous since they do not give any information on future rewards. In the general setting, small per-round rewards observed in the first rounds after the pull are not much informative to bound the values of future ones. To avoid this, we focus on those problems in which the maximum reward realized over a few rounds cannot exceed a fraction of the maximum reward $\overline{R}^i$.

Let us consider $\alpha \in [\tau_{\max}]$ s.t. α is a factor of $\tau_{\max}$, *i.e.*, $\frac{\tau_{\max}}{\alpha} =: \phi$ and $\phi \in \mathbb{N}$.[4] Let us define the vector $\boldsymbol{Z}^i_{t,\alpha} := \left(Z^i_{t,1}, \dots, Z^i_{t,\alpha}\right)$ whose element $Z^i_{t,k}$ is the random variable corresponding to the sum of a set of consecutive per-round rewards of cardinality ϕ. Formally, for every $k \in [\alpha]$:

$$Z^i_{t,k} := \sum_{j=(k-1)\phi+1}^{k\phi} W^i_{t,j}. \tag{4.3}$$

The support of $Z^i_{t,k}$ is denoted by $[\underline{Z}^i_{\alpha,k}, \overline{Z}^i_{\alpha,k}]$, where $\underline{Z}^i_{\alpha,k} := \sum_{j=(k-1)\phi+1}^{k\phi} \underline{W}^i_j$, and $\overline{Z}^i_{\alpha,k} := \sum_{j=(k-1)\phi+1}^{k\phi} \overline{W}^i_j$. Intuitively, the α-smoothness property states that the elements in $\boldsymbol{Z}^i_{t,\alpha}$ are independent and that, when $\alpha > 1$, the maximum reward $\overline{R}^i$ of a pull cannot be realized in a single time span corresponding to a $Z^i_{t,k}$ element. Formally:

Definition 4.1 (α-smoothness)**.** *In the TP-MAB setting, for $\alpha \in [\tau_{\max}]$, we say that the reward is* α-smooth *if and only if $\frac{\tau_{\max}}{\alpha} = \phi$, with $\phi \in \mathbb{N}$, and, for each $k \in [\alpha]$, the random variables $Z^i_{t,k}$ are independent and s.t. $\overline{Z}^i_{\alpha,k} = \overline{Z}^i_{\alpha} = \frac{\overline{R}^i}{\alpha}$.*

An example of α-smooth environment with $\phi = 3$ is presented in Figure 4.1, where colors denote the elements $z^i_{t,k}$ that are the realizations of the variables $Z^i_{t,k}$.

[4] We assume α is a factor of $\tau_{\max}$ for the sake of presentation. The following results also hold for generic $\alpha \in [\tau_{\max}]$.

Consider the extreme values of parameter α. When $\alpha = 1$, the reward has no constraint on how it distributes over time. This scenario includes the delayed-feedback bandit setting in which the cumulative reward provided by the arm pulled at t is entirely collected at a single round (including the last possible round $t + \tau_{\max} - 1$). Note that, in this case, at each round before $t + \tau_{\max} - 1$, the sum of the future per-round rewards is in the range $[0, \overline{R}^i]$. Conversely, when $\alpha = \tau_{\max}$, the vector of aggregated rewards coincides with the vector of per-round rewards, *i.e.*, $\boldsymbol{Z}^i_{t,\tau_{\max}} = \boldsymbol{W}^i_t$, and each per-round reward is at most $\overline{W}^i_j = \overline{R}^i/\tau_{\max}$. Thus, observing low rewards in the first rounds after the pull provides useful information on the actual cumulative reward. In particular, after observing the first $n < \tau_{\max}$ per-round rewards, we know that the cumulative reward achievable in the following rounds is in the range $[0, \frac{\tau_{\max}-n}{\tau_{\max}}\overline{R}^i]$. This information dramatically reduces the uncertainty on the future rewards w.r.t. a setting without smooth rewards (*e.g.*, $\alpha = 1$). The α-smoothness property characterizes those setting where not gaining much in the first rounds precludes the possibility of achieving the maximum possible reward over the entire interval.

Consider the playlist recommendation problem in Example 4.1. Since the reward corresponding to a song is composed of 4 Bernoulli variables and has a maximum of $\overline{Z}^i_\alpha = 4$, α-smoothness holds with $\alpha = \frac{\overline{R}^i}{\overline{Z}^i_\alpha} = \frac{4N}{4} = N$.

Assuming α-smoothness, we have a lower bound of:

Theorem 4.2. *The regret of any uniformly efficient policy $\mathfrak{U}$ applied to the TP-MAB problem with the α-smoothness property is bounded from below by:*

$$\liminf_{T\to+\infty} \frac{R^T(\mathfrak{U})}{\ln T} \geq \sum_{i:\mu_i<\mu^*} \frac{\Delta_i}{\alpha KL\left(\frac{\mu_i}{\overline{R}_{\max}}, \frac{\mu^*}{\overline{R}_{\max}}\right)}. \tag{4.4}$$

Proof. The proof follows the steps provided for Theorem 2.2 in the work by Bubeck and Cesa-Bianchi [2012] and generalize them to the setting in which multiple rewards, *i.e.*, α, are earned by a single arm pull.

Let us define an auxiliary MAB setting in which:

- only two arms with expected value μ_1 and μ_2, with $\mu_2 < \mu_1 < 1$;
- all the arm have maximum reward equal to $R^i_t = \overline{R}_{\max}$;
- the reward $Z^i_{t,k}$ are i.i.d. over $k \in \{1, \ldots, \alpha\}$, meaning that the expected value of each of the element is $\frac{\mu_i}{\alpha}$;

- the reward are $Z^i_{t,k} \in \{0, \frac{\overline{R}_{\max}}{\alpha}\}$, *i.e.*, the reward are Bernoulli scaled by a factor $\frac{\overline{R}_{\max}}{\alpha}$;

- pulling an arm at time t provides α reward for the arm $\{Z^i_{t,1}, \dots, Z^i_{t,\alpha}\}$, all observed by the learner at the time of the pull.

Let us remark that determining the optimality of an arm in this problem is no harder than the one in which the reward is spread over the period $\{t, \dots, t + \tau_{\max} - 1\}$. Therefore, the derivation of a lower bound for this problem would also provide a lower bound for the original TP-MAB setting with α-smoothness. Moreover, let us recall that learning in a problem where the reward are scaled by a factor $\frac{\overline{R}_{\max}}{\alpha}$, similarly to what has been done in Theorem 4.1, does not change the complexity of learning. From now on, we will consider as expected value of the two arms $\mu_{Z_1} := \frac{\mu_1}{\overline{R}_{\max}}$ and $\mu_{Z_2} := \frac{\mu_2}{\overline{R}_{\max}}$. Therefore, to compute the expected value of number of times an algorithm pulls the suboptimal arm $\mathbb{E}[N_2(T)]$ we can also use the scaled rewards. In what follows, we prove that the lower bound for the auxiliary problem for any uniformly efficient policy $\mathfrak{U}$.

Overall proof idea: Let us consider a second instance of the above defined MAB such that arm 2 is optimal and $\mu_{Z_1} < \mu'_{Z_2} < 1$. We refer to it as the modified bandit. Let $\varepsilon > 0$, since $w \mapsto KL(\mu_{Z_2}, w)$ is continuous one can find $\mu'_{Z_2} \in (\mu_{Z_1}, 1)$ such that:

$$KL\big(\mu_{Z_2}, \mu'_{Z_2}\big) \leq (1 + \varepsilon) KL(\mu_{Z_2}, \mu_{Z_1}). \tag{4.5}$$

In what follows, we use the notation $\mathbb{E}'$, $\mathbb{P}'$ to denote the expected value and probability computed in the second bandit instance. The goal is to compare the behavior of the forecaster on the initial and modified bandits. The idea of the proof is to show that, with a big enough probability, the forecaster is not able to distinguish between the two problems. Then, using the fact that the forecaster is uniformly efficient by hypothesis, we show that the algorithm does not make too many mistake on the modified bandit and, in particular, provide a lower bound on the number of times the optimal arm is played. This reasoning implies a lower bound on the number of times the suboptimal arm 2 is played in the initial problem.

First step: $\mathbb{P}(C_t) = o(1)$
Let us define, for $s \in \{1, \dots, t\}$, the empirical estimate of $KL\big(\mu_{Z_2}, \mu'_{Z_2}\big)$ at

round t when the arm 2 is pulled s times:

$$\widehat{KL}_{\alpha s} := \sum_{n=1}^{s}\sum_{k=1}^{\alpha} \ln \frac{\mu_{Z_2} Z_{n,k}^2 + (1-\mu_{Z_2})(1-Z_{n,k}^2)}{\mu'_{Z_2} Z_{n,k}^2 + (1-\mu'_{Z_2})(1-Z_{n,k}^2)}. \tag{4.6}$$

We introduce the following event linking the behavior of the forecaster on the initial and modified bandits:

$$C_t := \Big\{ \alpha N_2(t) < f_t \quad \text{and} \quad \widehat{KL}_{\alpha N_2(t)} \leq (1-\varepsilon/2)\ln t \Big\}, \tag{4.7}$$

where $f_t = \frac{1-\varepsilon}{KL(\mu_{Z_2}, \mu'_{Z_2})} \ln t$. Following the proof of Theorem 2.2 from Bubeck and Cesa-Bianchi [2012], we have:

$$\mathbb{P}'(C_t) = \mathbb{E}[1_{C_t} \exp\Big(-\widehat{KL}_{\alpha N_2(t)}\Big)] \geq e^{-(1-\varepsilon/2)\ln t} \mathbb{P}(C_t), \tag{4.8}$$

where we used the change of measure identity for the first equality and use the fact that $\widehat{KL}_{\alpha N_2(t)} \leq (1-\varepsilon/2)\ln t$ in C_t.[5] Combining Equation (4.8), the definition of C_t, and using the Markov's inequality, we have:

$$\begin{aligned} \mathbb{P}(C_t) \leq t^{(1-\varepsilon/2)} \mathbb{P}'(C_t) &\leq t^{(1-\varepsilon/2)} \mathbb{P}'(\alpha N_2(t) < f_t) \leq \\ &\leq t^{(1-\varepsilon/2)} \frac{\mathbb{E}'\left[t - N_2(t)\right]}{t - f_t/\alpha} = o(1), \end{aligned} \tag{4.9}$$

where with $o(1)$ we denote a quantity whose limit for $t \to +\infty$ is 0 and we used the fact that the policy $\mathfrak{U}$ is uniformly efficient, *i.e.*, $\mathbb{E}'[T_2(t)] = o(t^\beta)$ with $\beta < 1$.

Second step: $\mathbb{P}(\alpha N_2(t) \leq f_t) = o(1)$
Using the Third step of the proof of Theorem 2.2 from Bubeck and Cesa-Bianchi [2012], we get:

$$o(1) = \mathbb{P}(C_t) \leq \mathbb{P}\Bigg(\underbrace{\alpha T_2(t) < f_t}_{E_1} \wedge \underbrace{\frac{KL(\mu_{Z_2}, \mu'_{Z_2})}{(1-\varepsilon)\ln t} \max_{s < f_t/\alpha} \widehat{KL}_{\alpha s} \leq \frac{1-\varepsilon/2}{1-\varepsilon} KL(\mu_{Z_2}, \mu'_{Z_2})}_{E_2} \Bigg).$$

[5] For any event A in the σ-algebra generated by $\{Z_{n,k}^2\}_{n\in\{1,\dots,s\},k\in\{1,\dots,\alpha\}}$ holds that $\mathbb{P}'(A) = \mathbb{E}\Big[1_A \exp\Big(-\widehat{KL}_{\alpha N_2(t)}\Big)\Big]$.

Using the strong law of large numbers the event E_2 is s.t. $\lim_{t\to+\infty}\mathbb{P}(E_2)=1$, we infer that $\mathbb{P}(E_1)=\mathbb{P}(\alpha N_2(t)<f_t)=o(1)$, and that for $t\to+\infty$ we have $\mathbb{E}[N_2(t)]>f_t/\alpha$.

Final step: Using Equation (4.5) we have that, for $t\to+\infty$:

$$\mathbb{E}[N_2(t)]>f_t/\alpha=\frac{1-\varepsilon}{\alpha KL(\mu_{Z_2},\mu'_{Z_2})}\ln t\geq\frac{1-\varepsilon}{\alpha(1+\varepsilon)KL(\mu_{Z_2},\mu_{Z_1})}\ln t, \tag{4.10}$$

where the theorem statement follows from the arbitrarity of the value of ε, substituting μ_{Z_1} with $\frac{\mu^*}{R_{\max}}$ and μ_{Z_2} with $\frac{\mu_2}{R_{\max}}$, and summing over all the suboptimal arms.

□

We remark that this bound is tighter than the one provided in Theorem 4.1 by a multiplicative factor of $1/\alpha$.

4.3 Algorithms for the TP-MAB Setting

We propose two novel algorithms, namely Temporally-Partitioned rewards UCB with Fictitious Realizations (`TP-UCB-FR`) and Temporally-Partitioned rewards Element-Wise UCB (`TP-UCB-EW`), for the TP-MAB problem, which aim at maximizing the cumulative reward and exploit the α-smoothness property to do that. From now on, we denote the two corresponding policies by $\mathfrak{U}_{\texttt{FR}}$ and $\mathfrak{U}_{\texttt{EW}}$, respectively.

4.3.1 The `TP-UCB-FR` Algorithm

The pseudo-code of `TP-UCB-FR` is provided in Algorithm 4.1. The rationale is to use the rewards coming from not fully-realized reward vectors by replacing the missing elements with fictitious realizations. At round t, fictitious reward vectors are associated to each arm pulled in the time span $H:=\{t-\tau_{\max}+1,\dots,t-1\}$. We denote them by $\tilde{\boldsymbol{w}}_h^i=[\tilde{w}_{h,1}^i,\dots,\tilde{w}_{h,\tau_{\max}}^i]$ with $h\in H$, where $\tilde{w}_{h,j}^i:=w_{h,j}^i$, if $h+j\leq t$, and $\tilde{w}_{h,j}^i=0$, if $h+j>t$. The corresponding fictitious cumulative reward is $\tilde{r}_h^i:=\sum_{j=1}^{\tau_{\max}}\tilde{w}_{h,j}^i$. The algorithm takes as input the smoothness $\alpha\in[\tau_{\max}]$, and the maximum delay $\tau_{\max}$.[6] During the initialization phase, all arms are pulled once (Line 3).

[6] If these information are not available one should use $\alpha=1$, meaning we are not assuming any structure over the reward, and use as $\tau_{\max}$ the largest delay observed so far.

Algorithm 4.1 TP-UCB-FR

1: **Input**: $\alpha \in [\tau_{\max}]$, $\tau_{\max} \in \mathbb{N}^*$
2: **for** $t \in \{1, \dots, K\}$ **do** ▷ init phase
3: Pull arm $i_t = t$
4: **end for**
5: **for** $t \in \{K+1, \dots, T\}$ **do** ▷ loop phase
6: **for** $i \in \{1, \dots, K\}$ **do**
7: Compute $\hat{R}^i_{t-1}$ and c^i_{t-1} as in Eq.s (4.11)-(4.12)
8: $u^i_{t-1} \leftarrow \hat{R}^i_{t-1} + c^i_{t-1}$
9: **end for**
10: Pull arm $i_t = \arg\max_{i \in [K]} u^i_{t-1}$
11: Observe $w^{i_h}_{h,t-h+1}$ for $h \in \{t - \tau_{\max} + 1, \dots, t\}$
12: **end for**

After that, at each round t, it computes the estimated expected reward for each arm i:

$$\hat{R}^i_{t-1} := \frac{1}{n^i_{t-1}} \left(\sum_{h=1}^{t-\tau_{\max}} r^i_h \mathbf{1}_{\{i_h = i\}} + \sum_{h \in H} \tilde{r}^i_h \mathbf{1}_{\{i_h = i\}} \right), \tag{4.11}$$

where $n^i_{t-1} := \sum_{h=1}^{t-1} \mathbf{1}_{\{i_h = i\}}$ is the number of times arm i has been pulled by the policy up to round $t-1$, and the confidence interval:

$$c^i_{t-1} := \overline{R}^i \sqrt{\frac{2 \ln(t-1)}{\alpha n^i_{t-1}}} + \frac{\phi(\alpha+1)\overline{R}^i}{2 n^i_{t-1}}. \tag{4.12}$$

Finally, it pulls the arm with the largest upper confidence bound u^i_{t-1} (Line 10), and observes its reward (Line 11).

We provide the following upper bound on the regret:

Theorem 4.3. *In the TP-MAB setting with α-smooth reward, the pseudo-regret of* TP-UCB-FR *after T rounds is:*

$$R^T(\mathfrak{U}_{\text{FR}}) \leq \sum_{i:\mu_i < \mu^*} \frac{4(\overline{R}^i)^2 \ln T}{\alpha \Delta_i} \left(1 + \sqrt{1 + \frac{\alpha(\alpha+1)\phi\Delta_i}{2\overline{R}^i \ln T}} \right) + (\alpha+1)\phi \sum_{i:\mu_i < \mu^*} \overline{R}^i + \left(1 + \frac{\pi^2}{3}\right) \sum_{i:\mu_i < \mu^*} \Delta_i.$$

Proof. Let us define the true empirical mean of the cumulative reward of arm i computed over n^i_t samples as follows:

$$\hat{R}^{i,\text{true}}_t := \frac{1}{n^i_t} \sum_{h=1}^{t} r^i_h \mathbf{1}_{\{i_h = i\}}.$$

We aim to bound the difference between $\hat{R}_t^{i,\text{true}}$ and the approximated empirical mean of the cumulative reward $\hat{R}_t^i$ from arm i computed over n_t^i samples as in the TP−UCB−FR algorithm. Formally, we have:

$$\begin{aligned}\hat{R}_t^{i,\text{true}} - \hat{R}_t^i &= \frac{1}{n_t^i}\sum_{h=1}^{t}\sum_{j=1}^{\tau_{\max}}\big(w_{h,j}^i - \tilde{w}_{h,j}^i\big)\mathbf{1}_{\{i_h=i\}} \leq \frac{1}{n_t^i}\sum_{h=1}^{t}\sum_{j=1}^{\tau_{\max}}\big(w_{h,j}^i - x\tilde{w}_{h,j}^i\big) \\ &= \frac{1}{n_t^i}\sum_{h=\max\{1,t-\tau_{\max}+2\}}^{t}\sum_{j=t-h+2}^{\tau_{\max}} w_{h,j}^i \qquad (4.13) \\ &\leq \frac{1}{n_t^i}\sum_{j=1}^{\alpha}\phi j\frac{\overline{R}^i}{\alpha} \qquad (4.14) \\ &= \frac{\phi}{n_t^i}\frac{\overline{R}^i}{\alpha}\sum_{j=1}^{\alpha} j = \frac{\phi}{n_t^i}\frac{\overline{R}^i}{\alpha}\frac{\alpha(\alpha+1)}{2} = \frac{\overline{R}^i(\alpha+1)\phi}{2n_t^i},\end{aligned}$$

where, Equation (4.13) is due to the fact that $\underline{R}^i = 0$ for each $i \in [K]$, and the inequality in Equation (4.14) is due to the α-smoothness of the environment.

Following the proof of Theorem 1 by Auer et al. [2002a], we bound the expected number of time a suboptimal arm is pulled as follows:

$$\mathbb{E}[N_i(t)] \leq \ell + \sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_i=\ell}^{t-1}\mathbb{P}\Big\{\Big(\hat{R}_{t,s}^* + c_{t,s}^*\Big) \leq \Big(\hat{R}_{t,s_i}^i + c_{t,s_i}^i\Big)\Big\}, \quad (4.15)$$

where $\hat{R}_{t,s}^*$ and $c_{t,s}^*$ are the empirical mean computed as in the TP−UCB−FR algorithm and the confidence bound, respectively, of the optimal arm i^* in the case s pulls occurred in the first t rounds, and, $\hat{R}_{t,s_i}^i$ and c_{t,s_i}^i are the empirical mean computed as in the TP−UCB−FR algorithm and the confidence bound, respectively, of the arm i in the case s_i pulls occurred in the first t rounds.

Equation (4.15) implies that at least one of the following holds:

$$\hat{R}_{t,s}^* \leq \mu^* - c_{t,s}^*, \qquad (4.16)$$

$$\hat{R}_{t,s_i}^i \geq \mu_i + c_{t,s_i}^i, \qquad (4.17)$$

$$\mu^* < \mu_i + 2c_{t,s_i}^i. \qquad (4.18)$$

Let us focus on Equation (4.16). We have that:

$$\mathbb{P}\Big(\hat{R}_{t,s}^* - \mu^* \leq -c_{t,s}^*\Big) = \mathbb{P}\Big(\hat{R}_{t,s}^{*,\text{true}} - \mu^* \leq -c_{t,s}^* + \hat{R}_{t,s}^{*,\text{true}} - \hat{R}_{t,s}^*\Big) \leq$$

$$\leq \mathbb{P}\left(\hat{R}_{t,s}^{*,\text{true}} - \mu^* \leq -c_{t,s}^* + \frac{\overline{R}^i(\alpha+1)\phi}{2s}\right) =$$

$$= \mathbb{P}\left(\hat{R}_{t,s}^{*,\text{true}} - \mu^* \leq -\overline{R}^* \sqrt{\frac{2\ln t}{\alpha s}}\right) \leq$$

$$\leq \exp\left\{\frac{2\left(\overline{R}^* \sqrt{\frac{2\ln t}{\alpha s}}\right)^2 s^2}{\sum_{l=1}^{\alpha s}\left(\frac{\overline{R}^*}{\alpha}\right)^2}\right\} \leq e^{-4\ln t} \leq t^{-4}, \tag{4.19}$$

where $c_{t,s}^* := \overline{R}^* \sqrt{\frac{2\ln(t)}{\alpha s}} + \frac{\overline{R}^i(\alpha+1)\phi}{2s}$, $\overline{R}^* := \overline{R}^{i^*}$, $\hat{R}_{t,s}^{*,\text{true}}$ is the empirical mean of the optimal arm i^* in the case s pulls occurred in the first t rounds, and we use the Hoeffding inequality in Equation (4.19).

Similarly, Equation (4.17) is bounded by:

$$\mathbb{P}\left(\hat{R}_{t,s_i}^i - \mu_i \geq c_{t,s_i}^i\right) \leq \mathbb{P}\left(\hat{R}_{t,s}^{i,\text{true}} - \mu_i \geq \overline{R}^i \sqrt{\frac{2\ln t}{\alpha s_i}}\right) \leq$$

$$\leq e^{-4\ln t} = t^{-4}, \tag{4.20}$$

where we used the fact that $\hat{R}_{t,s_i}^{i,\text{true}} \geq \hat{R}_{t,s}^i$ by construction of the latter, and we used the Hoeffding inequality to derive Equation (4.20).

Define:

$$\ell := \left\lceil \frac{\overline{R}^i(\alpha+1)\phi}{\Delta_i} + \frac{4(\overline{R}^i)^2 \ln t}{\alpha\Delta_i^2}\left(1 + \sqrt{1 + \frac{\alpha(\alpha+1)\phi\Delta_i}{2\overline{R}^i \ln t}}\right)\right\rceil. \tag{4.21}$$

We have that the following holds:

$$\mu^* \geq \mu_i + 2c_{t,s}^i$$

$$\Delta_i \geq 2\left(\overline{R}^i \sqrt{\frac{2\ln t}{\alpha s_i}} + \phi\frac{\overline{R}^i(\alpha+1)}{2s_i}\right)$$

$$s_i^2\left(\frac{\Delta_i^2}{4}\right) - 2s_i\left(\frac{\Delta_i \overline{R}^i(\alpha+1)}{4}\phi + \frac{(\overline{R}^i)^2 \ln t}{\alpha}\right) + \phi^2\frac{(\overline{R}^i)^2(\alpha+1)^2}{4} \geq 0$$

$$s_i \geq \frac{4}{\Delta_i^2}\left(\frac{\Delta_i \overline{R}^i(\alpha+1)}{4}\phi + \frac{(\overline{R}^i)^2 \ln t}{\alpha} + \right.$$

$$\left. + \sqrt{\frac{(\overline{R}^i)^4 \ln^2 t}{\alpha^2} + \frac{\Delta_i(\overline{R}^i)^3(\alpha+1)\phi \ln t}{2\alpha}}\right)$$

$$s_i \geq \frac{\overline{R}^i(\alpha+1)}{\Delta_i}\phi + \frac{4(\overline{R}^i)^2 \ln t}{\Delta_i^2 \alpha}\left(1+\sqrt{1+\frac{\Delta_i \alpha(\alpha+1)\phi}{2\overline{R}^i \ln t}}\right),$$

and, therefore, for $s_i \geq \ell$ the inequality in Equation (4.18) is always false.

Finally, summing up the results derived above and using ℓ as defined in Equation (4.21), we have:

$$\begin{aligned}
\mathbb{E}[N_i(t)] &\leq \left\lceil \frac{\overline{R}^i(\alpha+1)}{\Delta_i}\phi + \frac{4(\overline{R}^i)^2 \ln t}{\alpha\Delta_i^2}\left(1+\sqrt{1+\frac{\alpha(\alpha+1)\phi\Delta_i}{2\overline{R}^i \ln t}}\right)\right\rceil + \\
&\quad + \sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_i=\ell}^{t-1}\left[\mathbb{P}\left(\hat{R}^*_{t,s} - \mu^* \leq -c^*_{t,s}\right) + \mathbb{P}\left(\hat{R}^i_{t,s_i} - \mu_i \geq c^i_{t,s_i}\right)\right] \\
&\leq 1 + \frac{\overline{R}^i(\alpha+1)}{\Delta_i}\phi + \frac{4(\overline{R}^i)^2 \ln t}{\alpha\Delta_i^2}\left(1+\sqrt{1+\frac{\alpha(\alpha+1)\phi\Delta_i}{2\overline{R}^i \ln t}}\right) + \\
&\quad + 1 + \sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_i=\ell}^{t-1} 2t^{-4} \\
&\leq \frac{\overline{R}^i(\alpha+1)}{\Delta_i}\phi + \frac{4(\overline{R}^i)^2 \ln t}{\alpha\Delta_i^2}\left(1+\sqrt{1+\frac{\alpha(\alpha+1)\phi\Delta_i}{2\overline{R}^i \ln t}}\right) + 1 + \frac{\pi^2}{3}.
\end{aligned}$$

The theorem follows from $R^T(\mathfrak{U}_{\text{FR}}) = \sum_{i:\mu_i<\mu^*} \Delta_i \mathbb{E}[N_i(T)]$. □

The dominant term in T has the order of $\mathcal{O}\left(\sum_{i:\mu_i<\mu^*} \frac{\overline{R}^2_{\max} \ln T}{\alpha\Delta_i}\right)$, where $\overline{R}_{\max} = \max_i \overline{R}^i$. When $\alpha = 1$, the upper bound scales as the one of classical MAB algorithms in stochastic settings. Notice that the pseudo-regret indirectly depends on $\tau_{\max}$ since $\overline{R}^i$ represents the cumulative reward obtained over $\tau_{\max}$ rounds. Let us compare this result with the one provided in Theorem 4.1 for general TP-MAB problems. Applying to Theorem 4.1 the inequality $KL(p,q) \leq \frac{(p-q)^2}{q(1-q)}$, where for $p, q \in [0,1]$, derived using the fact that $\ln x \leq x - 1$, we get:

$$\liminf_{T\to+\infty} \frac{R^T(\mathfrak{U})}{\ln T} \geq \sum_{i:\mu_i<\mu^*} \frac{\beta}{\Delta_i}, \tag{4.22}$$

where $\beta = \frac{\mu^*}{\overline{R}_{\max}}\left(1 - \frac{\mu^*}{\overline{R}_{\max}}\right)$.

For $\alpha > 4(\overline{R}^i)^2/\beta$, the multiplicative factor in the dominant term of the upper bound provided in Theorem 4.3 is better than that in the lower bound

Algorithm 4.2 `TP-UCB-EW`

1: **Input**: $\alpha \in [\tau_{\max}]$, $\tau_{\max} \in \mathbb{N}^*$
2: **for** $t \in \{1, \dots, K\}$ **do** ▷ init phase
3: Pull arm $i_t = t$
4: **end for**
5: **for** $t \in \{K+1, \dots, T\}$ **do** ▷ loop phase
6: **for** $i \in \{1, \dots, K\}$ **do**
7: **for** $k \in \{1, \dots, \alpha\}$ **do**
8: Compute $\hat{Z}^i_{t-1,k}$ and $c^i_{t-1,k}$ as in Eq.s (4.23)-(4.24)
9: **end for**
10: $u^i_{t-1} \leftarrow \sum_{k=1}^{\alpha} \left(\hat{Z}^i_{t-1,k} + c^i_{t-1,k} \right)$
11: **end for**
12: Pull arm $i_t \in \arg\max_{i \in [K]} u^i_{t-1}$
13: Observe $w^{i_h}_{h,t-h+1}$ for $h \in \{t - \tau_{\max} + 1, \dots, t\}$
14: **end for**

in Theorem 4.1. This suggests that exploiting the α-smoothness provides an improvement over the classical and delayed-feedback MABs.

4.3.2 The `TP-UCB-EW` Algorithm

The pseudo-code of `TP-UCB-EW` is provided in Algorithm 4.2. The key idea is to compute an upper confidence bound for the average of each set of k-th realized aggregated rewards $z^i_{t,k}$ from arm i and use them to build an upper bound on the overall average reward R^i_t. It takes as input the smoothness parameter α, and the maximum delay parameter $\tau_{\max}$. At first, it pulls each arm once (Line 3), while, in the following rounds, it computes the empirical mean:

$$\hat{Z}^i_{t-1,k} := \frac{\sum_{h=1}^{t-k\phi} z^i_{h,k} \mathbf{1}_{\{i_h = i\}}}{n^i_{t-1,k}}, \tag{4.23}$$

where $n^i_{t-1,k} := \sum_{h=1}^{t-k\phi} \mathbf{1}_{\{i_h=i\}}$ is the cardinality of the rewards observed up to round $t-1$ for the k-th element of $\boldsymbol{Z}^i_{t-1,\alpha}$, and the confidence bound:

$$c^i_{t-1,k} := \frac{\overline{R}^i}{\alpha} \sqrt{\frac{2\ln(t-1)}{n^i_{t-1,k}}}. \tag{4.24}$$

We remark that $\hat{Z}^i_{t-1,k} + c^i_{t-1,k}$ is an upper confidence bound for the k-th element of $\boldsymbol{Z}^i_{t-1,\alpha}$. Finally, the algorithm computes the upper bound u^i_{t-1}, summing the bounds above (Line 10), selects the arm i choosing the largest u^i_{t-1} (Line 12), and observes its reward (Line 13).

We provide the following upper bound on the regret:

Theorem 4.4. *In the TP-MAB setting with α-smooth reward, the pseudo-regret of* `TP-UCB-EW` *after T rounds is:*

$$\mathcal{R}_T(\mathfrak{U}_{\texttt{EW}}) \leq \sum_{i:\mu_i<\mu^*} \frac{8(\overline{R}^i)^2 \ln T}{\Delta_i} + \alpha\left(\phi + \frac{\pi^2}{3}\right)\sum_{i:\mu_i<\mu^*}\Delta_i.$$

Proof. Following the same proof strategy of Theorem 4.3, we want to bound the expected value of the number of pulls of suboptimal arms:

$$\mathbb{E}[N_i(t)] \leq l + \sum_{t=1}^{\infty}\sum_{s=1}^{t-1}\sum_{s_i=l}^{t-1}\mathbb{P}\left(\sum_{k=1}^{\alpha}(\hat{Z}^*_{t,k,s} + c^*_{t,k,s}) \leq \sum_{k=1}^{\alpha}(\hat{Z}^i_{t,k,s_i} + c^i_{t,k,s_i})\right),$$

where $\hat{Z}^*_{t,k,s}$ and $c^*_{t,k,s}$ are the empirical mean computed as in `TP-UCB-EW` algorithm and the confidence bound, respectively, of the optimal arm i^* in the case s pulls occurred in the first t rounds, and, $\hat{Z}^i_{t,k,s_i}$ and c^i_{t,k,s_i} are the empirical mean computed as in the `TP-UCB-EW` algorithm and the confidence bound, respectively, of the arm i in the case s_i pulls occurred in the first t rounds. Notice that in this case the number of samples collected from each one of the α aggregated rewards are $\leq s$ and $\leq s_i$, respectively. Moreover, for values of $l > \tau_{\max}$ the quantities regarding the suboptimal arm are estimated using at least one sample, *e.g.*, $0 < n^i_{t,k,s} < s_i$. Conversely, for $s \leq \tau_{\max}$ the optimal arm might have no sample available to estimate the expected value and the bound. However, since the values of the upper confidence bound is set $+\infty$ if no sample is collected, the probability that it is smaller than the one of a suboptimal arm is 0, (*i.e.*, $\mathbb{P}\Big(\sum_{k=1}^{\alpha}(\hat{Z}^*_{t,k,s} + c^*_{t,k,s}) \leq \sum_{k=1}^{\alpha}(\hat{Z}^i_{t,k,s_i} + c^i_{t,k,s_i})\Big) = 0$). As a consequence, the cases in which no sample is available for the optimal bound can be disregarded.

The condition above is satisfied if at least one of the following $2\alpha + 1$ inequalities holds:

$$\hat{Z}^*_{t,k,s} - \mu^*_k \leq -c^*_{t,k,s}, \qquad \forall k \in \{1,\dots,\alpha\} \tag{4.25}$$

$$\hat{Z}^i_{t,k,s_i} - \mu_{i,k} \geq c^i_{t,k,s_i}, \qquad \forall k \in \{1,\dots,\alpha\} \tag{4.26}$$

$$\sum_{k=1}^{\alpha}\mu^*_k - \mu_{i,k} - 2c^i_{t,k,s_i} < 0, \tag{4.27}$$

where $\mu_{i,k} := \mathbb{E}[Z^i_{t,k,s_i}]$ and $\mu^*_k := \mathbb{E}[Z^*_{t,k,s}]$ are the expected value of the aggregated reward Z^i_{t,k,s_i} from arm i, and $Z^{i^*}_{t,k,s}$ from the optimal arm, respectively.

Let us focus on the k-th inequality in Equation (4.25). We have:

$$\mathbb{P}(\hat{Z}^*_{t,k,s} - \mu^*_k \leq -c^*_{t,k,s}) \leq \exp\left\{-\frac{2(n^*_{t,k,s})^2(c^*_{t,k,s})^2}{\sum_{l=1}^{n^*_{t,k,s}}\left(\frac{\overline{R}^*}{\alpha}\right)^2}\right\} \tag{4.28}$$

$$\leq \exp\left\{-\frac{2n^*_{t,k,s}(c^*_{t,k,s})^2\alpha^2}{(\overline{R}^*)^2}\right\} \leq e^{-4\ln t} \leq t^{-4},$$

where $n^*_{t,k,s}$ is the number of samples available for the estimation of the expected value of $Z^*_{t,k,s}$ if we pulled s times the arm i^* at round t. Here, we assume that the estimates have at least one sample. If no samples are available, the original probability in Equation (4.25) is bounded by 0.

Similarly, for the inequalities in Equation (4.26), we have:

$$\mathbb{P}(\hat{Z}^i_{t,k,s_i} - \mu_{i,k} \geq c^i_{t,k,s_i}) \leq \exp\left\{-\frac{2(n^i_{t,k,s_i})^2(c^i_{t,k,s_i})^2}{\sum_{l=1}^{n^i_{t,k,s_i}}(\frac{\overline{R}^i}{\alpha})^2}\right\}$$

$$\leq \exp\left\{-\frac{2n^i_{t,k,s_i}(c^i_{t,k,s_i})^2\alpha^2}{(\overline{R}^i)^2}\right\} \leq e^{-4\ln t} \leq t^{-4}. \tag{4.29}$$

where n^i_{t,k,s_i} is the number of samples available for the estimation of the expected value of Z^i_{t,k,s_i} if we pulled s_i times the arm i at round t.

Define $l = \left\lceil \alpha\phi - 1 + \frac{8(\overline{R}^i)^2\ln t}{\Delta_i^2}\right\rceil$. Notice that $l \geq \tau_{\max}$. We have that the inequality in Equation (4.27) is false. Indeed, we have that:

$$\sum_{k=1}^{\alpha}\left(\mu^*_k - \mu_{i,k} - 2\frac{\overline{R}^i}{\alpha}\sqrt{\frac{2\ln t}{n^i_{t,k,s_i}}}\right) \geq \Delta_i - 2\frac{\overline{R}^i}{\alpha}\sum_{k=1}^{\alpha}\sqrt{\frac{2\ln t}{s_i - k\phi + 1}}$$

$$\geq \Delta_i - 2\alpha\frac{\overline{R}^i}{\alpha}\sqrt{\frac{2\ln t}{s_i - \alpha\phi + 1}} = \Delta_i - 2\overline{R}^i\sqrt{\frac{2\ln t}{s_i - \alpha\phi + 1}}, \tag{4.30}$$

where we used that $\sum_{k=1}^{\alpha}\mu^*_k - \mu_{i,k} = \mu^* - \mu_i = \Delta_i$.

If $s_i \geq \alpha\phi - 1 + \frac{8(\overline{R}^i)^2\ln t}{\Delta_i^2}$, we have that:

$$s_i \geq \alpha\phi - 1 + \frac{8(\overline{R}^i)^2\ln(t)}{\Delta_i^2} \tag{4.31}$$

$$\frac{\Delta_i^2}{4(\overline{R}^i)^2} \geq \frac{2\ln(t)}{s_i - \alpha\phi + 1} \tag{4.32}$$

$$\Delta_i - 2\overline{R}^i \sqrt{\frac{2\ln(t)}{s_i - \alpha\phi + 1}} \geq 0, \tag{4.33}$$

which implies that the inequality in Equation (4.30) is false.

Finally, summing the above results we have that:

$$\begin{aligned}
\mathbb{E}[N_i(t)] &\leq \left\lceil \alpha\phi - 1 + \frac{8(\overline{R}^i)^2 \ln(t)}{\Delta_i^2} \right\rceil + \\
&+ \sum_{t=1}^{\infty} \sum_{s=1}^{t-1} \sum_{s_i=l}^{t-1} \sum_{k=1}^{\alpha} \left[\mathbb{P}(\hat{Z}^*_{t,k,s} - \mu^*_k \leq -c^*_{t,k,s}) + \mathbb{P}(\hat{Z}^i_{t,k,s_i} - \mu_{i,k} \geq c^i_{t,k,s_i}) \right] \leq \\
&\leq \alpha\phi + \frac{8(\overline{R}^i)^2 \ln(t)}{\Delta_i^2} + \sum_{t=1}^{\infty} \sum_{s=1}^{t-1} \sum_{s_i=l}^{t-1} 2\alpha t^{-4} \leq \\
&\leq \frac{8(\overline{R}^i)^2 \ln t}{\Delta_i^2} + \alpha\left(\phi + \frac{\pi^2}{3}\right).
\end{aligned}$$

Recalling that $R^T(\mathfrak{U}_{\texttt{EW}}) = \sum_{i:\mu_i<\mu^*} \Delta_i \mathbb{E}[N_i(T)]$ concludes the proof. □

Focusing on the dominant term in T of the regret bound, we do not have an explicit improvement over the classical and delayed-feedback MAB algorithms. Therefore, in this case, the structure provided by the α-smoothness seems not to affect the regret bound. Hence, from an asymptotic point of view, there is not a clear advantage from having α-smooth rewards. However, the constant term is significantly smaller than that of `TP-UCB-FR` and allows `TP-UCB-EW` to be much more effective than `TP-UCB-FR` to tackle TP-MAB problems with a short time horizon.

4.3.3 Theoretical Results Summary

Finally, we provide a table summarizing the results known in the literature and provided in this chapter. Tables 4.1, 4.2, 4.3 reports the lower and upper bounds on the regret for different algorithms and settings. Notice that the lower bound results hold for $T \to +\infty$. Moreover, in the tables, we denote $\frac{8(\overline{R}^i)^2}{\Delta_i}$ by C_i and $\sum_{i:\mu_i<\mu^*}$ by $\sum_i$. The assumption is that the instantaneous (for the MAB and Delayed-MAB settings) and cumulative (for the TP-MAB setting) rewards have support in $[0, \overline{R}^i]$. Moreover, in the Delayed-MAB setting, the maximum stochastic delay is $\tau_{\max}$ The novel results have been highlighted in blue. `UCB1` does not have guarantees in the Delayed-MAB

	Setting: MAB
Lower bound	$\sum_i \frac{\Delta_i \ln T}{KL\left(\frac{\mu_i}{\overline{R}_{\max}}, \frac{\mu^*}{\overline{R}_{\max}}\right)}$
UCB1	$\sum_i C_i \ln T + \left(1 + \frac{\pi^2}{3}\right) \sum_i \Delta_i$
Delayed-UCB1	$\sum_i C_i \ln T + \left(1 + \frac{\pi^2}{3}\right) \sum_i \Delta_i$
TP-UCB-FR	$\sum_i \frac{C_i}{2} \ln T \left(1 + \sqrt{1 + \frac{2\Delta_i}{\overline{R}^i \ln T}}\right) + 2 \sum_i \overline{R}^i + \left(1 + \frac{\pi^2}{3}\right) \sum_i \Delta_i$
TP-UCB-EW	$\sum_i C_i \ln T + \left(1 + \frac{\pi^2}{3}\right) \sum_i \Delta_i$

Table 4.1: *Summary of the known theoretical results (in black) and original contributions provided in this chapter (in blue) for the MAB setting.*

	Setting: Delayed MAB
Lower bound	$\sum_i \frac{\Delta_i \ln T}{KL\left(\frac{\mu_i}{\overline{R}_{\max}}, \frac{\mu^*}{\overline{R}_{\max}}\right)}$
UCB1	N.a.
Delayed-UCB1	$\sum_i C_i \ln T + \left(1 + \frac{\pi^2}{3} + \tau_{\max}\right) \sum_i \Delta_i$
TP-UCB-FR	$\sum_i \frac{C_i}{2} \ln T \left(1 + \sqrt{1 + \frac{2\Delta_i}{\overline{R}^i \ln T}}\right) + 2\tau_{\max} \sum_i \overline{R}^i + \left(1 + \frac{\pi^2}{3}\right) \sum_i \Delta_i$
TP-UCB-EW	$\sum_i C_i \ln T + \left(\tau_{\max} + \frac{\pi^2}{3}\right) \sum_i \Delta_i$

Table 4.2: *Summary of the known theoretical results (in black) and original contributions provided in this chapter (in blue) for the Delayed MAB setting.*

	Setting: TP-MAB with α-smoothness
Lower bound	$\sum_i \frac{\Delta_i \ln T}{\alpha KL\left(\frac{\mu_i}{\overline{R}_{\max}}, \frac{\mu^*}{\overline{R}_{\max}}\right)}$
UCB1	N.a.
Delayed-UCB1	$\sum_i C_i \ln T + \left(1 + \frac{\pi^2}{3} + \tau_{\max}\right) \sum_i \Delta_i$
TP-UCB-FR	$\sum_i \frac{C_i}{2\alpha} \ln T \left(1 + \sqrt{1 + \frac{\alpha(\alpha+1)\Delta_i}{2\overline{R}^i \ln T}}\right) + (\alpha+1)\phi \sum_i \overline{R}^i + \left(1 + \frac{\pi^2}{3}\right) \sum_i \Delta_i$
TP-UCB-EW	$\sum_i C_i \ln T + \alpha \left(\phi + \frac{\pi^2}{3}\right) \sum_i \Delta_i$

Table 4.3: *Summary of the known theoretical results (in black) and original contributions provided in this chapter (in blue) for the TP-MAB setting with α-smoothness.*

and TP-MAB settings since it has been developed for a more restrictive scenario, i.e., $\tau_{\max} = 1$.

The results related to the proposed algorithms, i.e., `TP-UCB-FR` and `TP-UCB-EW`, for the MAB setting have been derived fixing $\tau_{\max} = 1$ and $\alpha = 1$ in the corresponding theorems. The results of the Delayed-MAB setting have been derived fixing $\alpha = 1$. We remark that `TP-UCB-FR` in the MAB setting has the same asymptotic order of upper bound of `UCB1`, while the upper bound of `TP-UCB-EW` reduces exactly to the one of `UCB1` in this setting.

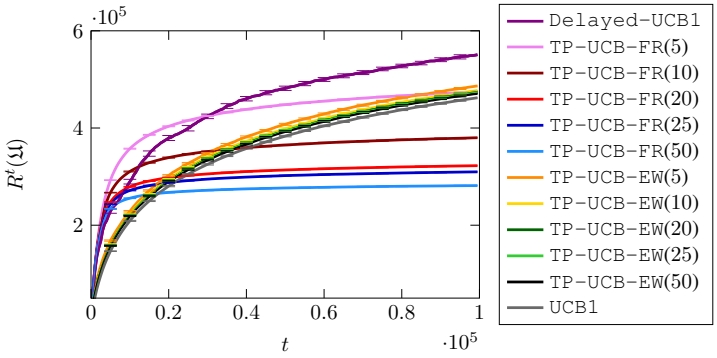


Figure 4.2: *Pseudo-regret over time for Experimental Setting #1.*

4.4 Empirical Evaluation

We compare `TP-UCB-FR` and `TP-UCB-EW` algorithms with the `UCB1` algorithm by Auer et al. [2002a] and the `Delayed-UCB1` algorithm by Joulani et al. [2013] in α-smooth TP-MAB environments. We recall that the `UCB1` algorithm cannot be run in a TP-MAB setting, unless we are in the degenerate case $\tau_{\max} = 1$. Therefore, we assume to immediately get the cumulative reward of a pull. In our setting, `UCB1` represents a *clairvoyant* algorithm observing R_t^i at round t. Section 2.3.1 presents further details on the standard `UCB1` algorithm. Moreover, `Delayed-UCB1` uses the realization of the pulls only when they are complete, i.e., with a constant delay of $\tau_{\max} - 1$. In Appendix A we provide further details on the `Delayed-UCB1` baseline.

In what follows, we compare the algorithms in three settings: two synthetically-generated environments and a real-world playlist recommendation scenario. More details on the description of the experimental settings and some additional experiments are deferred to Appendix A.2.2.

Setting #1. At first, we evaluate the influence of the parameter α. We model $K = 10$ arms, whose maximum reward is s.t. $\overline{R}^i = 100i$. The reward is collected over $\tau_{\max} = 100$ rounds, the smoothness parameter is $\alpha = 20$, and the aggregated rewards are s.t. $Z_{t,k}^i \sim \frac{\overline{R}^i}{\alpha}\mathrm{U}([0,1])$, for each $k \in [\alpha]$. We run the algorithms over a time horizon of $T = 10^5$ and average the results over 50 independent runs. In the results, `TP-UCB-FR`(η) and `TP-UCB-EW`(η) are s.t. the value of α taken as input is η, with $\eta \in \{5, 10, 20, 25, 50\}$.

Results. Figure 4.2 shows the pseudo-regret $R^t(\mathfrak{A})$ over the time horizon and the vertical bars represent the 95% confidence intervals for the

mean value. Let us focus on TP−UCB−FR(20) and TP−UCB−EW(20), for which η is equal to the α of the environment. TP−UCB−EW(20) provides better results than Delayed−UCB1 over the entire time horizon, while TP−UCB−FR(20) is better than Delayed−UCB1 for $t > 10^4$ and better than TP−UCB−EW(20) for $t > 2 \cdot 10^4$. This suggests that TP−UCB−FR(20) is more suitable for longer time horizons, and this behavior is confirmed by the asymptotic order of Theorem 4.3. Notice that UCB1 obtains the reward as soon as an arm has been pulled, but it does not exploit the α-smoothness property. *Vice versa*, our algorithms incorporates this information that, in some specific situations, allows us to beat even the non-delayed baseline.

During rounds $t \in [1, 7000]$, the Delayed−UCB1 algorithm outperforms TP−UCB−FR, since, during the initial rounds, incomplete samples may be far different from the corresponding unseen realizations, and, therefore, TP−UCB−FR initially pulls the suboptimal arms more often than Delayed−UCB1. Nonetheless, over longer time horizons, TP−UCB−FR outperforms Delayed−UCB1, as expected given the result in Theorem 4.3. TP−UCB−EW has a similar asymptotic behavior of those of UCB1 and Delayed−UCB1, *i.e.*, the regret curves becomes parallel after ≈ 4000 rounds. This is because the overall exploration term of the three algorithms is of the same order in t and α, and therefore the advantages of TP−UCB−EW are mainly experienced in the early stages of the learning process. Summarily, for short-time horizons, TP−UCB−EW is preferable to TP−UCB−FR, while TP−UCB−FR shows better performance over long periods.

Let us focus on the results obtained with TP−UCB−FR(η). Setting $\eta < \alpha$, *i.e.*, underestimating the value of α, provides worse results in terms of regret, while $\eta > \alpha$ seems to improve the performance of the algorithm without compromising the convergence properties. This suggests that if the α parameter is unknown, one should use an optimistic (large) value in the algorithm. Notice that the regret varies of $\approx 40\%$ w.r.t. the different versions of TP−UCB−FR changing the value of η, which suggests that TP−UCB−FR is strongly influenced by a mis-specification of the parameter η. Focusing on TP−UCB−EW(η), we have a behaviour similar to the one observed for TP−UCB−FR(η), showing how larger values for η provide better results. Conversely, the performance of TP−UCB−EW present a lower variability by changing the parameter η, and the gap in terms of regret among the different versions of TP−UCB−EW is of $\approx 3\%$.

Setting #2. We study the behavior of our algorithms in settings with different maximum delay $\tau_{\max}$ and smoothness α. The scenario is the same presented in Setting #1 except that the maximum reward for the arm i is

$\tau_{\max}$	α	$R^{T,(\%)}(\mathfrak{U}_{\texttt{FR}})$	$R^{T,(\%)}(\mathfrak{U}_{\texttt{EW}})$
100	10	68.06% (0.26%)	86.03% (0.59%)
200	20	95.42% (0.15%)	80.38% (0.34%)
100	50	50.84% (0.11%)	85.36% (0.33%)
200	100	81.55% (0.10%)	78.70% (0.24%)

Table 4.4: $R^{T,(\%)}(\mathfrak{U})$ *for Experimental Setting #2.*

$\overline{R}^i = \tau_{\max} \cdot i$.[7] We evaluate the algorithms in terms of percentage of the regret w.r.t. the one provided by `Delayed-UCB1`, whose policy is denoted by $\mathfrak{U}_{\texttt{D}}$, formally $R^{T,(\%)}(\mathfrak{U}) := R^T(\mathfrak{U})/R^T(\mathfrak{U}_{\texttt{D}}) \cdot 100$. We average the results over 50 independent experiments.

Results. Table 4.4 provides the values of $R^{T,(\%)}(\mathfrak{U})$ for our algorithms (95% CI in brackets). In all the scenarios, the proposed algorithms outperform the `Delayed-UCB1` algorithm, providing a regret smaller than 95.5% of the `Delayed-UCB1` one. Comparing the results with the same maximum delay $\tau_{\max}$ we notice that a larger value for α provides better performance. This was expected since larger values for α imply that the `TP-UCB-FR` and `TP-UCB-EW` algorithms can better exploit the reward structure. By comparing the settings with maximum delay $\tau_{\max} = 100$ and $\tau_{\max} = 200$, the two algorithms behave in opposite ways: the performance of `TP-UCB-EW` improves by more than 6%, while the regret of `TP-UCB-FR` increases of more than 30%. This is due to the fact that, with larger $\tau_{\max}$, `TP-UCB-FR` shows its better behaviour for larger time horizons.

Spotify Setting. We apply the TP-MAB approach to solve the user recommendation problem presented in Example 4.1, using a dataset by Spotify Brost et al. (2019). We select the $K = 6$ most played playlist as the arms to be recommended, and each time a playlist i is selected, the corresponding reward realizations $\boldsymbol{x}_t^i$ for the first $N = 20$ songs is sampled from the listening sessions of that playlist contained in the dataset. We recall that, in this setting, the maximum delay is $\tau_{\max} = 4N = 80$, and the smoothness parameter is $\alpha = 20$. More details on the setting and the distributions of the reward for each playlist are provided in Appendix A.2.2. We average the results over 50 independent runs.

[7] In Appendix A.2.2, we also report experiments in scenarios differing in how the aggregated rewards are distributed over the ϕ elements composing $Z_{t,k}^i$, which confirm what is shown in this section.

	$R^T(\mathfrak{U})$
Delayed-UCB1	56473 (805)
TP-UCB-FR	25367 (369)
TP-UCB-EW	55000 (951)
UCB1	47368 (1289)

Table 4.5: *Pseudo-regret for the Spotify experimental setting.*

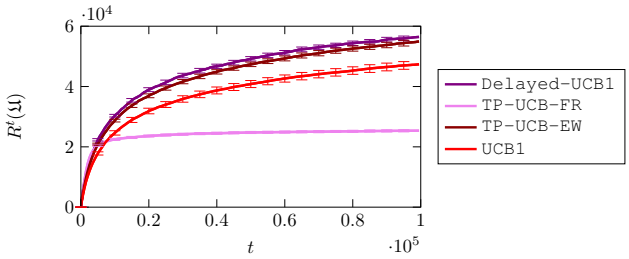


Figure 4.3: *Pseudo-regret over time for the Spotify setting.*

Results. Table 4.5 shows that the TP-UCB-FR algorithm provides the best performance among the analysed algorithms, outperforming UCB1 thanks to the exploitation of the α-smoothness property. The regret over time in Figure 4.3 shows that the TP-UCB-FR provides worse performance than TP-UCB-EW only for a limited amount of rounds ($t < 4000$). This suggests that, in this specific scenario, the TP-UCB-FR algorithm represents a good candidate to provide playlist recommendations.

Part II

Expanding Algorithmic Advertising: Price Displaying, Collusion, Metaverse

CHAPTER 5

Efficiency of Ad Auctions with Price Displaying

We study a novel ad auction framwork recently adopted by platforms, such as *Google Hotel Ads* and *Tripadvisor*, where ads of similar products or services are displayed together with their prices. As in classical ad auctions, the ads are ranked depending on the advertisers' bids, whereas, differently from classical settings, ads are displayed together with their prices, so as to provide a direct comparison among them. This dramatically affects users' behavior, as well as the properties of ad auctions. Section 1.2 provides an extended discussion which introduces the problem studied in this chapter. Section 5.1 presents a formal model af the ad auction setting with price displaying, a direct-revelation mechanism and two indirect-revelation mechanisms, characterized by GSP and VCG payments, respectively. Section 5.3 proves that, in our setting, the problem of allocating advertisers to slots can be solved in polynomial time by both the direct- and the indirect-revelation mechanisms. Section 5.4 analyzes the performance of the indirect-revelation mechanisms in terms *Price of Anarchy* (PoA) and *Price of Stability* (PoS) computed for the social welfare and the revenue. Section 5.5 shows that, under some assumptions and by requiring the advertisers to report an addi-

tional information, it is possible to improve the PoS for the revenue of the indirect-revelation mechanism with VCG payments.

5.1 Model

There is a set $N = \{1, \dots, n\}$ of n agents, who simultaneously play the role of advertisers and sellers. Each agent sells a single good on her own website (*e.g.*, an online marketplace) and relies on an external ad publisher that advertises the good through a single ad in which the price is displayed. Since the goods being sold by the agents are similar, the price comparison that users perform on the publisher's website results in a high competition level among the agents, as happening in classical comparator websites Jung et al. (2014). In the following, for the ease of presentation, we use index $i \in N$ to refer to the agent, her good, and also her ad. Figure 5.1 provides an overview of our scenario.

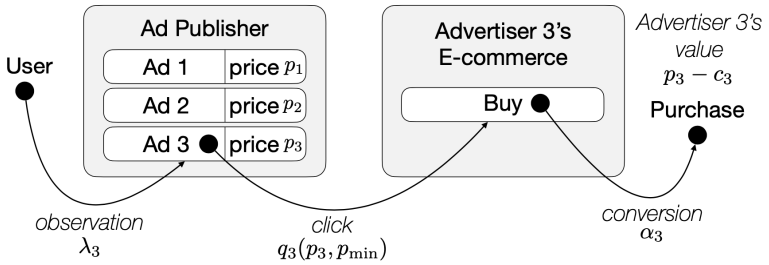


Figure 5.1: *An example of ad auction with price displaying. A user visits a Web page with three ads (ad* 1*, ad* 2*, and ad* 3*) together with their prices (p_1, p_2, and p_3). The user observes slot* 3 *with probability λ_3. Once observed slot* 3*, the user clicks on the ad displayed in slot* 3*,* i.e.*, ad* 3*, with probability $q_3(p_3, p_{\min})$ where $p_{\min}$ is the minimum price among p_1, p_2, p_3. The user visits the Web page of advertiser* 3 *(*e.g.*, an online marketplace), and, then, produces a conversion (*e.g.*, purchase) with probability α_3. The value that advertiser* 3 *gets from the conversion is $p_3 - c_3$.*

For every i, we denote with $c_i \in \mathbb{R}_{\geq 0}$ and $p_i \in \mathbb{R}_{\geq 0}$ the *cost* of supply and the selling *price* of agent i's good, respectively. Furthermore, we denote with $\alpha_i \in [0, 1]$ the probability with which a user buys agent i's good when visiting her website. Thus, agent i's expected *gain* from a visit of a user on her website is $\alpha_i (p_i - c_i)$. Let us remark that the *conversion probability* α_i is constant w.r.t. the price p_i, since we assume that the user is aware of the price before visiting the website and, thus, she does not visit it if the price is larger than her reserve value. As previously discussed, the user first observes the ads on the publisher's website, together with their prices, and,

then, she clicks on an ad so as to visit the corresponding advertiser's website. Therefore, the motivation behind an uncompleted conversion following the user's visit to the advertiser's Web page does *not* concern the price (*e.g.*, it may be due to the user acquiring more information on the seller, or potential extra fees and/or ancillary services). The pair (α_i, c_i) is a private information of agent i, and sometimes we will refer to it as her type θ_i. We let $\Theta = [0,1] \times \mathbb{R}_{\geq 0}$ be the space of types of every agent.

The ad publisher has a set $M = \{1, \ldots, m\}$ of slots in which the ads are displayed. An *assignment* of ads to slots (also called *allocation*) is represented by a function $f \colon N \to M \cup \{\perp\}$ such that there is at most one ad per slot (*i.e.*, there are no ads $i, h \in N$ such that $i \neq h$ and $f(i) = f(h) \in M$). All the ads that are not assigned to slots in M are assigned to $\perp$, meaning that these ads are not displayed. For every slot $j \in M$, we denote with $\lambda_j \in [0,1]$ the probability (called *prominence*) that a user observes the ad displayed in that slot. As customary in the literature, we assume that $\lambda_1 \geq \lambda_2 \geq \ldots \geq \lambda_m$. For the ease of notation, we define $\lambda_\perp = 0$. Furthermore, for every agent i, we denote with $q_i \in [0,1]$ the probability (called *quality*) that a user clicks on ad i conditioned on its observation. In our setting, q_i depends on the prices, as they are displayed with the ads. In particular, q_i is a function of the prices $\mathbf{p} = \{p_i\}_{i \in N}$ of agents whose ads are displayed, since the user can compare all the prices shown on the Web page when deciding the website from which to buy a good. This dependency introduces externalities among the ads. In this work, we assume that $q_i : \mathbb{R}_{\geq 0} \times \mathbb{R}_{\geq 0} \to [0,1]$, where $q_i(p_i, p_{\min})$ denotes the agent i's quality when her price is p_i and the minimum price among all the displayed ads is $p_{\min}$, with $p_{\min} = \min_{h \in N : f(h) \in M} \{p_h\}$ (for the sake of notation, we omit the dependency of $p_{\min}$ on f). Moreover, given $p_{\min}$, q_i is (non-strictly) monotonically decreasing in p_i since, as previously discussed, a user clicks on the ad if the price is non-larger than the user's reserve value. Finally, q_i is (non-strictly) monotonically increasing in $p_{\min}$, given p_i. The rationale behind this assumption is that, given p_i, the probability that a user clicks on ad i decreases as the gap between p_i and the minimum price $p_{\min}$ increases, capturing a potential reduction of the user's interest for agent i's good. A simple example is when the users are only interested in the price and, thus, q_i is zero if $p_i > p_{\min}$. We also assume that there exists $p_i \in \mathbb{R}_{\geq 0}$ maximizing $q_i(p_i, p_i)\, \alpha_i\, (p_i - c_i)$ and, thus, there exists $p_i < \infty$ that agent i would use when displayed alone. Finally, we remark that, as it is customary in the literature, parameters λ and q are estimated by the ad publisher.

Every *mechanism* receives some input (or *bid*) from every agent i, chooses an allocation f, and charges every agent i of a payment π_i. We

say that the mechanism is *direct-revelation* if the input provided by agent i belongs to Θ, *i.e.*, it consists of a conversion probability and a cost, which are *not* necessarily the real ones (her type). Otherwise we say that the mechanism is *indirect-revelation*.

In our setting, a direct-revelation mechanism takes as input a reported type $\theta'_i = (\alpha'_i, c'_i) \in \Theta$ for each agent i, and chooses some prices $\mathbf{p} = \{p_i\}_{i \in N}$ and an allocation function f. We let $\mathbf{b} = \{b_i\}_{i \in N}$ be the vector of declared gains, where $b_i = \alpha'_i (p_i - c'_i)$ is agent i's gain for the reported type θ'_i. On the other hand, an indirect-revelation mechanism takes as input a price p_i and a declared gain b_i for each agent i, and chooses an allocation function f. We say that agent i does not *overbid* if $b_i \leq \alpha_i (p_i - c_i)$, where p_i is the price given as input and $(\alpha_i, c_i) = \theta_i$ is the true agent i's type.

Given an allocation f, prices $\mathbf{p}$, and b_i, we denote with $\widehat{v}_i(f, \mathbf{p}, b_i) = \lambda_{f(i)}\, q_i(p_i, p_{\min})\, b_i$ the expected (w.r.t. clicks and purchase) *value* of agent i according to her declared gain. The true expected value that she receives from allocation f is $v_i = \lambda_{f(i)}\, q_i(p_i, p_{\min})\, \alpha_i (p_i - c_i)$, while agent i's expected *utility* is $u_i = v_i - \pi_i$ since the environment is quasi-linear.[1] The *social welfare* of an allocation with respect to the declared gains is $\widehat{\mathrm{SW}}(f, \mathbf{p}, \mathbf{b}) = \sum_i \widehat{v}_i(f, \mathbf{p}, b_i)$, where $\mathbf{b} = \{b_i\}_{i \in N}$. The true social welfare is $\mathrm{SW} = \sum_i v_i$. The *revenue* is instead $\mathrm{Rev} = \sum_i \pi_i$.

We informally introduce notable properties of mechanisms; see Mas-Colell et al. (1995) for formal definitions. A mechanism, both direct- and indirect-revelation, is *individually rational*, if for every agent i, the assigned payment π_i is non-larger than her value $\hat{v}_i(f, \mathbf{p}, b_i)$ according the declared gain. Furthermore, a mechanism is *weakly budget-balanced* if the sum of payments is always non-negative. A direct-revelation mechanism is *truthful* if for every agent i it is a dominant strategy to report the true type $\theta_i = (\alpha_i, c_i)$ to the mechanism, *i.e.*, the utility that agent i achieves by reporting θ_i is at least as large as with every alternative input, regardless of other agents' actions. For indirect-revelation mechanisms, we say that a set of inputs is in *equilibrium* according to Nash (1951) if no agent may increase her utility by submitting a different bid, whenever the inputs of other agents remain unchanged.

5.2 Mechanisms

Next, we introduce our direct-revelation mechanism and two indirect-revelation mechanisms.

[1] The dependency of v_i, u_i, π_i on the arguments $f, \mathbf{p}, b_i$ is omitted to avoid cumbersome notation.

5.2.1 Direct-revelation Mechanism

We let $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ be the direct-revelation mechanism defined as follows. Given the agent i's input $\theta_i' = (\alpha_i', c_i') \in \Theta$, the mechanism defines $b_i = \alpha_i' (p_i - c_i')$ for every price p_i. Then, the mechanism computes an assignment f^* and prices $\mathbf{p}^*$ that maximize the social welfare with respect to the declared gains; formally,

$$\widehat{\mathrm{SW}}(f^*, \mathbf{p}^*, \mathbf{b}) = \max_{f,\mathbf{p}} \widehat{\mathrm{SW}}(f, \mathbf{p}, \mathbf{b}).$$

Finally, the mechanism assigns to each advertiser i in the allocation (*i.e.*, such that $f(i) \in M$) the VCG payment

$$\begin{aligned}\pi_i &= \max_{f,\mathbf{p}:\ f(i)\notin M} \sum_{j\neq i} \Big(\widehat{v}_j(f, \mathbf{p}, b_j) - \widehat{v}_j(f^*, \mathbf{p}^*, b_j)\Big) \\ &= \widehat{v}_i(f^*, \mathbf{p}^*, b_i) - \Delta_i,\end{aligned}$$

where

$$\Delta_i = \widehat{\mathrm{SW}}(f^*, \mathbf{p}^*, \mathbf{b}) - \max_{f,\mathbf{p}:f(i)\notin M} \widehat{\mathrm{SW}}(f, \mathbf{p}, \mathbf{b}) \geq 0.$$

It is immediate to check that payments cannot be negative and they are never larger than the value corresponding to the declared gain. Thus, the mechanism is trivially individually-rational and weakly budget-balanced. Moreover, it is not hard to verify that these payments allow the mechanism to be truthful (essentially this is a VCG mechanism and there is no inter-dependence among types). Truthfulness also implies that the mechanism maximizes the true social welfare. These observations prove the following theorem.

Theorem 5.1. *Mechanism $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ is truthful, individually rational, weakly budget-balanced, and maximizes* SW.

5.2.2 Indirect-revelation Mechanisms

Next, we introduce two alternative mechanisms, namely $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$. These mechanisms share the same structure, but they differ in the way they compute the payments. They work as follows. Agent i inputs (p_i, b_i), where $p_i \in \mathbb{R}_{\geq 0}$ is the price that agent i wants to be displayed for her ad and $b_i \in \mathbb{R}$ is the expected gain that i declares to achieve from a click on her ad for price p_i. The mechanism computes an assignment g^* that maximizes the social welfare with respect to the submitted prices and gains; formally

$$\widehat{\mathrm{SW}}(g^*, \mathbf{p}, \mathbf{b}) = \max_{g} \widehat{\mathrm{SW}}(g, \mathbf{p}, \mathbf{b}).$$

Then, $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ assigns to each advertiser i such that $g^*(i) \in M$ the VCG payment

$$\begin{aligned}\pi_i &= \max_{g:\, g(i)\notin M} \sum_{j\neq i} \Big(\widehat{v}_j(g,\mathbf{p},b_j) - \widehat{v}_j(g^*,\mathbf{p},b_j)\Big)\\ &= \widehat{v}_i(g^*,\mathbf{p},b_j) - \delta_i,\end{aligned}$$

where

$$\delta_i = \widehat{\mathrm{SW}}(g^*,\mathbf{p},\mathbf{b}) - \max_{g:g(i)\notin M} \widehat{\mathrm{SW}}(g,\mathbf{p},\mathbf{b}) \geq 0.$$

W.l.o.g., let the optimal allocation g^* be such that only the first $\ell \leq m$ slots are assigned and no slot $j > \ell$ is assigned. $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ assigns to each i such that $g^*(i) \in M$ and $g^*(i) < \ell$ (*i.e.*, i is assigned to a slot different from ℓ) the following payments:

$$\varpi_i = \lambda_{g^*(i)} q_j(p_j, p_{\min}) b_j, \tag{5.1}$$

where j is such that $g^*(j) = g^*(i) + 1$. When $g^*(i) = \ell$, there are two possible payments. If all the not assigned agents j (*i.e.*, such that $g^*(j) = \perp$) have a price $p_j < p_{\min}$, then $\varpi_i = 0$. Otherwise, the payment is

$$\varpi_i = \lambda_{g^*(i)} \max_{j:p_j\geq p_{\min}\wedge g^*(j)=\perp} \{q_j(p_j,p_{\min})b_j\}. \tag{5.2}$$

As done for $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$, it is immediate to check that payments are at least zero, and they are always less than the value corresponding to the declared gain. Hence, $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ is individually rational and weakly budget-balanced. Moreover, one may hope that the inputs that agents select at any equilibrium are such that the allocation selected by the mechanism maximize the social welfare. Unfortunately, we will show in the next sections that this is *not* the case.

The payments of $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ are at least zero, and, thus, the mechanism is weakly budged-balanced. Let us also observe that, given agent i, $\forall j$ s.t. $g^*(j) > g^*(i)$ or $g^*(j) = \perp \wedge p_j \geq p_{\min}$, we have that $q_j(p_j,p_{\min})b_j \leq q_i(p_i,p_{\min})b_i$. Otherwise, the allocation g achieved from g^* by fixing $g(j) = g^*(i)$, $g(i) = g^*(j)$, and $g(k) = g^*(k)\ \forall k \notin \{i,j\}$ would achieve a larger social welfare (according to declared gains). Hence, we have that $\varpi_i \leq \widehat{v}_i(g^*,\mathbf{p},b_i)$, and, thus, the mechanism is individually rational. We remark that for this property to hold, it is fundamental that, in Equation 5.2, we consider only the not assigned agents j who have a declared price $p_j \geq p_{\min}$. Indeed, an agent j with $p_j < p_{\min}$ may have a large $q_j(p_j,p_j)b_j$ so that, if the j-th ad is displayed, the minimum price changes from $p_{\min}$ to p_j,

$q_j(p_j, p_j)b_j > q_i(p_i, p_{\min})b_i$, and $\varpi_i > \widehat{v}_i(g^*, \mathbf{p}, b_i)$, where i is the agent assigned to the slot ℓ. Nevertheless, this agent may not be chosen by the allocation g^* because of the negative externalities that its low price would put on other agents (by lowering their value and thus the social welfare). As a result an optimal allocation may not assign all the available slots. We finally observe that, as for $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$, even $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ may fail to optimize the true social welfare. The following sections will bound the extent of this failure.

5.3 Computational Complexity

In general, externalities make hard the problem of computing the allocation maximizing the social welfare. In this section, we prove that in our setting the problem of allocating advertisers to slots can be solved in polynomial time by both the direct- and the indirect-revelation mechanisms.

Let us start with the problem of computing the allocation g^* in the indirect-revelation mechanisms. We show in the next theorem that g^* can be efficiently computed.

Theorem 5.2. *There is an algorithm that computes the allocation g^* in time $O(n^2 \log n)$.*

Proof. Let $\mathbf{b}$ and $\mathbf{p}$ be the set of gains and prices submitted by agents. First observe that, given a minimum displayed price $p_{\min}$, the allocation that maximizes the social welfare (with respect to gains and prices in input), can be trivially computed by sorting agents in $\{i\colon p_i \geq p_{\min}\}$ in order of $q_i(p_i, p_{\min})b_i$ and assigning slot 1 to the agent that maximizes this quantity, slot 2 to the second such agent, and so on. Note that this operation requires $O(n \log n)$ steps.

However, in order to provide the allocation g^*, we also need to decide which is the best value for $p_{\min}$. However, since $p_{\min}$ must belong to $\mathbf{p}$, it is sufficient to compute the best allocation by using as minimum displayed price each of the at most n different prices in $\mathbf{p}$, and choosing the allocation that optimizes the social welfare. □

Computing g^* is an easier problem than the one faced by the direct-revelation mechanism, since, for the former, prices are given and we optimize only over the allocation function, while, for the latter, optimization occurs both on the allocation function and prices. Nevertheless, the following theorem shows that f^* and $\mathbf{p}^*$ can also be computed efficiently, as long as the set P of allowed prices is discrete and finite.

Theorem 5.3. *There is an algorithm that computes the allocation f^* and prices $\mathbf{p}^*$ in time $O(n^2|P|(|P| + \log n))$.*

Proof. Let $b_i(p) = \alpha'_i(p - c'_i)$ be the expected gain of agent i according to her input when ad i is displayed with price p, where (α'_i, c'_i) is the input of agent i. For each agent i and every price $\hat{p} \in P$ we compute $p_i^*(\hat{p})$ as follows: if $\max_{p \in P: p \geq \hat{p}} q_i(p, \hat{p}) b_i(p) > 0$, then

$$p_i^*(\hat{p}) = \arg \max_{p \in P: p \geq \hat{p}} q_i(p, \hat{p}) b_i(p),$$

otherwise we set $p_i^*(\hat{p}) = \perp$. Roughly speaking, $p_i^*(\hat{p})$ is the best price (according to her input) for agent i when the minimum displayed price is $\hat{p}$ and the i-th ad is displayed (and thus i's price is at least $\hat{p}$). Clearly, if there is no price larger than or equal to $\hat{p}$ guaranteeing to agent i a positive utility, then she prefers to be not displayed. For this reason, in the latter case, we do not assign any value to $p_i^*(\hat{p})$. Notice that $p_i^*(\hat{p})$ can be computed by evaluating the function for every $p \in P$ with $p \geq \hat{p}$, requiring at most $O(|P|)$ operations.

Then, if the minimum displayed price $p_{\min}$ was given, along with the agent to which it is assigned, then we simply choose price $p_i^*(p_{\min})$ for each remaining agent i (this can be done in $O(nP)$ steps), prune out agents for which $p_i^*(p_{\min}) = \perp$, and finally compute the corresponding optimal assignment by sorting the remaining agents in order of $b_i(p_i^*(p_{\min}))$, as shown in Theorem 5.2 (in $O(n \log n)$ steps).

Unfortunately, selecting $p_{\min}$ is much harder than in the indirect case: not only the value of $p_{\min}$ can assume every value in P (and not just one among at most n alternatives), but we also need to decide which agent should display this price. For this reason, we need to go through every price $p \in P$ and every agent i and compute the best solution that would be achieved when i is the agent displaying the minimum price p. Since for each of the nP possible choices, computing the best solution requires time $O(nP + n \log n)$, we achieve the desired running time. □

Observe that the dependence on $|P|$ in Theorem 5.3 is in some way necessary as long as we would like to keep quality function as general as possible. It is not hard to see that we can avoid to check all prices by doing opportune restriction on the quality functions.

We finally highlight that the discretization of the set of prices does not affect the property of the mechanism. In particular, truthfulness continues to hold, since the mechanism is maximal-in-the-range.

5.4 Performance of the Indirect Mechanisms

For the sake of presentation, we provide the informal definitions of PoS and PoA for social welfare and revenue; formal definitions can be found in Nisan et al. (2007).

- PoS for the social welfare is the minimum—w.r.t. all the Nash equilibria—ratio between the maximum achievable social welfare and the social welfare of an allocation achievable in a Nash equilibrium of an indirect-revelation mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ or $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$.
- PoA for the social welfare is the maximum—w.r.t. all the Nash equilibria—ratio between the maximum achievable social welfare and the social welfare of an allocation achievable in a Nash equilibrium of an indirect-revelation mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ or $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$.
- PoS for the revenue is the minimum—w.r.t. all the Nash equilibria—ratio between the maximum revenue achievable by an individually-rational mechanism and the revenue achievable in a Nash equilibrium of an indirect-revelation mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ or $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$.
- PoA for the revenue is the maximum—w.r.t. all the Nash equilibria—ratio between the maximum revenue achievable by an individually-rational mechanism and the revenue achievable in a Nash equilibrium of an indirect-revelation mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ or $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$.

Table 5.1 summarizes the lower and upper bounds over the mechanisms' inefficiency when agents do not overbid; the results when agents overbid are omitted since the inefficiency can be arbitrary even with a single slot. Interestingly, while $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ performs as well as $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$with a single slot as $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ are equivalent in this case since there is no externality;

	1 slot			$m \geq 2$ slots		
	SW		Rev	SW		Rev
	PoS	PoA	PoS	PoS	PoA	PoS
$\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$	1	1	1 (♠)	1	m	∞
$\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$	1	1	∞	≥ 2	$\geq m$	∞

Table 5.1: *Lower and upper bounds over PoS and PoA when agents do not overbid. ♠: PoS here is taken w.r.t. the mechanism $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ maximizing the social welfare (thus not necessarily maximizing the revenue).*

with more than 2 slots the inefficiency can be large both for social welfare and revenue even in the basic case in which slots are indistinguishable and $\lambda = 1$. In particular, in our proofs of the upper-bound results, we use a special class of quality functions that we denote as *only-min* functions, which assign a value 0 to the quality of an agent when her price is not the minimum among those displayed, and we prove that in many cases no worse instance is possible. With multiple slots, the positive result is that, with $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$, the optimal allocation is always achievable by some Nash equilibrium (*i.e.*, PoS $= 1$). Nevertheless, there are auction instances in which some Nash equilibria lead to allocations whose social welfare is $1/m$ of the optimal allocation (*i.e.*, PoA $= m$) or in which all the Nash equilibria lead to a revenue of zero whereas the direct-revelation mechanism $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ provides a strictly positive revenue (*i.e.*, PoS $= \infty$). $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ performs even worse than $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$, both with a single and multiple slots.

In the following, we formally provide the results on the lower and upper bounds over the mechanisms' inefficiency.

5.4.1 Price of Stability for the Social Welfare

Initially, we provide our main positive result in terms of indirect-revelation mechanisms inefficiency.

Theorem 5.4. *The PoS for the social welfare of* $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ *is* 1.

Proof. Suppose that each agent i reports the pair $(\tilde{p}_i, \tilde{b}_i)$ defined as follows: if the mechanism $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ displays the ad i when run on truthful bids, then $\tilde{p}_i$ is the corresponding price, and $\tilde{b}_i = \alpha_i(\tilde{p}_i - c_i)$, *i.e.*, the true gain associated to this price; otherwise $\tilde{p}_i = \tilde{b}_i = 0$. It is immediate to check that with these bids the allocation returned by $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ is exactly the same as the one returned by $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$, and, thus, it maximizes social welfare.

Unfortunately, we cannot conclude that inputs $(\tilde{p}_i, \tilde{b}_i)$ are in equilibrium directly from the truthfulness of $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$. Indeed, the payments assigned by the indirect mechanism are different from the ones assigned by the direct mechanism. Moreover, in the former the agent may lie both about the price and about the expected gain, while in the latter an agent may essentially lie only on the expected gain. Still, in the following we prove that inputs $(\tilde{p}_i, \tilde{b}_i)$ are in equilibrium, and, thus, the theorem follows.

In particular, let $\tilde{\mathbf{p}} = (\tilde{p}_1, \ldots, \tilde{p}_n)$ and $\tilde{\mathbf{b}} = (\tilde{b}_1, \ldots, \tilde{b}_n)$. We prove that the utility $\tilde{u}_i$ of agent i when the mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ is run on $\tilde{\mathbf{p}}$ and $\tilde{\mathbf{b}}$ is at least the utility u_i that she achieves if the mechanism would be run on input $\mathbf{p} = (p_i, \tilde{\mathbf{p}}_{-i})$ and $\mathbf{b} = (b_i, \tilde{\mathbf{b}}_{-i})$, for every i, p_i, and b_i. Indeed if i is

allocated by the mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ when run on input $\tilde{\mathbf{p}}$ and $\tilde{\mathbf{b}}$, then, since, by definition of $\tilde{b}_i$, $v_i = \widehat{v}_i(f^*, \tilde{\mathbf{p}}, \tilde{b}_i)$,

$$\begin{aligned}\tilde{u}_i &= v_i - \pi_i = \widehat{v}_i(f^*, \tilde{\mathbf{p}}, \tilde{b}_i) - \pi_i \\ &= \widehat{SW}(f^*, \tilde{\mathbf{p}}, \tilde{\mathbf{b}}) - \max_{g: g(i) \notin M} \widehat{SW}(g, \tilde{\mathbf{p}}, \tilde{\mathbf{b}}) \geq 0,\end{aligned}$$

where f^* is the allocation returned by $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ on truthful bids. If i is instead, unallocated then

$$\tilde{u}_i = 0 = \widehat{SW}(f^*, \tilde{\mathbf{p}}, \tilde{\mathbf{b}}) - \max_{g: g(i) \notin M} \widehat{SW}(g, \tilde{\mathbf{p}}, \tilde{\mathbf{b}}).$$

Thus, if the agent i is unallocated by the mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ when run on input $\mathbf{p}$ and $\mathbf{b}$, then the equilibrium condition is trivially satisfied. Otherwise, let $\check{b}_i = \alpha_i(p_i - c_i)$ and $\check{\mathbf{b}} = (\check{b}_i, \mathbf{b}_{-i})$. We have:

$$\begin{aligned}u_i = v_i - \pi_i &= \widehat{v}_i(g^*, \mathbf{p}, \check{b}_i) - \widehat{v}_i(g^*, \mathbf{p}, b_i) \\ &\quad + \widehat{SW}(g^*, \mathbf{p}, \mathbf{b}) - \max_{g: g(i) \notin M} \widehat{SW}(g, \mathbf{p}, \mathbf{b}) \\ &= \widehat{SW}(g^*, \mathbf{p}, \check{\mathbf{b}}) - \max_{g: g(i) \notin M} \widehat{SW}(g, \tilde{\mathbf{p}}, \tilde{\mathbf{b}}),\end{aligned}$$

where the last equality follows since $p_j = \tilde{p}_j$ and $b_j = \tilde{b}_j$ for every agent $j \neq i$.

Since $\widehat{SW}(f^*, \tilde{\mathbf{p}}, \tilde{\mathbf{b}}) \geq \widehat{SW}(g^*, \mathbf{p}, \check{\mathbf{b}})$, because f^* and $\tilde{\mathbf{p}}$ are the allocation and the prices that maximize the social welfare, we have that $\tilde{u}_i \geq u_i$, as desired. □

The proof of the theorem above shows that, with VCG payments, there is always a Nash equilibrium in which every agent i bids the truthful gain b_i and the price that $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ would use. Such a strategy profile leads to the same allocation of $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$, thus guaranteeing a PoS for the social welfare of 1, but, as we discuss in the following sections, the revenue of the two mechanisms can be different. The same result does not hold in the case of GSP payments, thus leading to a larger PoS for the social welfare.

Theorem 5.5. *The PoS for the social welfare of $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ is at least 2 even if agents do not overbid.*

Proof. We will next show a setting for which it occurs that, with GSP payments, all equilibrium bids make the mechanism to allocate agents with very low prices, implying a corresponding low social welfare, where the optimal allocation only allocates agents with high prices.

SETTING. For $\varepsilon > 0$, consider the following setting:

- $n = 3$, $m = 2$, $P = [\underline{p}, \overline{p}]$ with $\underline{p} > \varepsilon$ and $\overline{p} = \frac{3}{2}(\underline{p} - \varepsilon)$;
- for every agent i we have

$$q_i(p_i, p_{\min}) = \begin{cases} 1 & \text{if } p_i = p_{\min}, \\ 0 & \text{otherwise;} \end{cases}$$

- $c_1 = 0$, $c_2 = c_3 = \underline{p} - \varepsilon$;
- for every agent $i \in N$, $\alpha_i = 1$;
- for every slot $j \in M$, $\lambda_j = 1$;
- ties are broken in favour of agent 1.

Observe that for every $\underline{p} \leq p_1 < p_2 \leq \overline{p}$, it holds that

$$p_1 - c_1 \geq \underline{p} - c_1 = \underline{p} > \underline{p} - \varepsilon \geq 2(p_2 - c_2), \tag{5.3}$$

i.e. the social welfare achieved by displaying agents 1 at price p is larger than what we achieve by displaying agents 2 and 3 at price p_2. In other words, the mechanism always chooses the price submitted by agent 1 as minimum price.

BEST SOCIAL WELFARE WITH $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$. Let (p_1, p_2, p_3) and (b_1, b_2, b_3) be the price and gains given in input to the mechanism. Suppose that they form a Nash equilibrium. We will next provide a characterization of these values.

First observe that, in equilibrium, it must be the case that b_1 is large enough to allow ad 1 to be displayed. Indeed, if this is not the case, then agent 1 would have an incentive to submit the true gain, and thus, by (5.3) and the no overbidding assumption, to be displayed and to achieve a strictly positive utility.

Suppose first that $p_1 = p > \underline{p}$. We next show that in this case there is at least one agent $i \in \{2, 3\}$ such that $p_i = p$ and $b_i = p - c_i$ (a larger declared gain would not be possible because of the no overbidding assumption). Suppose indeed that this is not the case. If both agents have $p_i \neq p$, then they must have 0 utility (if $p_i > p$, their value must be 0 because of the quality function, and if $p_i < p$ they are not displayed since otherwise they will zeroth the value of agent 1). However, if one of these agents submits price p and the corresponding true gain, she would be displayed and achieve strictly positive utility, regardless of b_1 (if $b_1 \geq p - c_2$, the payment assigned to the deviating agent is 0, and if $b_i < p - c_2$, then the payment will be less than the value for being displayed at that price.) Suppose then that there

is agent $i \in \{2,3\}$ with $p_i = p$ but $b_i < p - c_i$ and agent $j = 5 - i$ with either $p_j \neq p$ or $b_j < p - c_j$. Note that the ad of one of these agents, say, wlog, j, is not displayed. Then j has an incentive to submit price p and gain $p - c_j$, since it would assure that her ad will displayed and she receives strictly positive utility.

Next we prove that $b_1 \geq p - c_2$, and thus agent 1 is assigned the first slot (because of the tie-breaking rule). Suppose instead that $b_1 < p - c_2$, and let $i \in \{2,3\}$ be the one agent with $p_i = p$ and $b_i = p - c_i$. Since, as observed above, b_1 must be large enough to have that ad 1 is displayed, this means that either $p_j \neq p$ or $b_j \leq b_1$, with $j = 5 - i$. However, as showed above, j has an incentive to deviate by submitting price p and gain $b_j \in (b_i, b_1)$.

Hence, if $p_1 = p > \underline{p}$, then agent 1 will be displayed in the first slot and will be assigned a payment $p - c_2$. Hence, her utility is $c_2 = \underline{p} - \varepsilon$. We will next show that agent 1 has then an incentive to deviate from this equilibrium. Specifically, let $i \in \{2,3\}$ be the agent submitting price p and gain $p - c_i$. We distinguish two cases based on p_j and b_j, where $j = 5 - i$: if $p_j > \underline{p}$ or $b_j < \underline{p} - c_2 = \varepsilon$, then agent 1 has an incentive to submit price $\underline{p}$ and the corresponding true gain, being allocated in the first slot and being assigned a payment of at most b_j, resulting in an utility $\underline{p} - b_j > \underline{p} - \varepsilon$; if $p_j = \underline{p}$ and $b_j = \varepsilon$ (b_j cannot be larger because of the no overbidding assumption), then agent 1 has an incentive to submit price $\underline{p}$ and gain $b_1 < \varepsilon$, being allocated in the second slot and receiving a null payment, resulting in utility $\underline{p} > \underline{p} - \varepsilon$.

We can then conclude that in an equilibrium $p_1 = \underline{p}$ and ad 1 must be displayed, that implies that every equilibrium cannot have social welfare larger than $(\underline{p} - c_1) + (\underline{p} - c_2) = \underline{p} + \varepsilon$.

SOCIAL WELFARE OF THE OPTIMAL ALLOCATION. The optimal allocation will display agents 1 and 2 at price $\overline{p}$. Hence, the optimal social welfare is $(\overline{p} - c_1) + (\overline{p} - c_2) = \frac{3}{2}(\underline{p} - \varepsilon) + \frac{3}{2}(\underline{p} - \varepsilon) - \underline{p} + \varepsilon = 2(\underline{p} - \varepsilon)$. Hence, the price of Stability is $\frac{2(\underline{p}-\varepsilon)}{\underline{p}+\varepsilon}$ that goes to 2 as ε goes to 0. □

5.4.2 Price of Anarchy for the Social Welfare

We initially focus on the basic case with a single slot, showing that in this case $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ are efficient.

Theorem 5.6. *The PoA for the social welfare of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ is 1 if $m = 1$ when agents do not overbid.*

Proof. When a single slot is available, the value of displayed agent i is $\lambda_1 q_i(p_i, p_i)\alpha_i(p_i - c_i)$, where p_i is the corresponding displayed price. That is, this value does not depend on the prices submitted by other agents.

Let $\tilde{v}_i = \max_p \lambda_1 q_i(p,p)\alpha_i(p-c_i)$ and set $\tilde{p}_i$ to any price p such that $\lambda_1 q_i(p,p)\alpha_i(p-c_i) = \tilde{v}\}$. Finally, sort agents in order of $\tilde{v}_i$, so that $\tilde{v}_1 \geq \tilde{v}_2 \geq \cdots \geq \tilde{v}_n$. Note that if these values are all equals, then, regardless of the displayed agent, the mechanism always maximizes the social welfare. Suppose instead that there are at least two different values. Let k be the first index such that $\tilde{v}_k > \tilde{v}_{k+1}$. Then we claim that in any equilibrium one agent $i \leq k$ must be displayed, otherwise she has the incentive to submit price $\tilde{p}_i$ and the corresponding true gain. This indeed causes the mechanism to display ad i, that provides this agent with a value $\tilde{v}_i$, and to assign a payment (both in case of VCG and GSP payments) that is at most $\tilde{v}_{k+1}$ (because of the non-overbidding assumption), resulting in this way in a positive utility. □

Then, we study the case with multiple slots providing a lower bound on PoA.

Theorem 5.7. *The PoA for the social welfare of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ is at least m if $m \geq 2$ when agents do not overbid.*

Proof. The proof is based on the following setting, in which the ratio between the social welfare of the optimal allocation and the social welfare of the allocation achievable in the worst Nash equilibria in $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ is exactly m.

SETTING. Consider the following setting:

- $n = m + 1$;
- for every agent $i \in N$,

$$q_i(p_i, p_{\min}) = \begin{cases} 1 & \text{if } p_i \leq \overline{p} \text{ and } p_i = p_{\min} \\ 0 & \text{otherwise} \end{cases};$$

- for every agent $i \in N$, $c_i = 0$;
- for every agent $i \in N$, $\alpha_i = 1$;
- for every slot $j \in M$, $\lambda_j = 1$.

SOCIAL WELFARE OF THE OPTIMAL ALLOCATION. One of the allocations maximizing the social welfare is such that $f(i) = i$ for every $i \in N, i \neq m+1$ and $f(m+1) = \perp$, while $p_i = \overline{p}$ for every $i \in N$. The optimal social welfare SW is $m\,\overline{p}$.

SOCIAL WELFARE WITH $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$. Define $\underline{p} = \overline{p}/m$. Consider the case in which, for every $i \in N$, it holds $p_i = \underline{p}$ and $b_i = \underline{p}$ and therefore every agent is declaring her true gain. For every $i \in N$, we have that $q_i = 1$, $p_i = \underline{p}$, and

$u_i = 0$, as payment π_i equals the expected value v_i. This strategy profile leads to a social welfare $\mathrm{SW} = m\,\underline{p} = m\,\overline{p}/m = \overline{p}$. In the following, we show that such a strategy profile is a Nash equilibrium of the full-information game in which the payments are VCG-like, thus proving the theorem.

Initially, we analyze possible deviations to values of b_i different from $\underline{p}$ when keeping $p_i = \underline{p}$, showing that no deviation allows agents to strictly increase their utility. Since $b_i = \underline{p}$ is the true gain of agent i from being allocated and agents are assumed not to overbid, no agent i declares a gain larger than $\underline{p}$. Furthermore, declaring a gain strictly smaller than $\underline{p}$ is a weakly dominated strategy. Indeed, agent i with $f(i) \in M$ would be not displayed by declaring a gain smaller than $\underline{p}$, while agent i with $f(i) = \perp$ keeps not be displayed when declaring a gain less than $\underline{p}$.

Now, we analyze possible deviations to values of p_i different from $\underline{p}$. We consider the restricted case in which, in the deviation, agent i changes both b_i and p_i such that $b_i = p_i$, discussing below that no deviation with $b_i < p_i$ is useful for agent i. Notice that, by setting $b_i = p_i$, an agent is declaring exactly her gain, and, therefore, any strictly larger declared gain would correspond to overbidding. For every $p_i > \underline{p}$, we have that $q_i = 0$ if ad i is displayed together other ads, as the other ads have a price strictly smaller than p_i. Thus, either ad i is displayed alone in the allocation, to guarantee that p_i is the minimum price of this new allocation and therefore that $q_i > 0$, or ad i is not displayed as we can allocate m ads each with a strictly positive value. Under the assumption that ties are opportunely broken, ad i is displayed alone only if her gain (*i.e.*, p_i) is strictly larger than the cumulative gain of the other ads (*i.e.*, $m\,\underline{p}$). This never happens as, by construction, $p_i \leq m\,\underline{p} = \overline{p}$, otherwise (*i.e.*, for $p_i > \overline{p}$) the value of q_i would be 0, and therefore $b_i \leq m\,\underline{p}$, not allowing ad i to be displayed. Notice that the same happens when, in the deviation, agent i underbids making $b_i < p_i$. Finally, by arguments similar to those used above, if $p_i < \underline{p}$, ad i is either displayed alone or not displayed. As above, when $b_i \leq p_i < \underline{p}$, agent i cannot be displayed as her gain cannot be larger than the cumulative gain of the other agents.

SOCIAL WELFARE WITH $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$. Consider the case in which, for every $i \in N$, $p_i = \underline{p}$ and $b_i = \underline{p}$. For the same arguments used above for $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$, such a strategy profile is a Nash equilibrium, thus leading to a social welfare that is $1/m$ of the optimal social welfare. This concludes the proof. □

In the specific case of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$, we show that a PoA larger than m is not possible, and therefore there are no instances worse than those used in the proof of Theorem 5.7. Most interestingly, this result holds even when q_i is

not monotonically decreasing in p_i.

Theorem 5.8. *The PoA for the social welfare of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ is at most m if $m \geq 2$ when agents do not overbid.*

Proof. For the sake of presentation, we introduce the following notation:

- we denote the maximum value agent i can get with $v_i^* = \lambda_1 \max_{p_i}\{q_i(p_i, p_i)\,\alpha_i\,(p_i - c_i)\}$;
- we denote the corresponding optimal price with p_i^*;
- we denote the corresponding true gain with $b_i = \alpha_i\,(p_i^* - c_i)$;
- we denote the allocation in a Nash equilibrium with f^{NE};
- we denote the declared gain used by agent i in the Nash equilibrium with b_i^{NE};
- we denote the price used by agent i in the Nash equilibrium with p_i^{NE};
- we denote the minimum price among those of the displayed ads in the Nash equilibrium with $p_{\min}^{\mathrm{NE}}$;
- we denote the value agent i gets in the Nash equilibrium with $v_i^{\mathrm{NE}} = \lambda_{f^{\mathrm{NE}}(i)}\,q_i(p_i^{\mathrm{NE}}, p_{\min}^{\mathrm{NE}})\,\alpha_i\,(p_i^{\mathrm{NE}} - c_i)$;
- we denote the payment of agent i in the Nash equilibrium with π_i^{NE};
- we denote the social welfare in a Nash equilibrium with $\mathrm{SW}^{\mathrm{NE}} = \sum_{i \in N} v_i^{\mathrm{NE}}$;
- we denote the optimal social welfare when agent i is discarded evaluated by $\{b_i^{\mathrm{NE}}\}_{i \in N}$ with $\widehat{\mathrm{SW}}_{-i}$.

Initially, we prove that, for every Nash equilibrium and agent $i \in N$, it holds $v_i^* \leq \mathrm{SW}^{\mathrm{NE}}$. According to the definition of the VCG payments, the utility of agent i in a Nash equilibrium can be written as:

$$\underbrace{\lambda_{f^{\mathrm{NE}}(i)}\,q_i(p_i^{\mathrm{NE}}, p_{\min}^{\mathrm{NE}})\,\alpha_i\,(p_i^{\mathrm{NE}} - c_i)}_{v_i^{\mathrm{NE}}} - \underbrace{\widehat{\mathrm{SW}}_{-i} + \sum_{h \neq i} \lambda_{f^{\mathrm{NE}}(h)}\,q_h(p_h^{\mathrm{NE}}, p_{\min}^{\mathrm{NE}})\,b_h^{\mathrm{NE}}}_{\pi_i^{\mathrm{NE}}} =$$

$$\left(\lambda_{f^{\text{NE}}(i)}\, q_i(p_i^{\text{NE}}, p_{\min}^{\text{NE}})\, \alpha_i\, (p_i^{\text{NE}} - c_i) + \right.$$
$$\left. \sum_{h \neq i} \lambda_{f^{\text{NE}}(h)}\, q_h(p_h^{\text{NE}}, p_{\min}^{\text{NE}})\, b_h^{\text{NE}} \right) - \left(\widehat{\text{SW}}_{-i} \right).$$

We call $\widehat{\text{SW}}_{-i}$ as negative-utility term and the remaining part as positive-utility term. Since agents are not allowed to overbid, we have that the positive-utility term is $\leq \text{SW}^{\text{NE}}$. Notice that negative-utility term $\widehat{\text{SW}}_{-i}$ is a constant for every deviation of agent i from $(b_i^{\text{NE}}, p_i^{\text{NE}})$, not depending on (b_i, p_i). This means that agents aim at maximizing the positive-utility term. Therefore, there is no pair (b_i, p_i) that agent i can play leading to a different allocation f and/or minimum price $p_{\min}$ providing a positive-utility term strictly larger than $\lambda_{f^{\text{NE}}(i)}\, q_i(p_i^{\text{NE}}, p_{\min}^{\text{NE}})\, \alpha_i\, (p_i^{\text{NE}} - c_i) + \sum_{h \neq i} \lambda_{f^{\text{NE}}(h)}\, q_h(p_h^{\text{NE}}, p_{\min}^{\text{NE}})\, b_h^{\text{NE}}$, otherwise agents would not play a Nash equilibrium. Among all the possible deviations of agent i, we have that the deviation towards (b_i^*, p_i^*) would provide a positive-utility term $\geq v_i^*$. Indeed, it is always possible to allocate ad i in slot 1 and not to display the other ads, thus obtaining exactly v_i^*, or allocating other ads in addition to ad i, thus obtaining strictly more than v_i^*. However, the positive-utility term provided when agent i deviates towards (b_i^*, p_i^*) is not larger than the positive-utility term in the Nash equilibrium, otherwise we would not be in a Nash equilibrium. Therefore, we have:

$$v_i^* \leq \lambda_{f^{\text{NE}}(i)}\, q_i(p_i^{\text{NE}}, p_{\min}^{\text{NE}})\, \alpha_i\, (p_i^{\text{NE}} - c_i) + \sum_{h \neq i} \lambda_{f^{\text{NE}}(h)}\, q_h(p_h^{\text{NE}}, p_{\min}^{\text{NE}})\, b_h^{\text{NE}} \leq \text{SW}^{\text{NE}}.$$

Now, we show that the value of the optimal allocation, say OPT, is smaller than or equal to $m\, \text{SW}^{\text{NE}}$. Indeed, we have:

$$\text{OPT} \leq \sum_{i \in \text{top}(m)} v_i^* \leq m\, \text{SW}^{\text{NE}},$$

where $\text{top}(m)$ are the ads with the top m values v_i^*. The first inequality holds as OPT cannot be larger than the sum of the top m values v_i^*. The second inequality follows from what showed above. Thus, it follows $\text{OPT} \leq m\, \text{SW}^{\text{NE}}$. □

Finally, we show that when agents overbid, the inefficiency can be arbitrarily large.

Theorem 5.9. *The PoA for the social welfare of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ is ∞ even if $m = 1$ when agents can overbid.*

Proof. The proof is based on the following setting.

SETTING. Consider the following setting:

- $n = 2$;
- $m = 1$;
- $q_1(p_1, p_{\min}) = \begin{cases} 1 & \text{if } p_1 \leq \overline{p} \text{ and } p_1 = p_{\min}; \\ 0 & \text{otherwise;} \end{cases}$
- $q_2(p_2, p_{\min}) = \begin{cases} \delta & \text{if } p_2 \leq \overline{p} \text{ and } p_2 = p_{\min}; \\ 0 & \text{otherwise;} \end{cases}$
  where $0 < \delta < 1$;
- $c_1 = c_2 = 0$;
- $\alpha_1 = \alpha_2 = 1$;
- $\lambda_1 = 1$.

SOCIAL WELFARE OF THE OPTIMAL ALLOCATION. The optimal allocation is $f(1) = 1, f(2) = \perp$ with $p_1 = p_2 = \overline{p}$.

SOCIAL WELFARE WITH $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ AND $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$. When overbidding is allowed, the following strategy profile is a Nash equilibrium $(b_1 = 0, p_1 = \overline{p}, b_2 = 2\,\overline{p}/\delta, p_2 = \overline{p})$. Let us notice that indirect-revelation VCG and GSP mechanisms are the same mechanism in this case. Indeed, agent 2 gets a utility of δ, as her payment is $\pi_2 = 0$, and there is no other strategy providing agent 2 a strictly larger utility. Agent 1 can get her ad allocated, but, in doing that, she would be charged of a payment $\pi_1 = 2\,\overline{p}$ strictly larger than her value. Thus, agent 1 will not do it. As a result, strategy profile $(b_1 = 0, p_1 = \overline{p}, b_2 = 2\,\overline{p}/\delta, p_2 = \overline{p})$ is a Nash equilibrium. Thus, the Price of Anarchy is $1/\delta$, that is unbounded from above as $\delta \to 0^+$. □

5.4.3 Price of Stability for the Revenue

Initially, we provide our main result, showing that $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ can be arbitrarily inefficient even with 2 slots.

Theorem 5.10. *The PoS for the revenue of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ is ∞ even if $m = 2$.*

Proof. The proof is based on the following setting, in which $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ provides a strictly positive revenue, while both $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ provide a revenue of zero in every Nash equilibrium.

SETTING. Consider the following setting:

- $n = m = 2$;
- $q_1(p_1, p_{\min}) = \begin{cases} 1 & \text{if } p_1 \leq \overline{p} \text{ and } p_1 = p_{\min} \\ 0 & \text{otherwise} \end{cases}$;
- $q_2(p_2, p_{\min}) = \begin{cases} 1 & \text{if } p_2 < \underline{p} \text{ and } p_2 = p_{\min} \\ \psi(p_2) & \text{if } \underline{p} \leq p_2 \leq \overline{p} \text{ and } p_2 = p_{\min} \\ 0 & \text{otherwise} \end{cases}$ where $1 \leq \underline{p} < \overline{p}/2$;
- $c_1 = c_2 = 0$;
- $\alpha_1 = \alpha_2 = 1$;
- $\lambda_1 = \lambda_2 = 1$.

Furthermore, function $\psi(p_2)$ is defined as follows:

$$\psi(p_2) = \begin{cases} 1 & \text{if } p_2 = \underline{p}; \\ \dfrac{(1+\delta)\,\left(\overline{p} - \underline{p}\right)}{\left(1 - \frac{1+\delta}{2}\right)\,p_2 + \frac{1+\delta}{2}\,\overline{p} - \underline{p}} - 1 & \text{if } \underline{p} < p_2 < \overline{p}; \\ \delta & \text{if } p_2 = \overline{p}; \end{cases}$$

where $0 < \delta < \underline{p}/\overline{p} < 1/2$. In particular, function $\psi(p_2)$ is an hyperbola such that $\psi(\underline{p}) = 1$ and $\psi(\overline{p}) = \delta$ and is continuous and monotonically decreasing in $[\underline{p}, \overline{p}]$.

REVENUE WITH $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$. We notice that, by definition of q_1, q_2, if $p_i = p_{\min} < p_{-i}$, then $q_{-i} = 0$. [2] Therefore, when $p_i < p_{-i}$, no more than one ad with strictly positive value can be allocated. That is, in this case, the allocation is composed of a single ad. In particular, the ad j with the maximum $q_j(p_j, p_j)\,p_j$ is the only ad to be allocated. However, it can be trivially observed that, when $p_i < p_{-i} \leq \overline{p}$, the mechanism can always increase the value of the allocation by changing the price of the ad that it would not allocate. In particular, the mechanism can set that price equal to the price of the ad that it would allocate. In this way, both ads will

[2] We denote with $-i$ the advertiser $j \neq i$.

be displayed, strictly increasing the value of the allocation. Thus, setting $p_1 \neq p_2$ is never optimal.

We restrict our attention to prices $\underline{p}, \overline{p}$ for both advertisers, and we search for the best allocation and prices. We show below that the agents cannot increase their utility by using prices different from $\underline{p}, \overline{p}$. The possible combinations of prices are the following:

- $(p_1 = \underline{p},\, p_2 = \underline{p})$: one of best allocations is $f(1) = 1$, $f(2) = 2$, and the social welfare of the allocation is $2\,\underline{p}$;
- $(p_1 = \overline{p},\, p_2 = \overline{p})$: the best allocation is $f(1) = 1$, $f(2) = 2$, and the social welfare of the allocation is $(1+\delta)\,\overline{p}$.

Thus, the best allocation is for $(p_1 = \overline{p},\, p_2 = \overline{p})$ as, by construction, $2\,\underline{p} < \overline{p} < (1+\delta)\,\overline{p}$.

Now, we show that no price different than $\underline{p}, \overline{p}$ can lead to a better social welfare. Notice that any price $< \underline{p}$ leads to a value strictly smaller than that provided by $\underline{p}$ as $q_1 = q_2 = 1$, and for any price $> \overline{p}$ we have $q_1, q_2 = 0$ and therefore the social welfare is zero. Thus, we can safely restrict our attention to the set $(\underline{p}, \overline{p})$. By construction of $\psi(p_2)$, for every $\underline{p} < p_2 < \overline{p}$ we have:

$$(1+\psi(p_2))\,p_2 = \left(1 + \frac{(1+\delta)\,\left(\overline{p}-\underline{p}\right)}{\left(1-\frac{1+\delta}{2}\right)\,p_2 + \frac{1+\delta}{2}\,\overline{p} - \underline{p}} - 1\right)\,p_2 =$$
$$= \frac{(1+\delta)\,\left(\overline{p}-\underline{p}\right)}{\left(1-\frac{1+\delta}{2}\right)\,p_2 + \frac{1+\delta}{2}\,\overline{p} - \underline{p}}\,p_2,$$

which is an hyperbola whose supremum is for $p_2 \to \overline{p}$. Thus, for every $p_2 < \overline{p}$ we have:

$$(1+\delta)\,\overline{p} > (1+\psi(p_2))\,p_2,$$

and therefore the optimal allocation is for $(p_1 = \overline{p},\, p_2 = \overline{p})$.

The payments are such that $\pi_1 = \underline{p} - \delta\,\overline{p} > 0$ (since the best allocation without ad 1 is $f(2) = 1$ with $p_2 = \underline{p}$) and $\pi_2 = p_1 - p_1 = 0$ (since ad 2 does not introduce any externality in the optimal allocation). Therefore, the revenue of the mechanism is $\underline{p} - \delta\,\overline{p} > 0$.

REVENUE WITH $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$. Initially, we observe that when $p_1 = p_2$, the payments are zero for every pair of declared gain b_1, b_2. Basically, this is because the prices are set by the agents, and therefore the price used in the optimal allocation with both ads and the price used by the VCG payments for the optimal allocation when an ad is removed from the market are the same. Since those prices are the same and $\lambda_1 = \lambda_2$, the values of an ad in

the optimal allocation with both ads and in the allocation when the other ad is discarded are the same. Therefore, a strictly positive revenue of the mechanism is possible only when $p_1 \neq p_2$. In the following, we show that there is no Nash equilibrium with $p_1 \neq p_2$.

Consider any input profile (b_1, p_1, b_2, p_2) in which $p_1 \neq p_2$. As argued above, when $p_1 \neq p_2$, no more than one ad with strictly positive value can be allocated due to the definition of functions q_1, q_2. Assume that ad i is allocated, while ad $-i$ is not. Focus on the case $p_i \leq \overline{p}$. We have $u_{-i} = 0$, ad $-i$ not being allocated. If agent $-i$ inputs $p_{-i} = p_i$ and any $b_{-i} > 0$, then she gets a utility $u_{-i} = q_{-i}(p_i, p_i)\, p_i > 0$ as agent i's payment is $\pi_{-i} = p_i - p_i = 0$. Therefore, agent $-i$ would deviate to play $p_{-i} = p_i$ and any $b_{-i} > 0$. Notice that inputting these values may be not the best response of agent $-i$. However, it is always the case that, if $p_i \neq p_{-i}$ and ad $-i$ is not allocated, then agent $-i$ can increase her utility by playing $p_{-i} = p_i$. Focus on the case $p_i > \overline{p}$. In this case, ad i gets no value from being allocated, and therefore she will reduce the price such that $p_i \leq \overline{p}$. Thus, in the setting provided above, there is no Nash equilibrium in which $p_1 \neq p_2$.

Finally, we show that the inputs $(b_1 = \overline{p}, p_1 = \overline{p}, b_2 = \overline{p}, p_2 = \overline{p})$ are in equilibrium. We notice that for any $b_1, b_2 > 0$, both ads are displayed and the value of the allocations keeps to be $(1 + \delta)\,\overline{p}$. Thus, we analyze possible deviations to different values of $p_1 = p_2 = \overline{p}$. Notice that such deviations would lead the agents to have different values of prices and therefore only one ad is displayed. Focus on agent 1. Any $p_1 > \overline{p}$ would make ad 1 not be allocated. Any $p_1 < \overline{p}$ making ad 1 be the only allocated ad would give agent 1 a value $< \overline{p}$ and would charge agent 1 of $\delta\,\overline{p}$, leading to a utility $u_1 < \overline{p}$. Thus, agent 1 cannot improve her utility by deviating from $p_1 = \overline{p}$. Focus on agent 2. Any $p_2 \neq \overline{p}$ would make ad 2 not be allocated. Thus, agent 2 cannot improve her utility by deviating from $p_2 = \overline{p}$. This means that there is always a pure-strategy Nash equilibrium in which $p_1 = p_2$. Notice that we do not exclude the case in which there are other Nash equilibria than $(b_1 = \overline{p}, p_1 = \overline{p}, b_2 = \overline{p}, p_2 = \overline{p})$. However, any other Nash equilibrium is with $p_1 = p_2$, thus providing a revenue of zero to the mechanism. This shows that the Price of Stability is unbounded in $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$.

REVENUE WITH $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$. The proof in this case follows arguments similar to those used above for the case of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$. In order to guarantee the existence of a Nash equilibrium, we need to assume that ad i is allocated even if she declares a gain $b_i = 0$. Initially, we observe that, when $p_1 = p_2$, there is no Nash equilibrium in which both b_1 and b_2 are strictly larger than 0. Assume by contradiction that both b_1 and b_2 are strictly larger than 0 and $p_1 = p_2$. Assume, w.l.o.g., that $f(i) = 1$ and $f(-i) = 2$. Agent i's

payment is $\pi_i = b_{-i} > 0$, while agent $-i$'s payment is $\pi_{-i} = 0$. Thus, agent i can improve her utility by bidding $b_i = 0$ such that ad i is displayed in the second slot so as to be charged of a payment $\pi_i = 0$. Therefore, we have a contradiction and b_1 and b_2 cannot be both strictly larger than 0 in a Nash equilibrium when $p_1 = p_2$. Notice that, if ad i is not displayed if $b_i = 0$, every agent's best response would be that of making a strictly positive bid smaller than the opponent's bid and therefore the best response dynamics will never reach a fixed point, thus leading to the non-existence of the equilibrium. Notice also that, if $b_i = 0$, then $f(i) = 2$ and the sum of the payments is such that $\pi_i + \pi_{-i} = 0$.

As discussed for the case of the indirect-revelation VCG mechanism, if prices p_1, p_2 are different, the non-allocated ad i can improve her utility by changing her price p_i as $p_i = p_{-i}$. Furthermore, as above, setting just $p_i = p_{-i}$ may be not the best response of agent i, but there is no Nash equilibrium when $p_i \neq p_{-i}$.

Finally, we prove that there is at last a Nash equilibrium when $p_1 = p_2$. In particular, a Nash equilibrium is $(b_1 = \overline{p}, p_1 = \overline{p}, b_2 = 0, p_2 = \overline{p})$. As remarked above, both payments π_1, π_2 are zero as $p_1 = p_2$. Furthermore, as in the case of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$, deviating toward a price different from $\overline{p}$ would make that only a single ad has a strictly positive value from being allocated. If agent i deviates to $p_i < \overline{p}$ and it is the ad that is displayed, then her value is smaller than that she gets when $\overline{p}$. Thus, agent i has no strictly positive incentive to deviate. This shows that $(b_1 = \overline{p}, p_1 = \overline{p}, b_2 = 0, p_2 = \overline{p})$ is a Nash equilibrium and that the revenue of the mechanism is zero. As in the case of the indirect-revelation VCG mechanism, there is no guarantee that such a Nash equilibrium is unique. However, any other Nash equilibrium is with $p_1 = p_2$ and $b_i = 0$ for at least one agent i. Thus, every Nash equilibrium provides a revenue of zero to the mechanism. □

In the specific case of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and $m = 1$, we have a positive result for PoS (PoA is trivially ∞ as it is ∞ even in second-price single-item auctions).

Theorem 5.11. *The PoS for revenue of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ with respect to the mechanism $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ is 1 if $m = 1$.*

Proof. It follows from two considerations:

- with VCG payments, bidding the real b_i and the price p_i that $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ would choose is a Nash equilibrium, and
- with $m = 1$ slot, the payments of $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ and $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ are the same.

This concludes the proof. □

Instead, the above positive result does not hold with $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$, as stated below.

Theorem 5.12. *The PoS for the revenue of $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ is ∞ even if $m = 1$ when agents do not overbid.*

Proof. The proof is based on the following setting.

SETTING. Consider the following setting:

- $n = 2$;
- $m = 1$;
- $q_1(p_1, p_{\min}) = \begin{cases} 1 & \text{if } p_1 \leq \overline{p} \text{ and } p_1 = p_{\min} \\ 0 & \text{otherwise} \end{cases}$;
- $q_2(p_2, p_{\min}) = \begin{cases} 1 & \text{if } p_2 \leq \underline{p} \\ 0 & \text{otherwise} \end{cases}$;
- $c_1 = c_2 = 0$;
- $\alpha_1 = \alpha_2 = 1$;
- $\lambda_1 = 1$,

where $0 < \underline{p} < 0.5\,\overline{p}$.

REVENUE WITH $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$. In The optimal allocation we have $f(1) = 1$ and $f(2) = \perp$ with $p_1 = \overline{p}$ and $p_2 = \underline{p}$. The revenue Rev is $\underline{p} > 0$.

REVENUE WITH $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$. Every Nash equilibrium prescribes that $b_1 > \underline{p} \geq b_2$ and $p_1 = \overline{p}$ as agent 1 can get the first slot, payment π_1 is zero as $p_1 = p_{\min} > p_2$, and the expected value of agent 1 is maximized. Furthermore, payment $\pi_2 = 0$ since ad 2 is not allocated. Thus, revenue Rev is zero leading to a PoS of ∞. □

In the proof of this Theorem 5.12 we strongly rely upon the definition of GSP payments described above, which restricts payments to depend only on agents submitting a price at least $p_{\min}$. This payment format turns out to be necessary in order to guarantee individual rationality. We leave open the problem of understanding if a better Price of Stability for the revenue of $\mathcal{M}_{\mathrm{I}}^{\mathrm{GSP}}$ would be possible by considering alternative non-individually rational GSP payments.

5.5 A Better PoS for the Revenue with Indirect-revelation Mechanisms

As discussed in the previous section, indirect-revelation mechanisms present major weaknesses in terms of efficiency. A natural question is whether we can design indirect-revelation mechanisms with a better efficiency when agents can choose their price. In particular, we focus on $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$, as it always guarantees PoS $= 1$ for the social welfare, and we show that a simple modification of the mechanism leads to PoS $= 1$ for the revenue when some assumptions hold. We call this new mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}*}$. The rationale is to ask agents for more information. More precisely, the input provided by every agent is a triple composed of (b_i, p_i, p_i^*) where (b_i, p_i) is the input to $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ and p_i^* is the price that advertiser i would choose when her ad is the only displayed ad. The property that PoS $= 1$ is guaranteed when function $q_i(p_i, p_i)$ is differentiable in p_i and non-zero in p_i^*. Mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}*}$ is defined as follows:

1. every agent i submits a bid (b_i, p_i, p_i^*), where b_i, p_i, and p_i^* are defined as above;

2. the mechanism infers the values of c_i and α_i for every agent i as follows: $\hat{c}_i = q(p_i^*, p_i^*) \,/\, \frac{dq(p_i,p_i)}{dp_i}\big|_{p_i=p_i^*} + p_i^*$, and $\hat{\alpha}_i = \frac{b_i}{p_i - \hat{c}_i}$ if $p_i \neq \hat{c}_i$ and $\hat{\alpha}_i = 0$ otherwise;

3. the mechanism computes an auxiliary allocation, say $\bar{f}$, by using the allocation function of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ when the input is (b_i, p_i) for every agent i; the corresponding social welfare (evaluated with the declared gain b_i) is $\widehat{\mathrm{SW}}$;

4. for every agent i, the mechanism computes an auxiliary allocation, say $\bar{f}^{-i}$, by using the allocation function of $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ when the values inferred above for $\{\hat{\alpha}_h\}_{h\in N}$ and $\{\hat{c}_h\}_{h\in N}$ are provided in input and agent i is removed from the optimization problem. For every maximization, we denote with $\overline{\mathrm{SW}}^{-i}$ the corresponding social welfare evaluated with the inferred values $\{\hat{\alpha}_h\}_{h\in N}$ and $\{\hat{c}_h\}_{h\in N}$. Notice that, as it happens with $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$, the prices in output to these maximizations can be different from those agents provide in input;

5. if $\widehat{\mathrm{SW}} \geq \max_i \overline{\mathrm{SW}}^{-i}$, then the mechanism chooses allocation $\bar{f}$ and charges every agent i of a payment $\pi_i = \overline{\mathrm{SW}}^{-i} - (\widehat{\mathrm{SW}} - \lambda_{\bar{f}(i)}\, q_i(p_i, p_{\min})\, b_i)$, else no ad is allocated and every agent is charged a payment of zero.

Basically, mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}*}$ exploits the additional information asked to the agents to infer their types and then uses this information to compute the same payments that $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ would charge. Step 5 is necessary to guarantee individual rationality. More precisely, since the allocation $\bar{f}$ is computed as the indirect mechanism does (without optimizing over prices), while the payments $\{\pi_i\}_{i\in N}$ are computed as the direct mechanism does (optimizing over prices), individual rationality may not be satisfied. We solve this problem setting the payments to 0 (and allocating no ads) when the payments $\{\pi_i\}_{i\in N}$ are too large. As a side effect, we have that if the submitted prices are different from the optimal one, it is possible that the mechanism does not assign any slot. Thus, the PoA for the social welfare and revenue can be unbounded.

Theorem 5.13. *Mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}*}$ is individually rational and weakly budget-balanced. Moreover, the PoS for the revenue of $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}*}$ is 1.*

Proof. First we show that the mechanism is individually rational. In particular, we show that $\pi_i \leq \lambda_{\tilde{f}(i)}\, q_i(p_i, p_{\min})\, b_i$ for every $i \in N$, where $\tilde{f}$ is the allocation chosen by the mechanism $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}*}$. We distinguish two cases. If $\widehat{\mathrm{SW}} \geq \max_j \overline{\mathrm{SW}}^{-j}$, then the allocation returned by the mechanism is $\bar{f}$ and the payment $\pi_i = \overline{\mathrm{SW}}^{-i} - \widehat{\mathrm{SW}} + \lambda_{\bar{f}(i)} q_i(p_i, p_{\min}) b_i \leq \lambda_{\bar{f}(i)} q_i(p_i, p_{\min}) b_i$, since $\widehat{\mathrm{SW}} \geq \overline{\mathrm{SW}}^{-i}$. If $\widehat{\mathrm{SW}} \geq \max_j \overline{\mathrm{SW}}^{-j}$, than $\pi_i = 0$ by construction. Therefore, u_i is always non-negative. Moreover, the mechanism is weakly budget-balanced since all payments are at least 0. In particular, if $\widehat{\mathrm{SW}} \geq \max_i \overline{\mathrm{SW}}^{-i}$, the payment of agent i is $\pi_i = \overline{\mathrm{SW}}^{-i} - \widehat{\mathrm{SW}} + \lambda_{\bar{f}(i)} q_i(p_i, p_{\min}) b_i \geq 0$ by the optimality of $\overline{\mathrm{SW}}^{-i}$. Otherwise, all the payments are 0.

Finally, we show that bidding truthfully and with the prices that $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ would charge is an equilibrium with the same revenue of $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$. Let α_i and c_i be the private information of agent i. Moreover, for each agent i, let p_i be the price selected by $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$ with truthful bidding, $b_i = \alpha_i(p_i - c_i)$, and $p_i^* = \arg\max_p \alpha_i q(p,p)(p - c_i)$. Since p_i^* maximizes $\alpha_i q(p,p)(p-c_i)$, then its derivative is 0 in p_i^*, *i.e.*, $\frac{dq(p_i,p_i)}{dp_i}\Big|_{p_i=p_i^*}(p_i^* - c_i) + q(p_i^*, p_i^*) = 0$, implying that $c_i = q(p_i^*, p_i^*) / \frac{dq(p_i,p_i)}{dp_i}\Big|_{p_i=p_i^*} + p_i^*$. This requires that the derivative of $q_i(p_i^*, p_i^*)$ is strictly positive in p_i^*. Hence, $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}*}$correctly computes $\hat{c}_i = c_i$. Moreover, since $b_i = \alpha_i(p_i - c_i)$, then $\alpha_i = \frac{b_i}{p_i - c_i}$ and $\hat{\alpha}_i = \alpha_i$. [3]

[3]Notice that we can assume that in the optimal allocation $p_i - c_i \neq 0$. Otherwise the utility of the agent is 0 and there exists an allocation with the same SW that does not assign any slot to the agent and increases the price and the expected gain.

Hence, since the agents submit the optimal prices, the mechanism computes a $\widehat{\mathrm{SW}}$ that is equivalent to the social welfare of $\mathcal{M}_{\mathrm{D}}^{\mathrm{VCG}}$. Moreover, also $\overline{\mathrm{SW}}^{-i}$ is the same of the direct mechanism for each i, since it optimizes over the real bidders' types. Hence, $\widehat{\mathrm{SW}} \geq \max_i \overline{\mathrm{SW}}^{-i}$ and the mechanism has the same payments and revenue of the direct mechanism. We conclude the proof showing that this is an equilibrium. Similarly to the proof of Theorem 5.4, we have:

$$u_i(\bar{f}, \mathbf{p}, \boldsymbol{\pi}) = \widehat{\mathrm{SW}} - \overline{\mathrm{SW}}^{-i} \geq 0$$

Two cases are possible. If agent i changes the strategy keeping $\widehat{\mathrm{SW}} \geq \overline{\mathrm{SW}}^{-i}$, the claim follows by the optimality of $\widehat{\mathrm{SW}}$. Otherwise, the utilities of all the agents are 0. This concludes the proof. □

We recall that the algorithm we provide to find the best allocation with $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}}$ works when the values that p_i can assume are discrete, and the same holds with $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}*}$. We also notice that $\mathcal{M}_{\mathrm{I}}^{\mathrm{VCG}*}$ requires that p_i^* is not restricted to a set of discrete values, the mechanism could not infer the exact values of α_i and c_i otherwise. However, requiring price p_i to belong to a finite, discrete set of values and price p_i^* to belong to $\mathbb{R}_{\geq 0}$ does not modify the properties of the mechanism since p_i^* is not used in the allocation algorithm.

CHAPTER 6

The Power of Media Agencies in Ad Auctions: Improving Utility through Coordinated Bidding

When a group of competing advertisers is managed by a common agency, many forms of collusion, can be implemented by coordinating bidding strategies, dramatically increasing advertisers' value. We study the computational problem faced by a media agency that has to coordinate the bids of a group of colluders, under GSP and VCG mechanisms. This problem was introduced in Section 1.2, while a formal model is provided in Section 6.1. Section 1.2 presents the optimization problem faced by the media agency, focusing on two settings that differ for the *individual rationality constraints* they require. Such constraints ensure that colluders do *not* leave the agency, and they can be enforced by implementing *monetary transfers* between the agency and the advertisers. In particular, we study the *arbitrary transfers* setting, where any kind of transfer to and from the advertisers is allowed, and the more realistic *limited liability* setting, in which no advertiser can be paid by the agency. Section 6.3 introduces an abstract bid optimization problem, called *weighted utility problem* (WUP), which is useful in proving our results. We show that the utilities of bidding strategies are related to the length of paths in a

directed acyclic weighted graph, whose structure and weights depend on the mechanism under study. This allows us to solve WUP in polynomial time by finding a shortest path in the graph. Section 6.4 provides an approximate solution to the media agency problem with arbitrary transfers constraints. In particular,we cast the problem as a WUP instance and solve it by our graph-based algorithm. Section 6.5 provides an approximate solution to the media agency problem with limited liability. constraints. We formulate it as a linear program with exponentially-many variables efficiently solvable by applying the ellipsoid algorithm to its dual. This requires solving a suitable separation problem in polynomial time, which can be done by reducing it to a WUP instance.

6.1 Model

We study the problem of coordinated bidding faced by a media agency in ad auctions, with both GSP and VCG payments. In this setting, the set $N \coloneqq \{1, \dots, n\}$ of *bidders* (or *agents*) is partitioned as $N \coloneqq N_c \cup N_e$, where: N_c is a set of advertisers whose advertising campaigns are managed by a common media agency, while N_e is a set of advertisers that are *not* part of the agency, but participate in the ad auction individually. In this work, we refer to the former as *colluders*, while we call the latter *external agents*. Moreover, we let $n_c \coloneqq |N_c|$ and $n_e \coloneqq |N_e|$ be the numbers of colluders and external agents, respectively. The advertisers compete for displaying their ads on a set $M \coloneqq \{1, ..., m\}$ of *slots*, with $m \leq n$. Each agent $i \in N$ has a *private valuation* $v_i \in [0, 1]$ for an advertising slot, which reflects how much they value a click on their ad. Furthermore, each slot $j \in M$ is associated with a *click through rate* parameter $\lambda_j \in [0, 1]$, encoding the probability with which the slot is clicked by a user.[1] Each agent $i \in N$ participates in the ad auction with a *bid* $b_i \in [0, 1]$, representing how much they are willing to pay for a click on their ad. We denote by $b = (b_i)_{i \in N}$ the bid profile made by all the agents' bids. We also let $b^c = (b_i)_{i \in N_c}$ be the profile of colluders' bids (also called *bidding strategy*), while $b^e = (b_i)_{i \in N_e}$ is the profile of external agents' bids. For the ease of notation, we sometimes write $b = (b^c, b^e)$ to denote the profile made by all the bids in b^c and b^e.

The media agency knows the valuations v_i of all the colluders $i \in N_c$, and it decides the bid profile b^c on their behalf. Additionally, the media agency defines a monetary *transfer* $q_i \in [-1, 1]$ for each colluder $i \in N_c$.

[1] For the ease of presentation, we assume that the click through rate only depends on the slot and *not* on the ad being displayed. This dependence can be easily captured by interpreting v_i as an expected value w.r.t. clicks once the user observed the slot.

We adopt the convention that, if $q_i > 0$, then the transfer is from the agent to the agency, while, if $q_i < 0$, then it is the other way around.[2]

W.l.o.g., we assume that the slots are ordered in decreasing value of click through rate, so that $\lambda_1 \geq \ldots \geq \lambda_m$. Moreover, for the ease of presentation, we let $\lambda_{m+1} = \ldots = \lambda_{n+1} = 0$.

The auction goes on as follows. First, the media agency selects a bidding strategy $b^c = (b_i)_{i \in N_c}$ and requires a transfer q_i from each colluder $i \in N_c$. Then, external agents individually report their bids to the auction mechanism, resulting in a profile $b^e = (b_i)_{i \in N_e}$, while the media agency reports bids b^c on behalf of the colluders. Finally, given all the agents' bids $b = (b^c, b^e)$, the mechanism allocates an ad to each slot and defines an *expected payment* $\pi_i(b) \in [0, 1]$ for each agent $i \in N$, where the expectation is with respect to the clicks. The media agency is responsible of paying the auction mechanism on behalf of the colluders.

Given a bid profile $b = (b_i)_{i \in N}$, assuming w.l.o.g. that each bidder $i \in [m]$ is assigned to slot i (by re-labeling bidders accordingly), we denote bidder i's *expected revenue* as $r_i(b) \coloneqq \lambda_i v_i$, while bidder i's *expected utility* is $u_i(b) \coloneqq r_i(b) - q_i$.[3] Instead, the expected utility of the agency is $\sum_{i \in N_c}(q_i - \pi_i(b))$. We also denote with U the cumulative expected utility of all the colluders and the media agency. Formally, it holds $U \coloneqq \sum_{i \in N_c}(r_i(b) - \pi_i(b))$.

Next, we review GSP and VCG mechanisms in ad auctions (see the paper by (Nisan and Ronen, 2001) for their general description). Given a bid profile $b = (b_i)_{i \in N}$, assuming w.l.o.g. that $b_1 \geq \ldots \geq b_n$ (by re-labeling bidders accordingly), both mechanisms orderly assign the first m agents, who are those with the highest bids, to the first m slots, which are those with the highest click through rates. Moreover, the mechanisms assign the following expected payments.

- *GSP mechanism*: $\pi_i^{\text{GSP}}(b) \coloneqq \lambda_i b_{i+1}$ for each agent $i \in [m]$, and $\pi_i^{\text{GSP}}(b) = 0$ for all the other agents.
- *VCG mechanism*: $\pi_i^{\text{VCG}}(b) \coloneqq \sum_{j=i+1}^{m+1} b_j(\lambda_{j-1} - \lambda_j)$ for each agent $i \in [m]$, and $\pi_i^{\text{VCG}}(b) = 0$ for the others.

The VCG payments are such that each agent is charged a payment that is equal to the externalities that they impose on other agents. This makes the VCG mechanism *truthful*, which means that it is a dominant strategy for

[2] Notice that there are some scenarios in which it is in the interest of the media agency to pay a colluder in order to ensure that they stay in the agency; see Example 6.2.

[3] We denote with $[m]$ the set $\{1, \ldots, m\}$.

each agent to report their true valuation to the mechanism, namely $b_i = v_i$ for every $i \in N$. This is *not* the case for the GSP mechanism.

6.2 Problem Formulation

In this section, we introduce the optimization problem faced by the media agency. In words, the goal of the media agency is to find a bidding strategy $b^c = (b_i)_{i \in N_c}$ that coordinates colluders' bids in a way that maximizes the cumulative expected utility U, while at the same time guaranteeing that they are incentivized to be part of the media agency. The rest of the section is devoted to formally defining such a problem.

Before introducing the optimization problem, let us notice that knowing the valuations of all the colluders allows the media agency to improve the cumulative expected utility U with respect to the case in which all the bidders act individually. This is formalized by the following proposition.

Proposition 6.1. *The cumulative expected utility U may be arbitrarily larger than the sum of the colluders' expected utilities when they participate in the ad auction individually.*

Proof. There are two colluders, $N_c = \{1, 2\}$, with valuations $v_1 = v_2 + \varepsilon$ and $v_2 \in [0, 1]$ for $\varepsilon > 0$. There is one slot $M = \{1\}$, with click through rate $\lambda_1 = 1$. Notice that, in one-slot settings, VCG and GSP mechanisms define the same payments and are both truthful mechanisms. We consider the following cases:

- *Without media-agency coordination:* the colluders' bid profile is $b^c = (b_1, b_2) = (v_1, v_2)$ and agent 1 wins the slot paying v_2 (since $b_1 > b_2$); then, the cumulative expected utility of the colluders is $U_{\text{w/o}} = u_1(b) + u_2(b) = \varepsilon$.
- *With media-agency coordination:* the colluders' bid profile is $b^c = (b_1, b_2) = (v_1, 0)$ and agent 1 wins the slot paying 0; then, the cumulative expected utility is $U_{\text{w}} = u_1(b) + u_2(b) = v_1 = v_2 + \varepsilon$.

Thus, by letting $\varepsilon \to 0$, we have that $\frac{U_{\text{w}}}{U_{\text{w/o}}} \to +\infty$. □

In general, the media agency may adopt a *randomized* bidding strategy in order to maximize U. By letting B^c be the set of all the possible colluders' bid profiles $b^c = (b_i)_{i \in N_c}$, we denote by $\gamma \in \Gamma$ any randomized bidding strategy, where Γ is the set of all the probability distributions over B^c. Moreover, whenever $\gamma \in \Gamma$ has a finite support, we denote with γ_{b^c} the probability of choosing a bidding strategy $b^c \in B^c$.

In this work, unless stated otherwise, we consider the case in which the bid profile $b^e = (b_i)_{i \in N_e}$ of the external agents is drawn from a probability distribution γ^e. Then, we define the expected revenue of bidder $i \in N_c$ for any bidding strategy $b^c \in B^c$ as $\tilde{r}_i(b^c) \coloneqq \mathbb{E}_{b^e \sim \gamma^e} r_i(b^c, b^e)$, while their expected payment is as $\tilde{\pi}_i(b^c) \coloneqq \mathbb{E}_{b^e \sim \gamma^e} \pi_i(b^c, b^e)$. In the rest of the chapter, we assume that all algorithms have access to an oracle that returns the value of the expectations $\tilde{r}_i(b^c)$ and $\tilde{\pi}_i(b^c)$, for a bidding strategy $b^c \in B^c$ given as input.[4]

The following Problem (6.1) encodes the maximization problem faced by the media agency, where the meaning of IR and AIR constraints is described in the following.

$$\max_{q, \gamma \in \Gamma} \sum_{i \in N_c} \mathbb{E}_{b^c \sim \gamma}[\tilde{r}_i(b^c) - \tilde{\pi}_i(b^c)] \quad \text{s.t.} \tag{6.1a}$$

$$\text{IR}: \quad \mathbb{E}_{b^c \sim \gamma}[\tilde{r}_i(b^c)] - q_i \geq t_i \qquad \forall i \in N_c \tag{6.1b}$$

$$\text{AIR}: \quad \sum_{i \in N_c} q_i \geq \sum_{i \in N_c} \mathbb{E}_{b^c \sim \gamma}[\tilde{\pi}_i(b^c)]. \tag{6.1c}$$

The elements of Problem (6.1) are defined as follows.

Objective (6.1a) encodes the cumulative expected utility U of the colluders and the media agency, in expectation with respect to the randomized bidding strategy γ.

Constraints (6.1b), which are called *individual rationality* (IR) constraints, ensure that the colluders are incentivized to be part of the media agency, rather than leaving it and participating in the ad auction as external agents. In particular, they guarantee that each colluder $i \in N_c$ achieves at least a minimum expected utility t_i, where the values $t_i \in [0, 1]$ for $i \in N_c$ are given as input.[5]

Constraint (6.1c) is an *agency individual rationality* (AIR) constraint which provides guarantees over the utility of the media agency. Since the agency corresponds to the mechanism a payment $\sum_{i \in N_c} \mathbb{E}_{b^c \sim \gamma}[\tilde{\pi}_i(b^c)]$ in expectation over the clicks, the constraint requires that the sum of transfers $\sum_{i \in N_c} q_i$ covers the payment, so that the expected utility attained by the agency is non-negative.

[4] Our results can be easily extended—only incurring in a small additive loss in cumulative expected utility—to the case in which the distribution γ^e is unknown, but the algorithms have access to a black-box oracle that returns i.i.d. samples drawn according to γ^e (rather than returning expected values).

[5] To the best of our knowledge, in the literature there is only one work by Bachrach [2010] that formalizes IR constraints for a setting that is similar to ours. Bachrach [2010] takes inspiration from the concept of core Peleg and Sudhölter (2007) in cooperative games in order to define suitable IR constraints. However, this approach has many downsides. The most relevant issue of such an approach is that it is *not* computationally viable, since computing the core would require exponential time in the number of colluders.

In the following, we sometimes relax IR constraints by using δ-IR constraints, for $\delta > 0$, which are defined as follows:

$$\delta\text{-IR}: \quad \mathbb{E}_{b^c \sim \gamma}[\tilde{r}_i(b^c)] - q_i \geq t_i - \delta \qquad \forall i \in N_c. \tag{6.2}$$

In the following, we call the scenario described so far, in which transfers q_i could be negative, the *arbitrary transfers* setting. Monetary transfers from the media agency to the agents are *not* always feasible in practice. Indeed, in some real-world scenarios a media agency could potentially lose customers by adopting a strategy for which some agents pay and some others are paid for participating in the same auction. For these reasons, we introduce and study a second scenario, which we call *transfers with limited liability* setting, where no agent is paid by the agency. In such setting, Problem (6.1) is augmented with the following additional *limited liability* (LL) constraints on the monetary transfers q_i:

$$\text{LL}: \quad q_i \geq 0 \qquad \forall i \in N_c. \tag{6.3}$$

As we show in Section 6.4, there always exists an optimal solution to Problem (6.1) without LL constraints that is *not* randomized. The same does not hold for the problem with LL constraints, in which an optimal bidding strategy may be randomized, as in Example 6.2.

We conclude by introducing the following assumption on the values t_i, guaranteeing that Problem (6.1) is feasible.

Assumption 6.1. *There always exists a bidding strategy $b^c \in B^c$ such that $\tilde{r}_i(b^c) - \tilde{\pi}_i(b^c) \geq t_i$ for all $i \in N_c$.*

Proposition 6.2. *With Assumption 6.1, Problem* (6.1) *with LL constraints admits a non-randomized feasible solution.*

Proof. Suppose that there exists a bidding strategy $b^c \in B^c$ such that $\tilde{r}_i(b^c) - \tilde{\pi}_i(b^c) \geq t_i$ for all $i \in N_c$. We prove the proposition by showing that Problem (6.1) has a feasible solution composed by the non-randomized bidding strategy b^c and transfers $q_i = \tilde{\pi}_i(b^c)$ for all $i \in N_c$. Constraints (6.1b) are satisfied because of the condition $\tilde{r}_i(b^c) - \tilde{\pi}_i(b^c) \geq t_i$ for all $i \in N_c$. Then, by substituting $q_i = \tilde{\pi}_i(b^c)$ in Constraint (6.1c), we show that also the AIR constraint is satisfied. We conclude the proof observing that under GSP and VCG mechanisms $\tilde{\pi}_i(b^c) > 0$ for all $i \in N_c$, therefore also LL constraints are satisfied. □

In the following example, we show how the set of feasible solutions depends on parameters t_i, with $i \in N_c$.

Example 6.1. *In the proof of Proposition 6.1 we consider a deterministic bidding strategies $b^c = (v_1, 0)$ played with probability $\gamma_{b^c} = 1$. Agent 2 has the same utility $u_2 = 0$ with or without the coordination of the agency. Moreover, agent 2 is incentivized to stay under the agency when $t_2 = 0$. However, suppose that $t_2 = \delta$ and $t_1 = \varepsilon$. In this case, the optimal expected utility $U = v_1$, achieved by the strategy $b^c = (v_1, 0)$, should be partitioned between the agents s.t. $u_2(b) \geq \delta$. A feasible solution is $q_1 = \delta$ and $q_2 = -\delta$ (recall that $\pi_1(b) = \pi_2(b) = 0$). The corresponding utilities are $u_1(b) = r_1(b) - q_1 = v_1 - \delta$ for agent 1 and $u_2(b) = r_2(b) - q_2 = \delta$ for agent 2. Under the hypothesis that $v_1 - \delta \geq \varepsilon$, this is a feasible solution maximizing the objective function (6.1a).*

Then, we show a case in which a randomized bidding strategy can provide a higher utility than the best non-randomized bidding strategy.

Example 6.2. *We show a case in which the best non-randomized bidding strategy provides a lower utility w.r.t. the best randomized bidding strategy. Consider a set of two slots $M = \{1, 2\}$ with click-through rates $\lambda_1 \geq \lambda_2$, a set of two colluders $N_c = \{1, 2\}$ and a set of one external agent $N_e = \{3\}$. It holds that $v_3 \leq v_2 \leq v_1$. The ties are broken in favour of the colluders. Suppose that $v_1 = v_2 = 1$, $v_3 = \frac{2}{3}$, $\lambda_1 = \lambda_2 = 1$, $t_1 = \delta_1$, and $t_2 = \delta_2$, with δ_1 and δ_2 arbitrarily small values such that $\delta_1, \delta_2 < \frac{1}{3}$. Consider the non-randomized bidding strategy $b_1 = v_1$, $b_2 = b_3$. Consider the randomized bidding strategy: play with probability $\gamma_{b^{c'}} = \frac{1}{2}$ stategy $(\epsilon, 0)$ and play with probability $\gamma_{b^{c''}} = \frac{1}{2}$ stategy $(0, \epsilon)$, with small ϵ. Under both VCG and GSP mechanisms we have that:*

- *The non-randomized strategy provides cumulative utility $U_{NR} = u_1(b^c) + u_2(b^c) = \frac{1}{3} + \frac{1}{3} = \frac{2}{3}$. The IR and AIR constraints can be easily satisfied, for instance, by setting $q_1 = q_2 = 0$.*
- *The randomized strategy provides a cumulative utility $U_{R'} = 1$ when bidding $b^{c'}$, and provides $U_{R''} = 1$ when bidding $b^{c''}$. The expected cumulative utility on the stochasticity of the bidding strategy is $U_R = 1$. The IR and AIR constraints can be satisfied by setting $q_1 \in \left[\delta_1, \frac{1}{2}\right]$ and $q_2 \in \left[\delta_2, \frac{1}{2}\right]$.*
- $U_R > U_{NR}$

Now, we provide some examples of natural choices for the values t_i, which arise from allocation and payment rules of the considered auction mechanisms and satisfy Assumption 6.1.

Consider a VCG mechanism and fix t_i as the utility of agent i, for every $i \in N_c$, when all agents in N paritcipate to the auction without the coordination of the agency. With such choice of parameters t_i, a feasible solution to the optimization problem (6.1) with LL always exists by fixing the bidding strategy $b^c = (v_i)_{i \in N_c}$.

As for the GSP mechanism, there are many possible choices of value t_i as there is no dominant strategy. For instace, consider the Balanced Bidding strategy from Cary et al. [2007]. Parameter t_i can be set as the expected utility of agent i, when the colluders adopt such strategy considering the external bids in expectation.

6.3 Weighted Utility Problem

In this section, we provide a polynomial-time algorithm for an abstract bid optimization problem, called *weighted utility problem* (WUP). This will be crucial in the following sections in order to solve Problem (6.1) with or without LL constraints.

Let $\Delta \coloneqq \{\delta_1, \ldots, \delta_d\}$ be a discrete set of d different bid values, with $\delta_1 \geq \ldots \geq \delta_d$. Moreover, given a bidding strategy $b^c = (b_i)_{i \in N_c} \in B^c$ such that $b_i \in \Delta$ for all $i \in N_c$, we write $b^c \in B^c_\Delta$ to denote that each b_i belongs to Δ.

Then, WUP reads as follows:

$$\max_{b^c \in B^c_\Delta} \sum_{i \in N_c} (\hat{y}_i \, \tilde{r}_i(b^c) - \hat{x} \, \tilde{\pi}_i(b^c)), \tag{6.4}$$

where $\hat{y}_i \geq 0$ for $i \in N_c$ and $\hat{x} \geq 0$ are give problem parameters. A solution to Problem (6.4) is a bidding strategy $b^c \in B^c_\Delta$ that maximizes the sum of suitable *weighted* utilities of the colluders, which are defined so that colluder i's expected revenue $\tilde{r}_i(b)$ is weighted by coefficient $\hat{y}_i$, while their expected payment $\tilde{\pi}_i(b)$ is weighted by coefficient $\hat{x}$. Notice that, when $\hat{y}_i = 1$ for all $i \in N_c$ and $\hat{x} = 1$, then the objective of Problem (6.4) coincides with the cumulative expected utility U.

As a first step, we consider Problem (6.4) in which the profile of external agents' bids b^e is fixed, which reads as follows:

$$\max_{b^c \in B^c_\Delta} \sum_{i \in N_c} (\hat{y}_i \, r_i(b^c, b^e) - \hat{x} \, \pi_i(b^c, b^e)). \tag{6.5}$$

For the sake of presentation, we first provide our results for Problem (6.5), and, then, we show how they can be extended to Problem (6.4), where the

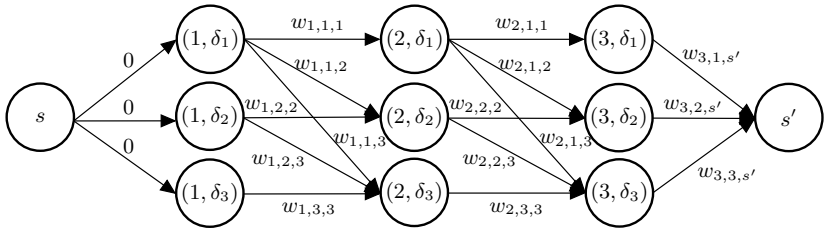


Figure 6.1: *Example of graph for* $N_c = \{1, 2, 3\}$, $\Delta = \{\delta_1, \delta_2, \delta_3\}$.

bids of external agents are stochastic. Thus, in the rest of the section, we always assume that a profile of external agent's bids $b^e = (b_i)_{i \in N_e}$ is given.

The main idea underpinning our results is to map Problem (6.4) into a *shortest path problem* Dijkstra (1959). Specifically, we show that the weighted utilities of bidding strategies are related to the length of paths in a particular *directed acyclic weighted graph*, whose structure and weights derive from the considered auction mechanism. The following lemma is crucial for the construction of the graph.

Lemma 6.1. *Given any bidding strategy* $b^c = (b_i)_{i \in N_c} \in B^c_\Delta$ *that is optimal for Problem* (6.5)*, it holds* $b_i \geq b_j$ *for every pair of colluders* $i, j \in N_c$ *such that* $\hat{y}_i v_i \geq \hat{y}_j v_j$.

Proof. We prove the result for GSP auction mechanisms. Consider a set of colluders, *i.e.* N_c in which colluders are ordered such that, if $i, j \in N_c$ and $\hat{y}_i v_i > \hat{y}_j v_j$, then $i < j$. Consider two colluders h and k, having weighted valuations defined as $v^{\text{w}}_h := \hat{y}_h v_h$ and $v^{\text{w}}_k := \hat{y}_k v_k$ such that $v^{\text{w}}_h \geq v^{\text{w}}_k$. Consider a set of external bidders N_e, their bid profile b^e and a set of slot $M = \{1, \ldots, m\}$ such that their click-through rates are $\lambda_1 > \ldots > \lambda_m$. Suppose that ties are broken in favor of the colluders and fix a bidding strategy b^c, composed of bids in non-increasing order, *i.e.*, if $b_i, b_j \in b^c$ and $b_i > b_j$, then $i < j$. Consider GSP allocation function f such that $f(i) = j$ if agent i is allocated to slot j. The expected utility is:

$$
\begin{aligned}
U =& \sum_{i \in N_c} U_i \\
=& U_h + U_k + \sum_{i \in N_c \setminus \{h,k\}} U_i = \\
=& \lambda_{f(h)} \left(v^{\text{w}}_h - \max \left\{ \max_{\ell \in N_c : b_\ell \leq b_h \wedge \ell > h} b_\ell, \max_{j \in N_e : b_j \leq b_h} b_j \right\} \right) + \\
&+ \lambda_{f(k)} \left(v^{\text{w}}_k - \max \left\{ \max_{\ell \in N_c : b_\ell \leq b_k \wedge \ell > h} b_\ell, \max_{j \in N_e : b_j \leq b_k} b_j \right\} \right) +
\end{aligned}
$$

$$
\begin{aligned}
&+ \sum_{i \in N_c \setminus \{h,k\}} \lambda_{f(i)} \left(v_i^{\mathrm{w}} - \max \left\{ \max_{\ell \in N_c : b_\ell \leq b_i \wedge \ell > h} b_\ell, \max_{j \in N_e : b_j \leq b_i} b_j \right\} \right) \geq \\
\geq &\lambda_{f(h)} \left(v_k^{\mathrm{w}} - \max \left\{ \max_{\ell \in N_c : b_\ell \leq b_h \wedge \ell > h} b_\ell, \max_{j \in N_e : b_j \leq b_h} b_j \right\} \right) + \\
&+ \lambda_{f(k)} \left(v_h^{\mathrm{w}} - \max \left\{ \max_{\ell \in N_c : b_\ell \leq b_k \wedge \ell > h} b_\ell, \max_{j \in N_e : b_j \leq b_k} b_j \right\} \right) + \\
&+ \sum_{i \in N_c \setminus \{h,k\}} \lambda_{f(i)} \left(v_i^{\mathrm{w}} - \max \left\{ \max_{\ell \in N_c : b_\ell \leq b_i \wedge \ell > h} b_\ell, \max_{j \in N_e : b_j \leq b_i} b_j \right\} \right) = \\
= &\hat{U},
\end{aligned}
$$

where $\hat{U}$ is the utility provided by bidding strategy b^c in which b_h and b_k are switched. Therefore, the maximum utility is reached through a bidding strategy ordered in non-increasing values of b_i, where $i \in N_c$.

The result can be analogously proved for VCG auction mechanism. □

In the rest of this section, for the ease of notation and w.l.o.g. thanks to Lemma 6.1, we re-label bidders in N so that $N_c \coloneqq \{1, \ldots, n_c\}$ and $\hat{y}_i v_i \geq \hat{y}_j v_j$ for any $i, j \in N_c : i < j$. Moreover, w.l.o.g., we also re-label external agents so that $N_e \coloneqq \{n_c + 1, \ldots, n\}$ and $b_i \geq b_j$ for any $i, j \in N_e : i < j$.

We build the graph as follows (see Figure 6.1).

- The set of vertices V contains dn_c nodes, plus a source node s and a sink node s'. In particular, there are d nodes for each colluder, one for each possible bid value. For every $i \in N_c$ and $\delta_j \in \Delta$, we let (i, δ_j) be the node corresponding to colluder i and bid value δ_j. Intuitively, selecting a path passing through such a node encodes the fact that the bid b_i of colluder i is set to value δ_j.

- The set of arcs A has cardinality $O(d^2 n_c)$. In particular, the source node s is connected to all the nodes $(1, \delta_j)$ (for $\delta_j \in \Delta$), while all nodes (n_c, δ_j) (for $\delta_j \in \Delta$) are connected to the sink node s'. Moreover, for every $i \in N_c \setminus \{n_c\}$ and $\delta_j \in \Delta$, node (i, δ_j) is connected to all nodes $(i + 1, \delta_{j'})$ such that $\delta_{j'} \in \Delta$ and $j' \geq j$. Intuitively, each path in the graph going from s to s' defines a bidding strategy $b^c = (b_i)_{i \in N_c} \in B^c_\Delta$ such that $b_i \geq b_j$ for every $i, j \in N_c$ with $\hat{y}_i v_i \geq \hat{y}_j v_j$. Notice that focusing on such bidding strategies is w.l.o.g. by Lemma 6.1. In the following, we denote the arc going from (i, δ_j) to $(i + 1, \delta_{j'})$ with the tuple $(i, \delta_j, \delta_{j'})$.

- Each arc $(i, \delta_j, \delta_{j'})$ has a weight $w_{i,j,j'}$. For $i \in N_c \backslash \{n\}$ and $\delta_j, \delta_{j'} \in \Delta$, the weight $w_{i,j,j'}$ encodes the fraction of cumulative weighted utility obtained by setting $b_i = \delta_j$ and $b_{i+1} = \delta_{j'}$, once all the (higher) bids $b_{i'}$ with $i' \in N_c : i' < i$ are assigned to a value given the preceding nodes in the selected path. The weights $w_{n_c,j,s'}$ on arcs going from nodes (n_c, δ_j) (for $\delta_j \in \Delta$) to the sink s' are defined analogously, while those on arcs exiting from the source s are denied as $w_{s,j} = 0$ for all $\delta_j \in \Delta$.

A *directed path* $\sigma \in \Sigma$ is a sequence of arcs connecting the source node s to the sink node s'. The *length* W_σ of path σ is the sum of the weights of the arcs in the path:

$$W_\sigma = \sum_{(i,\delta_j,\delta_{j'}) \in \sigma} w_{i,j,j'}.$$

Then, the shortest path problem on the weighted graph is

$$\min_{\sigma \in \Sigma} -W_\sigma.$$

In conclusion, we define the weights of all arcs $(i, \delta_j, \delta_{j'})$, which depend on the considered auction mechanism, either GSP or VCG. The weights $w_{n_c,j,s'}$ of arcs entering the sink s' are defined analogously, by letting $\delta_{j'} = 0$. For the ease of notation, given $\delta_j \in \Delta$, we let $\tau(\delta_j) \coloneqq \sum_{i \in N_e} \mathbf{1}_{\{b_i > \delta_j\}}$ be the number of external agents with a bid larger than δ_j.

For the GSP mechanism, the weight of arc $(i, \delta_j, \delta_{j'})$ is

$$w_{i,j,j'} \coloneqq \lambda_{i+\tau(\delta_j)} \left(\hat{y}_i v_i - \hat{x} \max \left\{ \delta_{j'}, \max_{k \in N_e : b_k < \delta_j} b_k \right\} \right).$$

Intuitively, for the GSP mechanism, the weight $w_{i,j,j'}$ is exactly equal to the weighted utility of colluder i when they bid value δ_j and the colluder $i+1$ bids value $\delta_{j'}$.

For the VCG mechanism, the weight of arc $(i, \delta_j, \delta_{j'})$ is

$$w_{i,j,j'} \coloneqq \hat{y}_i \, \lambda_{i+\tau(\delta_j)} v_i - \hat{x} \Big[g_i(\delta_j) + \ell_i(\delta_j, \delta_{j'}) \Big],$$

where we let

$$g_i(\delta_j) \coloneqq (i-1)\, \delta_j \left(\lambda_{i+\tau(\delta_j)-1} - \lambda_{i+\tau(\delta_j)} \right),$$
$$\ell_i(\delta_j, \delta_{j'}) \coloneqq \sum_{\substack{k \in N_e : \\ b_k \in (\delta_{j'}, \delta_j]}} i \, b_k \Big(\lambda_{k-n_c+i-1} - \lambda_{k-n_c+i} \Big).$$

For the VCG mechanism, $w_{i,j,j'}$ has a less intuitive interpretation than for the GSP mechanism. In particular, $w_{i,j,j'}$ is composed of a revenue term, which is agent i's expected revenue $\lambda_{i+\tau(\delta_j)}v_i$, weighted by $\hat{y}_i$, and two payment terms, $g_i(\delta_j)$ and $\ell_i(\delta_j, \delta_{j'})$, weighted by $\hat{x}$. The latter are two parts of the cumulative payment from the agency to the mechanism, which are related to the externalities of colluders $i' \in N_c$ with $i' < i$, when colluders i and $i+1$ bid δ_j and $\delta_{j'}$, respectively. In particular, the fraction of the expected payment related to some colluder $i' \in N_c$ with $i' < i$, due to the presence of colluder i bidding δ_j, is $\delta_j(\lambda_{i+\tau(\delta_j)-1} - \lambda_{i+\tau(\delta_j)})$. The term $g_i(\delta_j)$ defines the summation of such payments over all agents $i' \in N_c : i' < i$. Moreover, the fraction of expected payment related to i', due to the presence of an external agent $k \in N_e$ bidding b_k, is $b_k(\lambda_{k-n_c+i-1} - \lambda_{k-n_c+i})$. The term $\ell_i(\delta_j, \delta_{j'})$ is the summation of such expected payments due to external agents with bids $b_k \in (\delta_{j'}, \delta_j]$.

The following lemma establishes the relation between the length of the paths in the weighted graph defined above and the objective of Problem (6.5).

Lemma 6.2. *Given any path $\sigma \in \Sigma$ composed by the sequence of nodes $\{(1, \delta_{j_1}), \ldots, (n_c, \delta_{j_{n_c}})\}$, it holds*

$$W_\sigma = \sum_{i \in N_c} \hat{y}_i r_i(b^c, b^e) - \hat{x}\pi_i(b^c, b^e), \tag{6.6}$$

where $b^c = (\delta_{j_1}, \ldots, \delta_{j_{n_c}})$. Moreover, for any bidding strategy $b^c = (b_i)_{i \in N_c} \in B^c_\Delta$, there exists a corresponding path composed by the sequence of nodes $\{(1, b_1), \ldots, (n_c, b_{n_c})\}$.

Proof. We prove that every path in the graph corresponds to a specific bidding strategy in the auction such that the sum of the weights W_σ of the arcs in path σ is the weighted utility in the optimization Problem (6.5) given the bidding strategy corresponding to that path, and viceversa. First, we prove the existence of a bidding strategy given a path, then, we show the existence of a path given a bidding strategy. The proof follows from the construction of the graph. Recall that, in this section, for the ease of notation and w.l.o.g. thanks to Lemma 6.1, we re-label bidders in N so that $N_c \coloneqq \{1, \ldots, n_c\}$ and $\hat{y}_i v_i \geq \hat{y}_j v_j$ for any $i, j \in N_c : i < j$. Moreover, w.l.o.g., we also re-label external agents so that $N_e \coloneqq \{n_c + 1, \ldots, n\}$ and $b_i \geq b_j$ for any $i, j \in N_e : i < j$.

- Given a path $\{(1, \delta_j), \ldots, (n_c, \delta_{j'})\}$, the corresponding bidding strategy is $b_1 = \bar{b}^c_j, \ldots, b_{n_c} = \bar{b}^c_{j'}$.

- Given a bidding strategy $b_1 = \delta_j, \ldots, b_{n_c} = \delta_{j'}$, such that w.l.o.g. $b_1 \geq \ldots \geq b_{n_c}$ by Lemma 6.1, the corresponding path is $\{(1, \delta_j), \ldots, (n_c, \delta_{j'})\}$, which always exists by the way in which the graph is built.

Now, we prove Equation (6.6) under GSP and VCG mechanisms.

In GSP auction, the weight $w_{i,j',j}$ is colluder i's weighted utility under GSP payments when i is assigned to the $(i + \tau(\delta_i))$-th slot:

$$W_\sigma = \sum_{(i,\delta_{j'},\delta_j)\in\sigma} w_{i,j',j} = \sum_{i\in N_c} \hat{y}_i r_i(b^c) - \hat{x}\pi_i^{\text{GSP}}(b^c). \tag{6.7}$$

The equivalence between the solutions of the shortest path problem and Problem (6.5) follows from equality (6.7) and from the fact that the following optimization problem is a shortest path problem:

$$\max_{\sigma\in\Sigma} W_\sigma. \tag{6.8}$$

Now, we analyze the case of VCG mechanism.

We let $\rho(b_j) \coloneqq \sum_{i\in N_c} \mathbf{1}_{\{b_i \geq b_j\}}$ be the number of colluders with a bid larger than or equal to the external bid b_j, with $j \in N_e$. We define $n(b_j^e) = \sum_{k=1}^{n_c} \mathbf{1}1_{\{b_k^c \geq b_j^e\}}$ the number of colluders with a bid larger than or equal to external bid b_j^e. We have:

$$\begin{aligned}
W_\sigma &= \sum_{(i,\delta_{j'},\delta_j)\in\sigma} w_{i,j',j} = \\
&= \sum_{i\in N_c} \Bigg(\hat{y}_i r_i(b^c) - \hat{x} \Bigg((i-1) b_i (\lambda_{i+\tau(b_i)-1} - \lambda_{i+\tau(b_i)}) - \\
&\quad - \sum_{h\in N_e : b_h \in (b_{i+1}, b_i]} i\, b_h (\lambda_{h+i-1} - \lambda_{h+i}) \Bigg) \Bigg) = \\
&= \sum_{i\in N_c} \Bigg(\hat{y}_i r_i(b^c) - \hat{x} \Bigg(\sum_{i'\in N_c : i' > i} b_{i'} (\lambda_{i'+\tau(b_{i'})-1} - \lambda_{i'+\tau(b_{i'})}) - \\
&\quad - \sum_{j\in N_e : b_j \leq b_i} b_j^e (\lambda_{j+\rho(b_j)-1} - \lambda_{j+\rho(b_j)}) \Bigg) \Bigg) = \\
&= \sum_{i\in N_c} \Bigg(\hat{y}_i r_i(b^c) - \hat{x} \sum_{j=i+1}^{m+1} \bar{b}_j (\lambda_{j-1} - \lambda_j) \Bigg) = \\
&= \sum_{i\in N_c} \left(\hat{y}_i r_i(b^c) - \hat{x}\pi_i^{\text{VCG}}(b^c) \right)
\end{aligned}$$

where, for the ease of notation, we let $\lambda_{m+1} = 0$ and $\bar{b}$ is a bid profile including both colluders' and external agents' bids ordered in non-increasing order, *i.e.*, given b_i and b_j in $\bar{b}$ they are such that $i < j$ when $\bar{b}_i > \bar{b}_j$, and $i, j \in N_c \cup N_e$. □

The following theorem provides the polynomial-time algorithm for solving Problem (6.5), which works by simply finding a shortest path of the graph defined above.

Theorem 6.1. *Problem* (6.5) *can be solved in polynomial time.*

Finally, by substituting the quantities involved in Problem (6.5) with their expectations, thanks to the linearity of the objective, we get the following result:

Theorem 6.2. *Problem* (6.4) *can be solved in polynomial time.*

Proof. It suffices to solve the shortest path problem by assigning to each arc $(i, \delta_{j'}, \delta_j)$ the weight computed in expectetion w.r.t. the distribution of b^e, which we denote by $\tilde{w}_{i,j',j} \coloneqq \mathbb{E}_{b^e \sim \gamma^e}[w_{i,j',j}]$. The result follows from the linearity of the objective function of the Problem (6.5).

$$\begin{aligned}
\sum_{(i,\delta_{j'},\delta_j)\in\sigma} \tilde{w}_{i,j',j} &= \sum_{(i,\delta_{j'},\delta_j)\in\sigma} \mathbb{E}_{b^e\sim\gamma^e_{b^e}}[w_{i,j',j}] = \\
&= \mathbb{E}_{b^e\sim\gamma^e}\left[\sum_{(i,\delta_{j'},\delta_j)\in\sigma} w_{i,j',j}\right] = \\
&= \mathbb{E}_{b^e\sim\gamma^e}\left[\sum_{i\in N_c} \hat{y}_i r_i(b^c) - \hat{x}\pi_i(b^c)\right] = \\
&= \sum_{i\in N_c} \hat{y}_i \tilde{r}_i(b^c) - \hat{x}\tilde{\pi}_i(b^c)
\end{aligned}$$

□

6.4 Arbitrary Transfers Setting

In this section, we provide an approximate solution to the media agency problem with arbitrary transfers. In particular, we design a bi-criteria additive FPTAS that returns solutions providing an arbitrary small loss $\varepsilon > 0$ with respect to the optimal value of the problem, by relaxing the IR constraints by the additive factor ε. As a first step, we show that there always exists a non-randomized solution to Problem (6.1).

Algorithm 6.1 REC$((\alpha, \beta], p, \eta)$

1: **if** $\mathbb{P}_{b^e \sim \gamma^e} \{\exists j \in N_e : b_j \in (\alpha, \beta]\} \leq p \vee \beta - \alpha \leq \eta$ **then return** $\{(\alpha, \beta]\}$
2: **else**
3: $\quad \mathcal{I}_{\text{L}} \leftarrow \text{REC}\left(\left(\alpha, \frac{\alpha+\beta}{2}\right], p, \eta\right)$
4: $\quad \mathcal{I}_{\text{R}} \leftarrow \text{REC}\left(\left(\frac{\alpha+\beta}{2}, \beta\right], p, \eta\right)$ **return** $\mathcal{I}_L \cup \mathcal{I}_{\text{R}}$
5: **end if**

Lemma 6.3. *In the arbitrary transfers setting, there always exists an optimal non-randomized solution to Problem* (6.1)*.*

Proof. We provide an optimal solution (γ^*, q) to LP (6.1) such that $\gamma^*_{b^{c,*}} = 1$ for a bid profile $b^{c,*}$. Let $b^{c,*} \in \text{argmax}_{b^c \in B^c} \sum_{i \in N_c} \tilde{r}_i(b^c) - \tilde{\pi}_i(b^c)$. Moreover, let $q_i = \tilde{r}_i(b^{c,*}) - t_i$. By the optimality of $b^{c,*}$, the value of the objective (6.1a) is optimal. Moreover, Constraints (6.1b) are satisfied by construction. To conclude the proof, we show that constraint (6.1c) is satisfied. In particular $\sum_{i \in N_c}[q_i - \tilde{\pi}_i(b^{c,*})] = \sum_{i \in N_c}(\tilde{r}_i(b^{c,*}) - \tilde{\pi}_i(b^{c,*}) - t_i) \geq 0$ by Assumption 6.1. □

Then, we show how to reduce the problem to a new one working with a finite (discretized) set of bids, in order to apply the results provided in Section 6.3. As a first result, given a probability value $p \in [0, 1]$ and a minimum discretization step $\eta \in [0, 1]$, we show how to split the space of bids $[0, 1]$ into a suitably-defined set of intervals using the recursive algorithm whose pseudo-code is provided in Algorithm 6.1.

We prove the following:

Lemma 6.4. *Given $p \in [0, 1]$ and $\eta \in [0, 1]$,* REC$((0, 1], p, \eta)$ *returns a set $\{(\alpha_j, \beta_j]\}_{j \in [k^*]}$ composed of $k^* \leq \frac{2n_e}{p} \log \frac{1}{\eta}$ intervals such that, for every interval $(\alpha_j, \beta_j]$, it holds either $\mathbb{P}_{b^e \sim \gamma^e} \{\exists i \in N_e : b_i \in (\alpha_j, \beta_j]\} \leq p$ or $\beta_j - \alpha_j \leq \eta$. Moreover, it holds that $\bigcup_{j \in [k^*]} (\alpha_j, \beta_j] = (0, 1]$ and the procedure runs in time polynomial in n_e, $\frac{1}{p}$, and $\log \frac{1}{\eta}$.*

Proof. It is easy to see that for each interval $(\alpha_j, \beta_j]$ returned by algorithm REC$((\alpha_j, \beta_j], p, \eta)$ it holds $\mathbb{P}_{b^e \sim \gamma^e}(\exists i \in N_e : b_i \in (\alpha_j, \beta_j]) \leq p$ or $\beta_j - \alpha_j \leq \eta$ and that the union of the returned intervals is $(0, 1]$. Then, we prove that $k^* \leq \frac{2n_e}{p} \log\left(\frac{1}{\eta}\right)$. Suppose by contradiction that Algorithm 6.1 returns $k^* > \frac{2n_e}{p} \log\left(\frac{1}{\eta}\right)$ intervals $\{(\alpha_j, \beta_j]\}_{j \in [k^*]}$. Each of these intervals has been generated by the recursive call of algorithm $REC((\alpha_j, \beta_j], p, \eta)$ from an interval I_h such that $\mathbb{P}_{b^e \sim \gamma^e}(\exists i \in N_e : b_i \in I_h) > p$ and $(\alpha_j, \beta_j] =$

$(\alpha_h, \frac{\alpha_h+\beta_h}{2}]$ or $(\alpha_j, \beta_j] = (\frac{\alpha_h+\beta_h}{2}, \beta_h]$. We call $\mathcal{I} = \cup_{h\in[k^*]}\{I_h\}$ the set of all such intervals I_h. If an interval does not respect the condition at Line 1, Algorithm 6.1 performs two recursive calls and, therefore, that interval can be seen as an internal node of a binary tree. Vice versa, each interval satisfying such condition is a leaf node. In a binary tree the number of internal nodes is at least half the number of leaf nodes, hence the cardinality of $\mathcal{I}$ is at least $C = \frac{k^*}{2} > \frac{n_e}{p}\log\left(\frac{1}{\eta}\right)$. Moreover, at least $\frac{C}{\log\left(\frac{1}{\eta}\right)} > \frac{\frac{n_e}{p}\log\left(\frac{1}{\eta}\right)}{\log\left(\frac{1}{\eta}\right)} = \frac{n_e}{p}$ intervals $I_h \in \mathcal{I}$ are disjoint. We denote by $\mathcal{I}_d$ the set of such disjoint intervals. This results follows from the tree representation introduced above. Each interval, corresponding to a node N, is disjoint from another interval, corresponding to a node N', if N is neither a parent nor a child of N'. We refer to N and N' as disjoint nodes. Notice that all the nodes in the same level are disjoint. The number of such nodes can be bounded by $\frac{C}{h}$, where h is the depth of the tree. In particular, the C internal nodes are partitioned over $h-1$ levels of the tree. Thus, there exists at least a level with $\frac{C}{h-1}$ nodes. The result follows from the fact that the maximum depth is at most $\log\frac{1}{\eta}$. This implies that $\sum_{I\in\mathcal{I}_d}\mathbb{P}_{b^e\sim\gamma^e}(\exists i \in N_e : b_i \in I) > \frac{n_e}{p}p = n_e$. We reach a contradiction since $\sum_{I\in\mathcal{I}^*}\mathbb{P}_{b^e\sim\gamma^e}(\exists i \in N_e : b_i \in I) \leq n_e$ for each set $\mathcal{I}^*$ of disjoint intervals.

We conclude the proof showing that the algorithm runs in polynomial time in n_e, $1/p$, and $log(1/\eta)$. Recall the tree representation described above: the number of recursive calls of Algorithm 6.1 is equal to the number of nodes which are at most twice the number of leaf nodes in a binary tree. Therefore the recursive calls of the algorithm are at most $2k^* \leq \frac{4n_e}{p}\log\left(\frac{1}{\eta}\right)$. □

Let $I^{p,\eta} \coloneqq \{(\alpha_j, \beta_j]\}_{j\in[k^*]}$ be the set of intervals returned by algorithm $\textsc{Rec}((0,1], p, \eta)$. The next step is to show that, for η small enough,

$$\mathbb{P}_{b^e\sim\gamma^e}\{\exists i \in N_e : b_i \in (\alpha_j, \beta_j)\} \leq p,$$

for every interval $(\alpha_j, \beta_j]$. This holds by definition for each interval $(\alpha_j, \beta_j]$ with $\beta_j - \alpha_j > \eta$. Thus, let us consider the intervals $(\alpha_j, \beta_j]$ such that $\beta_j - \alpha_j \leq \eta$. Let M be the maximum number of bits needed to represent the bids in the support of probability distribution γ^e. By setting $\eta = 2^{-M}$, we have that all the bids b_i in the interval $(\alpha_j, \beta_j]$ are equal to β_j.[6] Hence, it holds $\mathbb{P}_{b^e\sim\gamma^e}\{\exists i \in N_e : b_i \in (\alpha_j, \beta_j)\} = 0$.

[6]Our algorithm runs in time logarithmic in $\frac{1}{\eta}$ and hence polynomial in the size of the binary representation of bids in b^e.

Letting $B_p^c \coloneqq \bigcup_{(\alpha,\beta]\in I^{p,\eta}} \bigcup_{i\in N_c} \{\alpha + \tau i\}$ be a suitable set of discretized bidding strategies for $\tau > 0$ and $\eta = 2^{-M}$, we show that we can restrict the attention to bid profiles in B_p^c with a small loss in utility and by relaxing IR constraints.[7]

First, we provide the following auxiliary result.

Lemma 6.5. *Given $p \in [0,1]$, for any bidding strategy $b^c \in B^c$, there exists a discretized bidding strategy $\hat{b}^c \in B_p^c$:*

- $\tilde{\pi}_i(\hat{b}^c) \leq \tilde{\pi}_i(b^c)$ *for all* $i \in N_c$*; and*
- $\tilde{r}_i(\hat{b}^c) \geq \tilde{r}_i(b^c) - p$ *for all* $i \in N_c$*.*

Proof. Recall that $I^{p,\eta} \coloneqq \{(\alpha_j, \beta_j]\}_{j\in[k^*]}$ is the set of intervals returned by $\textsc{Rec}((0,1], p, \eta)$. Take a bid profile $b^c \in B^c$. Consider the bid profile $\hat{b}^c \in B^{c,p}$ such that, for each $i \in N_c$, $\hat{b}_i^c$ is the largest element in $\bigcup_{(\alpha,\beta]\in I^{p,\eta}} \{\alpha\}$ such that $\hat{b}_i^c \leq b_i$. Then, we increase each bid $\hat{b}_i^c$ by $\tau(n_c - i)$, for all $i \in [n_c]$ in order to have the same ordering of the colluders' bids in b^c and $\hat{b}^c$. This step is equivalent to introducing a specific tie breaking rule. Now, we show that the two conditions stated in the lemma hold for $\hat{b}^c$. First, it is easy to see that since each colluder decreases his bid (ignoring the arbitrary small τ), the payment decreases both in VCG and GSP auctions. Then, we show that the revenue of each colluder $i \in N_c$ decreases by a small amount. In particular, since with probability at least $1-p$ there is no external bid in interval $(\hat{b}^c, b^c)$ and the partial ordering of the colluders' bids does not change, then, with probability at least $1 - p$, colluder i is assigned to the same slot. Hence, their utility is at least $\tilde{r}_i(b^c) - p$. This concludes the proof. □

Then, by exploiting Lemma 6.5 we can prove the following Lemma 6.6. Intuitively, the lemma shows that, given a probability $p \in [0,1]$ and an optimal discretized bidding strategy, one can find an approximate solution to Problem (6.1) in polynomial time.

Lemma 6.6. *Given $p \in [0,1]$ and an optimal discretized bidding strategy $\hat{b}^c \in \operatorname{argmax}_{b^c\in B_p^c} \sum_{i\in N_c} \tilde{r}_i(b^c) - \tilde{\pi}_i(b^c)$, we can find in polynomial time a p-IR (see Equation* (6.2)*) and AIR solution to Problem* (6.1) *with value at least $OPT - pn_c$, where OPT is the optimal value of Problem* (6.1)*.*

[7] In the following, we ignore the loss in cumulative expected utility that results from the introduction of $\tau > 0$. Notice that this parameter is only necessary to induce specific tie-breaking rules and our results can be easily extended to consider the loss in utility due to τ. Moreover, τ can be taken to be exponentially small in the size of the problem instance, and hence negligible.

Proof. Let $b^{c,*} \in \operatorname{argmax}_{b^c \in B^c} \sum_{i \in N_c} \tilde{r}_i(b^c) - \tilde{\pi}(b^c)$. By Lemma 6.5, there exists a discretized bid profile $\tilde{b}^c \in B^{c,p}$ such that:

- $\tilde{\pi}_i(\tilde{b}^c) \leq \tilde{\pi}_i(b^{c,*})$ for all $i \in N_c$; and
- $\tilde{r}_i(\tilde{b}^c) \geq \tilde{r}_i(b^{c,*}) - p$ for all $i \in N_c$.

Hence, $\sum_{i \in N_c} \tilde{r}_i(\hat{b}^c) - \tilde{\pi}_i(\hat{b}^c) \geq \sum_{i \in N_c} \tilde{r}_i(\tilde{b}^c) - \tilde{\pi}_i(\tilde{b}^c) \geq \sum_{i \in N_c} \tilde{r}_i(b^{c,*}) - \tilde{\pi}_i(b^{c,*}) - n_c p$. We consider the solution to Problem (6.1) composed of bidding strategy $\hat{b}^c$, with and $q_i = \tilde{r}_i(\hat{b}^c) - t_i + p$. The solution has objective at least $OPT - n_c p$ and satisfies p-IR by construction. To conclude the proof, we show that the solution is AIR. In particular, it holds $\sum_{i \in N_c}(q_i - \tilde{\pi}_i(\hat{b}^c)) = \sum_{i \in N_c}\Big(\tilde{r}_i(\hat{b}^c) - \tilde{\pi}_i(\hat{b}^c) - t_i\Big) + p n_c \geq \sum_{i \in N_c}(\tilde{r}_i(b^{c,*}) - \tilde{\pi}_i(b^{c,*}) - t_i) \geq 0$, where the last inequality follows from Assumption 6.1. □

By Theorem 6.2, for any $p \in [0, 1]$, it is possible to find an optimal discretized bidding strategy $\hat{b}^c$ in time polynomial in the instance size and in $\frac{1}{p}$, since, as it is easy to check, the number of possible discretized bids in B^c_p is polynomial in $\frac{1}{p}$. Moreover, by employing Lemma 6.6, we can use the bidding strategy $\hat{b}^c$ to find an approximated solution to Problem (6.1) in polynomial time. Hence, given any $\varepsilon > 0$, it is sufficient to choose $p \in [0, 1]$ so that $\frac{1}{p} \in \text{poly}(\frac{1}{\varepsilon}, n)$ in order to obtain an ε-IR (see Equation (6.2)) and AIR approximate solution to Problem (6.1), as stated by the following theorem.

Theorem 6.3. *Given $\varepsilon > 0$, there exists an algorithm that runs in time polynomial in the instance size and $\frac{1}{\varepsilon}$, which returns an ε-IR and AIR solution to Problem* (6.1) *with value at least $OPT - \varepsilon$, where OPT is the optimal value of Problem* (6.1)*.*

6.5 Transfers with Limited Liability Setting

In this section, we provide an approximate solution to the media agency problem with limited liability constraints. In particular, similarly to the arbitrary transfers setting, we design a bi-criteria additive FPTAS that returns solutions providing an arbitrary small loss $\varepsilon > 0$ with respect to the optimal value of the problem, by relaxing the IR constraints by ε.

As a first step, we show that we can restrict Problem (6.1) with LL constraints to work with the set B^c_p by only incurring in a small loss in the objective function value and IR constraints satisfaction. Notice that, Problem (6.1) with LL constraints restricted to discretized bids does not

only have a smaller optimal value than Problem (6.1) with LL constraints, but it can also result in an infeasible problem, since Assumption 6.1 is *not* necessarily satisfied for a discretized bidding strategies. However, we can prove that, given a probability $p \in [0, 1]$, the following LP (6.9) that uses only bids in B_p^c and relaxes the IR constraints of quantity p is feasible and has value at least $OPT - pn_c$, where OPT is the optimal value of Problem (6.1) with LL constraints.

$$\max_{q \geq 0, \gamma \in \Delta_{B_p^c}} \sum_{b^c \in B_p^c} \gamma_{b^c} \sum_{i \in N_c} \tilde{r}_i(b^c) - \tilde{\pi}_i(b^c) \quad \text{s.t.} \tag{6.9a}$$

$$\sum_{b^c \in B_p^c} \gamma_{b^c} \tilde{r}_i(b^c) - q_i \geq t_i - p \qquad \forall i \in N_c \tag{6.9b}$$

$$\sum_{i \in N_c} q_i \geq \sum_{b^c \in B_p^c} \gamma_{b^c} \sum_{i \in N_c} \tilde{\pi}_i(b^c). \tag{6.9c}$$

Formally, we prove the following:

Lemma 6.7. *LP* (6.9) *is feasible. Moreover, the optimal value of LP* (6.9) *is at least $OPT - pn_c$, where OPT is the optimal value of Problem* (6.1) *with LL constraints.*

Proof. We show how to construct a solution that satisfies the two constraints. Take the optimal solution γ, q to LP (6.1) with LL. Let $S : B^c \to B^{c,p}$ be the function that maps each bid profile $b \in B^c$ to the bid profile $b' \in B^{c,p}$ that satisfies Lemma 6.5. We build a solution $\bar{\gamma}$, $\bar{q}$ to LP (6.9) such that $\bar{\gamma}_{b^c} := \mathbb{P}_{\tilde{b}^c \sim \gamma}(b^c = S(\tilde{b}^c))$ for each $b^c \in B^{c,p}$. Moreover, we take $\bar{q}_i = q_i$ for each $i \in N_c$. By lemma 6.5, we have that for each $b^c \in B^c$ and $i \in N_c$, $\tilde{r}_i(S(b^c)) \geq r_i(b^c) - p$ and $\tilde{\pi}_i(S(b^c)) \leq \pi_i(b^c)$. Hence,

$$\sum_{b^c \in B^{c,p}} \bar{\gamma}_{b^c} \sum_{i \in N_c} \tilde{\pi}_i(b^c) \leq \mathbb{E}_{b^c \sim \gamma}[\sum_{i \in N_c} \tilde{\pi}_i(b^c)]$$

and

$$\sum_{b^c \in B^{c,p}} \bar{\gamma}_{b^c} \sum_{i \in N_c} \tilde{r}_i(b^c) \geq \mathbb{E}_{b^c \sim \gamma}[\sum_{i \in N_c} \tilde{r}_i(b^c)] - p.$$

Thus, the objective decreases of at most pn_c while $\sum_{b^c \in B^{c,p}} \bar{\gamma}_{b^c} \tilde{r}_i(b^c) - \bar{q}_i \geq E_{b^c \sim \gamma}[\tilde{r}_i(b^c)] - p - q_i \geq t_i - p$ for each $i \in N_c$ and Constraints (6.9b) are satisfied. Finally, $\sum_{i \in N_c} \bar{q}_i = \sum_{i \in N_c} q_i \geq \mathbb{E}_{b^c \sim \gamma}[\sum_{i \in N_c} \tilde{\pi}_i(b^c)] \geq \sum_{b^c \in B^{c,p}} \bar{\gamma}_{b^c} \sum_{i \in N_c} \tilde{\pi}_i(b^c)$. This concludes the proof. □

Next, we provide an algorithm to solve LP (6.9) by using the ellipsoid method. To do that, we use the dual LP (6.10), in which variables $y =$

$\{y_1, \ldots, y_{n_c}\}$, x, and z are related to Constraints (6.9b), (6.9c), and $\gamma \in \Delta_{B_p^c}$, respectively.

$$\min_{y \leq 0, x, z} \sum_{i \in N_c} (t_i - p) y_i + z \quad \text{s.t.} \tag{6.10a}$$

$$\sum_{i \in N_c} y_i \tilde{r}_i(b^c) - x \sum_{i \in N_c} \tilde{\pi}_i(b^c) + z \geq \sum_{i \in N_c} \tilde{r}_i(b^c) - \tilde{\pi}_i(b^c) \qquad \forall b^c \in B_p^c \tag{6.10b}$$

$$-y_i + x \geq 0 \qquad \forall i \in N_c. \tag{6.10c}$$

By Lemma 6.7, the primal LP (6.9) is feasible (and bounded), and, thus, it holds strong duality. As a consequence, in order to provide a polynomial-time algorithm to solve LP (6.9), it is enough to apply the ellipsoid method to the dual LP (6.10), which can be done in polynomial time since the latter has polynomially-many variables and exponentially-many constraints. This is possible by providing a polynomial-time separation oracle that, given an assignment of values to the variables as input, returns a violated constraint (if any). Since there are only polynomially-many Constraints (6.10c), we can check if one of them is violated in polynomial time. Moreover, in order to find whether there exists a violated Constraint (6.10b), it is sufficient to solve the weighted utility problem in Equation (6.4) by setting $\hat{y}_i = (1 - y_i)$ for each $i \in N_c$ and $\hat{x} = x - 1$. By Theorem 6.2, this can be done in polynomial time by computing a shortest path of a suitable graph. Hence, we can prove the following theorem.

Theorem 6.4. *Given $\varepsilon > 0$, there exists an algorithm that runs in time polynomial in the instance size and $\frac{1}{\varepsilon}$ and returns an ε-IR (see Equation* (6.2)*) and AIR solution to Problem* (6.1) *with LL constraints having value at least $OPT - \varepsilon$, where OPT is the optimal value of Problem* (6.1) *with LL constraints.*

Proof. By Lemma 6.7, to provide the desired guarantees it is enough to provide a solution to LP (6.9) with $p = \epsilon / n_c$. By strong duality, it is sufficient to solve the dual LP (6.10). The algorithm employs the ellipsoid method to solve the dual. To do so, it needs a polynomial-time separation oracle that given an assignment (y, x, z) to the variables returns a violated constraint (if any). Since there are only polynomially-many Constraints (6.10c), we can check if one of these constraints is violated in polynomial time. Moreover, we can find in polynomial time if the exists a violated Constraint (6.10b) solving $\max_{b^c \in B^{c,p}} \sum_{i \in N_c} [(1 - y_i)\tilde{r}_i(b^c) + (x - 1)\tilde{\pi}_i(b^c)]$.

This is an instance of the weighted utility problem in Equation (6.4) with $\hat{y}_i = 1 - y_i$ for each $i \in N_c$ and $\hat{x} = x - 1$. If the value is higher than z, we return the constraint relative to the solution of Equation (6.4). Otherwise, all the constraints (6.10b) are satisfied. Finally, (6.4) can be solved in polynomial time by Theorem 6.2. This concludes the proof. □

CHAPTER 7

Algorithmic Advertising in the Metaverse: Finding Effective Ads Allocations

Section 1.2 introduces the topic and main challenges we address in this chapter. First, we provide an advertising model for the Metaverse setting in Section 7.1. In our model, users traverse several scenes during which they could be targeted with multiple ads of different formats, whose performance depends on the specific scene in which they are displayed and the externalities they are subject to. In particular, displaying an ad in a scene may raise negative forward externalities to other ads dis- played in future scenes. Differently from classical advertising setting, a user moves through sequence of scenes starting from the root of a tree of scenes. Furthermore, users could observe the same ad multiple times during their experience. Sections 7.2-7.3-7.4-7.5 analyze variations of this model. In particular, before studying the general problem characterized by extenalities and scene-dependednt ads, we study intermediate scenarios in which there are no externalities among ads or their qualities do not depend on the scene. In these scenarios, we assess the computational complexity of finding an optimal ad allocation and provide approximation algorithms with tight theoretical guarantees. We also discuss under which conditions our algorithms are monotone in the sense

of Myerson, thus leading to truthful auction mechanisms. In Section 7.6, we discuss the features of our user model. We compare our approximation algorithms with algorithms disregarding such features and show that the latters may be arbitrarily inefficient with respect to ours, for some instances.

7.1 Model

We introduce the following advertising model for the Metaverse.

Scenes Tree and User Transitions We assume that a user moves through a sequence of possible scenes, starting from the root of a tree of scenes and following one of the potential paths according to a probability distribution over the successors of every scene. Formally, $T = (S, \rho)$ is a *tree of scenes*, where S is the *set of scenes* in which a user can be, $s \in S$ is a scene, and $\rho : S \to \mathcal{P}(S)$ is the *successor function* taking as input a scene $s \in S$ and returning the subset $\rho(s)$ of S composed of all the scenes that are immediate successors of s in the tree; $\mathcal{P}(S)$ denotes the powerset of S. We say that the scenes s such that $\rho(s) = \emptyset$ are *terminal*. We denote with $\pi_{s,s'} \in [0, 1]$, where $s \in S, s' \in \rho(s)$, the *transition probability* that a user in scene s moves to immediate successor scene s'. Furthermore, for every non-terminal scene $s \in S$, it holds $\sum_{s' \in \rho(s)} \pi_{s,s'} = 1$. Notice that, we can model that a user leaves the metaverse with a non-null probability from scene s by using a successor of s that is terminal (this corresponds to stopping to observe the slots in the case of search advertising). We denote with $\boldsymbol{\sigma}$ a generic *ordered sequence* of scenes such that σ_i is the i-th scene of $\boldsymbol{\sigma}$. In particular, we denote with $\boldsymbol{\sigma}^s$ the sequence of scenes from the root node to scene $s \in S$, with $|\boldsymbol{\sigma}^s|$ the length of $\boldsymbol{\sigma}^s$, and with σ^s_i the i-th element of $\boldsymbol{\sigma}^s$, where $i \in [\,|\boldsymbol{\sigma}^s|\,]$.[1] Hence, for every $s \in S$, the root scene corresponds to σ^s_1 and scene s to $\sigma_{|\boldsymbol{\sigma}^s|}$. The *reach probability* of s is $\Pi^s = \prod_{i=1}^{|\boldsymbol{\sigma}^s|-1} \pi_{\sigma^s_i, \sigma^s_{i+1}}$, stating the probability a user reaches s starting from the root σ^s_1.

Ads, Qualities, and Externalities We denote with A the *set of ads* and with $a \in A$ an ad. For simplicity, we assume that at most one ad can be displayed in every scene. In particular, we denote with $x : S \to A \cup \{a_\emptyset\}$ the *allocation function* taking as input scene $s \in S$ and returning ad $a \in A$ or $a_\emptyset$ allocated to scene s, where ad $a_\emptyset$ is fictitious, meaning that no ad is allocated in that scene. Every ad $a \in A$ allocated in scene $s \in S$ is characterized by a quality $q_{a,s} \in [0, 1]$, that is the user's conversion probability conditioned to the fact

[1] We denote with $[n]$ the set $\{1, \ldots, n\}$, where $n \in \mathbb{N}$.

that scene s has been reached by the user and the fact that no other ad has been displayed before s. For the sake of presentation, whenever we focus on settings in which the quality is scene-independent, we use q_a in place of $q_{a,s}$. By convention, $q_{a_\emptyset,s} = 0$ for every $s \in S$. Furthermore, ads are subject to *forward externalities*, such that the display of ad a allocated in scene s affects the quality of ad a' allocated in scene s' when s precedes s' in the tree. Formally, we model such an externality with $\gamma_{a,a'} \in [0,1]$, where $a, a' \in A$ and a is allocated in a scene preceding (immediately or not) in the tree the scene where a' is allocated. We assume that $\gamma_{a,a'} \leq 1$ for every $a \neq a' \in A$, while $\gamma_{a,a} = 1$ for every $a \in A$. Notice that, when $\gamma_{a,a'} < 1$, the externality is *negative*, meaning that the display of a before a' negatively affects the quality of a', while, when $\gamma_{a,a'} = 1$, the externality is *neutral*, meaning that the display of a before a' does not affect the quality of a'. By convention, leaving a scene without any ad allocated does not introduce any externality and therefore $\gamma_{a_\emptyset,a'} = 1$ for every $a' \in A$. Whenever we refer to no-externalities settings, we assume that $\gamma_{a,a'} = 1$ for every $a, a' \in A$. Furthermore, we assume that the user may forget the ads seen in the past. More precisely, we assume that the user's behavior only depends on the ads seen in the previous $k \in \mathbb{N}$ scenes (where $k = 0$ means that the user forgets every ad previously seen). The *total externality* to which ad a in scene s is subject to is given by $\Gamma(x,s) = \prod_{i=\max\{1,|\boldsymbol{\sigma}^s|-k\}}^{|\boldsymbol{\sigma}^s|-1} \gamma_{x(\sigma_i^s),x(s)}$ and is due to all the ads displayed in the k scenes preceding s in the sequence $\boldsymbol{\sigma}^s$ (whose number is $\min\{k, |\boldsymbol{\sigma}^s| - 1\}$). Notice that when $k = \infty$, the user perfectly recalls all the ads seen. Thus, the probability that a user converts on an ad a in scene s conditioned to the reach of scene s is $\Gamma(x,s)\, q_{s,a}$. This holds whenever ad a is not displayed in scenes preceding s, while we treat separately the case in which an ad is displayed multiple times along the same path.

Customarily in the literature on search and social advertising, an ad can be displayed only once in an allocation, see, *e.g.*, (Kempe and Mahdian, 2008). The rationale is that the users click on an ad only the first time this ad is displayed. From our understanding, this model does not exactly capture the actual behavior of the user, particularly in the metaverse. Indeed, in practice, if users observe an ad and convert, then it is likely that they will never convert again when observing the same ad in the future as the users either will not repeat the conversion or exploit channels different from the advertising to repeat it. Instead, if users observe an ad and do not convert to it, they could convert in the future. This assumption is closer to the classical advertising funnel, in which users observe multiple times the same ads

Allocation problem	Complexity	Best apx. ratio	Best monotone apx. ratio
META-SI-NE	Poly	1	1
META-SD-NE	APX-Complete	$(1-1/e)$	—
META-SI-E	Poly-APX-Complete	$1/(k+1)$	$1/(k+1)$
META-SD-E	Poly-APX-Complete	$(1-1/e)/(k+1)$	—

Table 7.1: *Summary of the computational complexity results. 'SI' = 'scene independent quality', 'SD' = 'scene dependent quality', 'NE' = 'no externalities', 'E' = 'externalities'. Monotonicity refers to Myerson's weak monotonicity.*

before converting. Formally, we assume that, if a user converts on ad a in scene s, then she will never convert again on a when displayed in a scene s' following s, while, if a user does not convert on ad a in scene s, then she can convert on the same a in a scene s' following s. This assumption requires adjusting the quality of an ad when displayed multiple times along a single path. In particular, we denote with $H(x,s) \subseteq S$ the subset of scenes s' along sequence $\boldsymbol{\sigma}^s$ in which ad $a = x(s) = x(s')$ is allocated, excluded scene s. We define $\Xi(x,s) = \prod_{s' \in H(x,s)} (1 - \Gamma(x,s')\, q_{x(s'),s'})$ as the probability that the user never converts on ad $a = x(s)$ when allocated in scenes s' strictly before scene s conditioned to the reach of s'. Finally, we denote with $\tilde{q}(x,s) = \Gamma(x,s)\, q_{x(s),s}\, \Xi(x,s)$ the *adjusted quality* of the ad allocated in s given the ads allocated in the preceding scenes. Thus, the *conversion rate* of the ad a allocated in scene s is $\Pi^s\, \tilde{q}(x,s)$.

We denote with $\theta_a \in [0,1]$ the *value per conversion* of ad a. Its expected value (w.r.t. the reach of s and conversion of s) is $\Pi^s\, \tilde{q}(x,s)\, \theta_{x(s)}$. Finally, the *allocation expected value* of x is $\sum_{s \in S} \left(\Pi^s\, \tilde{q}(x,s)\, \theta_{x(s)}\right)$.

In our work, we also study the model under some simplifications. We call META the allocation problem, and we use the suffixes SI and SD for the cases in which the qualities are scene-independent and scene-dependent respectively, and the suffixes NE and E for the cases in which there are no externalities and there are externalities respectively. Table 7.1 summarizes our results in these settings.

For the sake of clarity, we provide an explanatory example of our model.

Example 7.1. *Figure 7.1 shows a setting described by a tree where the set of scenes is $S = \{s_1, \ldots, s_8\}$ and the set of ads is $A = \{a_1, a_2, a_3\} \cup a_\emptyset$. The quality of the ads is $q_{a,s} = 0.1$ for all $a \in A$ and $s \in S$, the externalities are $\gamma_{a_1,a_2} = \gamma_{a_1,a_3} = \gamma_{a_1,a_3} = 0.8$, and the values per conversion are $\theta_{a_1} = 0.5$, $\theta_{a_2} = 0.6$ and $\theta_{a_3} = 0.7$. The transition prob-*

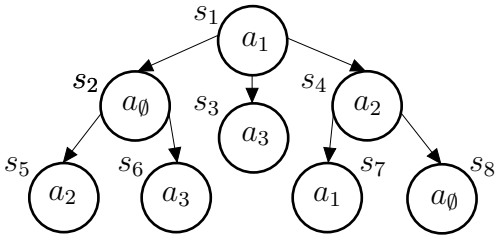


Figure 7.1: *Example of our metaverse advertising model.*

abilities are $\pi_{s_1,s_2} = \pi_{s_2,s_6} = \pi_{s_4,s_7} = 0.7$, $\pi_{s_1,s_3} = 0.1$, $\pi_{s_1,s_4} = 0.2$, $\pi_{s_2,s_5} = \pi_{s_4,s_8} = 0.3$. *Moreover, we set* $k \geq 2$. *Consider, for instance, scene* s_7*: the total externality is* $\Gamma(x, s_7) = \gamma_{a_1,a_1}\,\gamma_{a_1,a_2} = 0.8$, *the adjusted quality is* $\tilde{q}(x, s_7) = \Gamma(x, s_7)\,q_{a_1,s_7}\,\Xi(x, s_7) = 0.072$, *the expected value is* $\Pi^{s_7}\,\tilde{q}(a_1, s_7)\,\theta_{a_1} = 0.00504$. *The allocation expected value is* $\sum_{s\in S}\left(\Pi^{s}\,\tilde{q}(x, s)\,\theta_{x(s)}\right) = 0.10776$. *Notice that if* $k = 1$, *the allocation expected value increases to* 0.11714 *as the negative effects of the externalities are mitigated further.*

Myerson's Weakly Monotonicity When designing allocation algorithms in the following sections, we investigate whether they satisfy Myerson's weakly monotonicity property. Indeed, since our metaverse advertising model is a single-parameter (*i.e.*, θ_a) linear environment, Myerson's weakly monotonicity is *necessary* and *sufficient* for the design of a truthful mechanism in dominant strategies Myerson (1981). In our case, the property can be defined as follows.

Definition 7.1. *In the metaverse single-parameter environment, an allocation mechanism* $\mathcal{M}$ *that maps a type profile* $(\theta_a)_{a\in A}$ *to an allocation* x *is weakly monotone if for every ad* $\hat{a}$ *and types* $\theta_{a'}$ *of the other ads* $a' \in A\setminus\{\hat{a}\}$, *the allocation mechanism* $\mathcal{M}$ *is such that the term* $\sum_{s\in S: x^{\theta_{\hat{a}}}(s)=\hat{a}}\left(\Pi^{s}\,\tilde{q}(x^{\theta_{\hat{a}}}, s)\right)$ *is non-decreasing in* $\theta_{\hat{a}}$, *where* $x^{\hat{\theta}_{\hat{a}}} = \mathcal{M}((\theta_a)_{a\in A})$ *is the allocation returned by the mechanism with type profile* $(\theta_a)_{a\in A}$.

7.2 Poly-time Algorithm for META-SI-NE

We focus on the basic META-SI-NE case in which there are no externalities and the quality of the ads does not depend on the scene. This case differs from the allocation problem in classical ad auctions for two reasons: the allocation may be on a tree instead of a line, and an ad can be displayed

multiple times along a single path of the tree. Interestingly, we show that we can design a polynomial-time greedy algorithm facing this setting and that this algorithm plays a central role when solving the more general settings. The pseudocode is reported in Algorithm 7.1. It works iteratively. We denote with R the subset of scenes that at every iteration are not assigned any ad. Furthermore, at every iteration, the algorithm chooses a scene-ad pair $(s^*, a^*) \in S \times A$ which maximizes the expected value of allocating an ad in an available scene. We define the following tie-breaking rule to identify the unique pair chosen at each iteration (Line 3) among all the possible value-maximizing pairs.

Definition 7.2 (Tie-Breaking Rule)**.** *Be $\bar{P}$ the set of pairs $(\bar{s}, \bar{a})$ returned by $\operatorname{argmax}_{s\in S, a\in A} \Pi^s \tilde{q}(x, s)\, \theta_a$. Whenever $\bar{P}$ is not a singleton, break ties assigning to (s^*, a^*) any pair $(\bar{s}', \bar{a}')$ such that $|\boldsymbol{\sigma}^{\bar{s}'}|$ is the minimum among all $|\boldsymbol{\sigma}^{\bar{s}}|$ where $(\bar{s}, \bar{a}) \in \bar{P}$ for some $\bar{a} \in A$.*

Once the value-maximizing pair (s^*, a^*) has been found, Algorithm 7.1 allocates ad a^* to scene s^* (Line 4). Then, scene s is removed from the set R which contains the available scenes (Line 5). The algorithm iterates until every scene has been filled with one ad. Finally, it returns the allocation function $x(\cdot)$.

Algorithm 7.1 GREEDY

Inputs: set of scenes S, set of ads A

1: Initialize $R \leftarrow S$, $x(s) \leftarrow a_\emptyset \; \forall s \in S$
2: **while** $R \neq \emptyset$ **do**
3: $\quad (s^*, a^*) \leftarrow \operatorname{argmax}_{s\in R, a\in A} \Pi^s \tilde{q}(x, s)\, \theta_a$
$\quad\quad$ ▷ Ties are broken according to Definition 7.2
4: $\quad x(s^*) \leftarrow a^*$
5: $\quad R \leftarrow R \backslash s^*$
6: **end while**
7: **return** $x(\cdot)$

The following theorem shows that Algorithm 7.1 returns an optimal allocation.

Theorem 7.1. *Algorithm 7.1 computes an optimal solution to the* `META-SI-NE` *problem.*

Proof. As a first step, we show that the expected value of an allocation x can be decomposed into a component for each possible path. Formally, we show that

$$\sum_{s\in S} \left(\Pi^s \tilde{q}(x, s)\, \theta_{x(s)}\right) = \sum_{s\in S: \rho(s)=\emptyset} \Pi^s V_s(x)$$

where $V_s(x) = \sum_{s' \in \boldsymbol{\sigma}^s} \tilde{q}_{x,s}\theta_{x(s)}$. To see that it is sufficient to observe that given an s, it holds

$$\begin{aligned}\Pi^s \, \tilde{q}(x,s) \, \theta_{x(s)} &= \left(\sum_{s' : s \in \boldsymbol{\sigma}^{s'}, \rho(s) = \emptyset} \Pi^{s'} \right) \tilde{q}_{x,s}\theta_{x(s)} \\ &= \sum_{s' : s \in \boldsymbol{\sigma}^{s'}, \rho(s) = \emptyset} \Pi^{s'} \tilde{q}_{x,s}\theta_{x(s)}\end{aligned}$$

and hence

$$\begin{aligned}\sum_{s \in S} \left(\Pi^s \, \tilde{q}(x,s) \, \theta_{x(s)} \right) &= \sum_{s \in S} \sum_{s' : s \in \boldsymbol{\sigma}^{s'}, \rho(s) = \emptyset} \Pi^{s'} \tilde{q}_{x,s}\theta_{x(s)} \\ &= \sum_{s \in S : \rho(s) = \emptyset} \Pi^s \sum_{s' \in \boldsymbol{\sigma}^s} \tilde{q}_{x,s}\theta_{x(s)} \\ &= \sum_{s \in S : \rho(s) = \emptyset} V_s\end{aligned}$$

Then, we observe that thanks to the tie breaking rule in Definition 7.2, the algorithm assigns ads to nodes from the top to the bottom of the tree. Suppose by contradiction that Algorithm 7.1 assigns an ad a to a node s^1 such there exists a node $s^2 \neq s^1$ in $\boldsymbol{\sigma}^{s^1}$ that is not assigned, *i.e.*, with $x(s^2) = a_\emptyset$. Then, we have that $\Pi^{s^2} \, \tilde{q}(x, s^2)\theta_a \geq \Pi^{s^1} \, \tilde{q}(x, s^1)\theta_a$ and by the tie breaking rule the ad is assigned to node s^2. Let x^* be the allocation returned by Algorithm 7.1. Moreover, given a node s, let x' be a different allocation with $x'(s') = x^*(s')$ for all s' that are predecessors of s. Then, the assignment rule in Line 3 of the algorithm implies that

$$\tilde{q}(x^*, s) \, \theta_{x^*(s)} \geq \tilde{q}(x', s) \, \theta_{x'(s)}, \tag{7.1}$$

where the inequalities follows since the value of assigning any ad to s does not change from the partial allocation x considered by the algorithm and the final allocation x^* (and x') since all the scenes that precede s have already been assigned. Let x^* be the allocation returned by Algorithm 7.1. We show that this allocation is optimal for each possible path. Formally, given a terminal node $\bar{s}$, *i.e.*, such that $\rho(\bar{s}) = \emptyset$, and an optimal allocation x_0 for the path that terminates in $\bar{s}$, *i.e.*, $x_0 \in \operatorname{argmax}_x V_{\bar{s}}(x)$, we show that $V_{\bar{s}}(x^*) \geq V_{\bar{s}}(x_0)$. This is sufficient to prove the theorem since it implies

$$\sum_{s \in S} \left(\Pi^s \, \tilde{q}(x^*, s) \, \theta_{x^*(s)} \right) = \sum_{s \in S : \rho(s) = \emptyset} \Pi^s V_s(x)$$

$$
\begin{aligned}
&\geq \sum_{s \in S : \rho(s) = \emptyset} \Pi^s \max_x V_s(x) \\
&\geq \max_x \sum_{s \in S : \rho(s) = \emptyset} \Pi^s V_s(x) \\
&= \max_x \sum_{s \in S} \left(\Pi^s \, \tilde{q}(x, s) \, \theta_{x(s)} \right)
\end{aligned}
$$

Given a terminal node $\bar{s}$, let x_0 be the optimal allocation for the path terminating in $\bar{s}$. We show how to modify iteratively x_0 into x^* without decreasing the value of the allocation. This directly implies that $V_{\bar{s}}(x^*) \geq V_{\bar{s}}(x_0)$ and the optimality of x^*. We iterate over all the $i \in \{1, \ldots, |\boldsymbol{\sigma}^{\bar{s}}|\}$ and for each i, we build an allocation x_i such that the expected value of x_i is at least the expected value of x_{i-1}. Moreover, the procedure guarantees that for each i it holds $x_i(\sigma_j^{\bar{s}}) = x^*(\sigma_j^{\bar{s}})$ for all $j \leq i$, implying $x_{|S|} = x^*$. The procedure works as follows. We iterate over all the i and given an i we consider three cases. Let S_i be the set of scene s in $\boldsymbol{\sigma}^{\bar{s}} \setminus \boldsymbol{\sigma}^{s_i}$ such that $x_{i-1}(s) = x^*(s_i)$.

Case 1. Suppose that $x^*(\sigma_i^{\bar{s}}) = x_{i-1}(\sigma_i^{\bar{s}})$. Then, setting $x_i = x_{i-1}$ we trivially satisfy the required conditions.

Case 2. Suppose $x^*(\sigma_i^{\bar{s}}) \neq x_{i-1}(\sigma_i^{\bar{s}})$ and $S_i = \emptyset$. Let $x_i(s_i) = x^*(s_i)$ while the allocation x_i is equivalent to x_{i-1} in all the other nodes. Then, the difference between the values of the allocations x_i and x_{i-1} is $\tilde{q}(x_i, \sigma_i^{\bar{s}})\theta_{x_i(\sigma_i^{\bar{s}})} - \tilde{q}(x_{i-1}, s')\theta_{x_{i-1}(s')}$, where s' is the last node in the path $\boldsymbol{\sigma}^{\bar{s}}$ with $x_{i-1}(s') = x_{i-1}(\sigma_i^{\bar{s}})$ (it may be $\sigma_i^{\bar{s}}$). Moreover,

$$
\begin{aligned}
\tilde{q}(x_i, \sigma_i^{\bar{s}})\theta_{x_i(\sigma_i^{\bar{s}})} &= \tilde{q}(x^*, \sigma_i^{\bar{s}}) \, \theta_{x^*(\sigma_i^{\bar{s}})} \\
&\geq \tilde{q}(x_{i-1}, \sigma_i^{\bar{s}}) \, \theta_{x_{i-1}(\sigma_i^{\bar{s}})} \\
&\geq \tilde{q}(x_{i-1}, s')\theta_{x_{i-1}(s')},
\end{aligned}
$$

where the equality comes from the equivalence between x_i and x^* for all the scenes $\sigma_1^{\bar{s}}, \ldots, \sigma_i^{\bar{s}}$, the first inequality follows from Eq. (7.1) and the second inequality from the fact that the quality decreases when an ad is displayed more times. This proves that the expected value of the allocation x_i is at least the expected value of the allocation x_{i-1}.

Case 3. Suppose $x^*(\sigma_i^{\bar{s}}) \neq x_{i-1}(\sigma_i^{\bar{s}})$ and $S_i \neq \emptyset$. Let $x_i(\sigma_i^{\bar{s}}) = x^*(\sigma_i^{\bar{s}})$ and $x_i(s') = x^*(\sigma_i^{\bar{s}})$, where s' is an arbitrary scene in S_1. Moreover, let x_i be equivalent to x_{i-1} in all the other scenes. Then, every ad appears the same number of times in the path $\boldsymbol{\sigma}^{\bar{s}}$ in x_i and x_{i-1} and hence the expected value of the allocation does not change from x_{i-1} to x_i. This concludes the proof.

□

Notice that since the above greedy algorithm returns an optimal allocation, it can be used together with the Vickrey-Clarke-Groves mechanism Nisan et al. (2007) to obtain a truthful mechanism in dominant strategies which runs in polynomial time (trivially, Myerson's weak monotonicity is satisfied). Therefore, such a mechanism can scale up to real-world settings.

7.3 META-SD-NE: Dealing with Scene-dependent Qualities

In this section, we focus on the setting in which the ad qualities depend on the scene and there are no externalities. Initially, we show that the allocation problem in this setting is APX-Hard. Our reduction is based on the satisfiability problem 3-SAT-5 which is defined as follows.

Definition 7.3. *A 3-SAT-5 instance is a 3-*SAT *instance in which each variable appears in exactly 5 clauses.*

As shown by Feige (1998), the following theorem holds.

Theorem 7.2 (Feige (1998))**.** *For some constant* $0 < c < 1$*, it is* NP*-Hard to distinguish whether a 3-*SAT*-5 instance is satisfiable or there is no assignment satisfying a* c *fraction of the clauses.*

Now, we can prove the following.[2]

Theorem 7.3. *META-SD-NE is APX-Hard.*

Most interestingly, we can show that META-SD-NE is APX-Complete by designing a polynomial-time algorithm that works in a greedy fashion providing a constant approximation factor. To provide the algorithm, we need some preliminary steps that we introduce in the following.

Initially, we establish a relation between ad allocations and matroids. A matroid $M \coloneqq (G, \mathcal{I})$ is defined by a finite ground set G and a collection $\mathcal{I}$ of independent sets, *i.e.*, subsets of G satisfying some characterizing properties (see (Schrijver, 2003) for a detailed formal definition). We denote with $\mathcal{B}(M)$ the set of the *bases* of M, which are the maximal sets in $\mathcal{I}$. We show that feasible allocations can be represented by the matroid $M \coloneqq (G, \mathcal{I})$ such that:

- the ground set is $G \coloneqq \{(a, s) \mid a \in A \cup \{a_\emptyset\}, s \in S\}$, *i.e*, the set of all the possible assignments of ads to scenes;

[2]Some of the proofs of the chapter are deferred to Appendix B

- a subset $I \subseteq G$ belongs to $\mathcal{I}$ if and only if I contains *at most one* pair in $\{(a,s)\}_{a \in A \cup \{a_\emptyset\}}$ for each scene $s \in S$, *i.e.*, every scene is assigned to no more than one ad (while an ad can be allocated to multiple scenes).

Intuitively, an element (a,s) of the ground set G belongs to the independent set I if the ad a is allocated to scene s. However, sets $I \in \mathcal{I}$ do *not* characterize allocations, as they may not specify an ad for each scene. Indeed, allocations are captured by the basis set $\mathcal{B}(M)$ of the matroid M. Let us recall that $\mathcal{B}(M)$ contains all the maximal sets in $\mathcal{I}$, and, thus, a subset $I \subseteq \mathcal{I}$ belongs to $\mathcal{B}(M)$ if and only if I contains *exactly one* pair for each scene $s \in S$. Intuitively, a basis $I \in \mathcal{B}(M)$ defines an allocation in which, for each scene $s \in S$ it is allocated the ad a such that $(a,s) \in I$. This ad is unique by construction as discussed above.

Then, we define the utility function f on the subset of G as follows.

Definition 7.4. *Let $f : 2^G \to \mathbb{R}_+$ be the function such that, given a subset $D \in 2^G$, $f(D)$ denotes the welfare of assigning to a scene s the ad such that $(a,s) \in G$ without externalities.*[3] *Formally, we write:*

$$f(D) = \sum_{s \in S} \sum_{a \in A : (a,s) \in D} \Pi^s q_{a,s} \theta_a \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a,s') \in D} (1 - q_{a,s'}).$$

Function f satisfies a crucial property: it provides a *decreasing marginal return*. In particular, in the following, we show that the utility function $f : 2^G \to \mathbb{R}_+$ is *monotone submodular*. Formally, a function is monotone if for every pair of subsets D_1, D_2 such that $D_1 \subseteq D_2 \subseteq G$, the property $f(D_1) \leq f(D_2)$ holds. Moreover, we say that f is submodular if, for every pair of subsets D_1, D_2 such that $D_1 \subseteq D_2 \subseteq G$ and $(a,s) \in G$, the following property holds:

$$f(D_1 \cup \{(a,s)\}) - f(D_1) \geq f(D_2 \cup \{(a,s)\}) - f(D_2).$$

Now, we provide a characterization of the function $f(\cdot)$.

Lemma 7.1. *Given a subset $D \in 2^G$, $f(D)$ can be written as:*

$$f(D) = \sum_{s \in S : \rho(s) = \emptyset} \Pi^s \sum_{a \in A} \theta_a \, f_{s,a}(D),$$

where

$$f_{s,a}(D) = \sum_{s' \in \boldsymbol{\sigma}^s : (a,s') \in D} q_{a,s'} \prod_{s'' \in \boldsymbol{\sigma}^{s'} \setminus \{s'\} : (a,s'') \in D} (1 - q_{a,s''}).$$

[3] Notice that this defines a feasible allocation only if $D \in \mathcal{I}$.

Proof. We have that

$$\begin{aligned}
&\sum_{s\in S:\rho(s)=\emptyset} \Pi^s \sum_{a\in A} \theta_a f_{s,a}(D) = \\
&= \sum_{s\in S:\rho(s)=\emptyset} \Pi^s \sum_{a\in A} \theta_a \sum_{s'\in \boldsymbol{\sigma}^s:(a,s')\in D} q_{a,s'} \prod_{s''\in \boldsymbol{\sigma}^{s'}\setminus\{s'\}:(a,s'')\in D} (1-q_{a,s''}) = \\
&= \sum_{a\in A} \theta_a \sum_{s\in S:\rho(s)=\emptyset} \Pi^s \sum_{s'\in \boldsymbol{\sigma}^s:(a,s')\in D} q_{a,s'} \prod_{s''\in \boldsymbol{\sigma}^{s'}\setminus\{s'\}:(a,s'')\in D} (1-q_{a,s''}) = \\
&= \sum_{a\in A} \theta_a \sum_{s'\in S:(a,s')\in D} \left(\sum_{s\in S:\rho(s)=\emptyset\wedge s'\in\boldsymbol{\sigma}^s} \Pi^s \right) \cdot \\
&\cdot q_{a,s'} \prod_{s''\in \boldsymbol{\sigma}^{s'}\setminus\{s'\}:(a,s'')\in D} (1-q_{a,s''}) = \\
&= \sum_{a\in A} \theta_a \sum_{s'\in S:(a,s')\in D} \Pi^{s'} q_{a,s'} \prod_{s''\in \boldsymbol{\sigma}^{s'}\setminus\{s'\}:(a,s'')\in D} (1-q_{a,s''}) = \\
&= \sum_{s\in S} \sum_{a\in A:(a,s)\in D} \Pi^s \; q_{a,s} \; \theta_a \prod_{s'\in \boldsymbol{\sigma}^s\setminus\{s\}:(a,s')\in D} (1-q_{a,s'}) = \\
&= f(D)
\end{aligned}$$

This concludes the proof. □

Exploiting the above characterization, we can show that function $f(\cdot)$ is monotone submodular (see Appendix B).

Lemma 7.2. *Function $f(\cdot)$ is monotone submodular.*

Then, given the submodularity of function $f(\cdot)$, we can resort to the standard tools of submodular maximization to provide a polynomial-time algorithm to optimize f over $\mathcal{I}$. In particular, we can exploit the *continuous greedy algorithm* to provide a $(1-1/e)$-approximation (Calinescu et al., 2011). Then, to provide an approximation to the optimal ad allocation, it is sufficient to consider the equivalence between independent sets $I \in \mathcal{I}$ and ad allocations x.

Theorem 7.4. *META-SD-NE admits a polynomial-time algorithm that provides a $(1-1/e)$ approximation.*

Proof. The optimization problem of maximizing the function f over the set $\mathcal{I}$ is the maximization of a monotone submodular function over a matroid. Notice that f is monotone submodular by Lemma 7.2. Hence, applying the

continuous greedy algorithm to the problem provides a $1 - 1/e$ approximation in polynomial time (Calinescu et al., 2011). The proof is concluded by considering the equivalence between maximizing f over $\mathcal{I}$ and the ad allocation problem. In particular, an independent set I is equivalent to an ad allocation in which to each scene $s \in S$ is allocated the ad $a \in A$ such that $(a, s) \in I$ (if any). □

The impossibility of designing polynomial-time algorithms finding the optimal allocation for the `META-SD-NE` problem (unless $\mathsf{P} = \mathsf{NP}$) rules out the resort to the Vickrey-Clarke-Groves mechanism and poses the question whether we can design a truthful mechanism in dominant strategies running in polynomial time. Moreover, the analysis of the weak monotonicity of the continuous greedy approach is elusive. An intriguing idea is to use Algorithm 7.1. Indeed, to maximize monotone submodular functions we can use the simpler greedy approach instead of the more complex continuous greedy with a small loss in the approximation factor. In particular, the greedy algorithm provides a $\frac{1}{2}$-approximation to monotone submodular maximization on a matroid Nemhauser et al. (1978). Moreover, Algorithm 7.1 is weakly monotone for the `META-SI-NE` problem. However, as we show in the following proposition, such an algorithm is not weakly monotone, and thus it cannot be used to design a truthful mechanism.

Proposition 7.1. *Algorithm 7.1 is not weakly monotone (in the sense of Myerson) for* `META-SD-NE`.

Proof. To prove the statement, we provide an instance in which Algorithm 7.1 is *not* weakly monotone. Consider an instance with two ads a_1 and a_2 with value per conversion $\theta_{a_1} = \frac{1}{2} + \epsilon$ and $\theta_{a_2} = \frac{1}{2}$, where $\epsilon > 0$ is an arbitrary small value. There are three scenes s_1, s_2, and s_3 arranged in a line. In particular, s_1 is the root of the tree and $\pi_{s_1,s_2} = \pi_{s_2,s_3} = 1$. The qualities are $q_{a,s_1} = \frac{2}{3}$, $q_{a,s_2} = \frac{1}{2}$, and $q_{a,s_3} = \frac{1}{2}$ for each ad $a \in \{a_1, a_2\}$.

Consider the behavior of the greedy algorithm. As a first step, the greedy algorithm will assign to scene s_1 the ad a_1. Then, the algorithm must choose which ad to assign to scene s_2. Assigning ad a_1, the additional value is given by $q_{a_1,s_2}\theta_{a_1}(1 - q_{a_1,s_1}) = \frac{1}{2}(\frac{1}{2} + \epsilon)(1 - \frac{2}{3}) = \frac{1}{12} + \frac{1}{6}\epsilon$, while assigning ad a_2, the additional value of the allocation is $q_{a_2,s_2}\theta_{a_2} = \frac{1}{4}$. Hence, ad a_2 is assigned to scene s_2. Finally, the greedy algorithm assigns scene s_3 to ad a_2 since the additional value assigning ad a_2 is $q_{a_2,s_3}\theta_{a_2}(1 - q_{a_2,s_2}) = \frac{1}{8}$, while the additional value assigning ad a_1 is $q_{a_1,s_3}\theta_{a_1}(1 - q_{a_1,s_1}) = \frac{1}{12} + \frac{1}{6}\epsilon$. Hence, the utility of advertiser a_1 is $q_{a_1,s_1}\theta_{a_1} = \frac{2}{6} + \frac{2}{3}\epsilon = \frac{2}{3}\theta_{a_1}$, while the utility of advertiser a_2 is $q_{a_2,s_2}\theta_{a_2} + q_{a_2,s_3}\theta_{a_2}(1 - q_{a_2,s_2}) = \frac{1}{4} + \frac{1}{8} = \frac{3}{8}$. A

similar argument shows that if $\theta_{a_1} = \frac{1}{2} - \epsilon$, ad a_1 is assigned to scenes s_2 and s_3. Thus, the utility of advertiser a_1 is $q_{s_2,a_1}\theta_{a_1} + q_{s_3,a_1}\theta_{a_1}(1 - q_{s_2,a_1}) = \frac{3}{4}(\frac{1}{2} - \epsilon) = \frac{3}{4}\theta_{a_1}$. Since $\frac{3}{4} > \frac{2}{3}$, this shows that the mechanism is not monotone. □

7.4 META-SI-E: Dealing with Externalities

In this section, we focus on the META-SI-E problem in which there are externalities among the ads but the ad qualities are scene-independent. We start our analysis by providing a strong negative result. We show that the allocation problem is hard to approximate and that the hardness of approximation depends on the memory length k. Our reduction is from the following promise problem related to the problem of finding cliques in graphs.

Theorem 7.5 (Håstad (1999), Zuckerman (2007))**.** *For every $\epsilon > 0$, it is NP-Hard to distinguish whether a graph $G = (V, E)$ with vertexes V and edges E has a clique of size $|V|^{1-\epsilon}$ or all the cliques have a size of at most $|V|^{\epsilon}$.*

We can show that it is NP-Hard to provide an approximation to META-SI-E sublinear in the memory length k. Formally, we can state the following:

Theorem 7.6. *For any $\epsilon > 0$, it is NP-Hard to approximate META-SI-E to within a factor $|k + 1|^{1-\epsilon}$, where k is the memory length.*

Most interestingly, we can show that META-SI-E admits a polynomial-time approximation algorithm that provides a $\frac{1}{k+1}$-approximation, thus matching the lower bound stated above. The pseudocode is provided in Algorithm 7.2. It extends the greedy algorithm described in Section 7.2 as follows.

Algorithm 7.2 allocates ads only to scenes at depth $\{1+i(k+1)\}_{i\in\mathbb{N}}$, *i.e.*, it allocates ads only to the scenes $s \in S$ such that $|\boldsymbol{\sigma}^s| \in \{1 + i(k + 1)\}_{i\in\mathbb{N}}$. In this way, the allocated ads are not subject to any externality. Moreover, as we show in the following theorem, we allocate ads to a sufficient subset of scenes to guarantee a $1/(k + 1)$-approximation of the optimal utility. Then, Algorithm 7.2 computes the optimal allocation resorting to the greedy Algorithm 7.1. The following theorem formally states the guarantees of the algorithm.

Theorem 7.7. *Algorithm 7.2 provides a $\frac{1}{k+1}$-approximation to META-SI-E. Moreover, it runs in polynomial time.*

Proof. It is easy to see that the algorithm runs in polynomial time. In the following we prove the approximation guarantees of the algorithm. Given a $i \in \{1, \ldots, k+1\}$, let $S_i = \{s \in S : |\boldsymbol{\sigma}^s| \in \{1 + j(k+1)\}_{j \in \mathbb{N}}\}$. As a first step, we show that $\max_{x(\cdot)} \sum_{s \in S_1} \left(\Pi^s \, \tilde{q}(x, s) \, \theta_{x(s)}\right) \geq \max_{x(\cdot)} \sum_{s \in S_i} \left(\Pi^s \, \tilde{q}(x, s) \, \theta_{x(s)}\right)$ for each $i \in \{2, \ldots, k+1\}$. To do so, we show that given an $i \in \{2, \ldots, k+1\}$ and an optimal allocation x_i of ads to scenes in S_i, it is possible to design an assignment x_1 to scenes in S_1 with at least the same utility. Notice that an allocation problem restricted to nodes S_i, is equivalent to a problem without externalities since no scene in the set does provide externalities to other scenes in the set. Hence, Algorithm 7.1 provides an optimal solution to the problem. As shown in the proof of Theorem 7.1, Algorithm 7.1 assigns ads from the top to the bottom of the tree. Moreover, if the ties are broken always in the same way, it is easy to see that an allocation returned by an Algorithm 7.1 is such that the same ad is allocated to all the scenes at the same depth. Formally, given the allocation x_i returned by the algorithm we can define a function $\bar{x}_i : \mathbb{N} \to A$ such that for each $s \in S_i$ it holds $x_i(s) = \bar{x}_i(|\boldsymbol{\sigma}^s|)$. Let x_i be an optimal allocation for the set of scenes S_i and let $\bar{x}_i : \mathbb{N} \to A$ be the function that defines the allocation. For each $s \in S_1$, let $\psi(s)$ be the set of nodes s' such that $s \in \boldsymbol{\sigma}^{s'}$ and $S_1 \cap (\boldsymbol{\sigma}^{s'} \setminus \boldsymbol{\sigma}^s) = \emptyset$, *i.e.*, s is the last node in S_1 that precedes s'. Notice that we have shown that given an $s \in S_1$, for each $s', s'' \in \psi(s)$, it holds $x_i(s') = \bar{x}_i(|\boldsymbol{\sigma}^{s'}|) = \bar{x}_i(|\boldsymbol{\sigma}^{s''}|) = x_i(s'')$. Hence, we can define a new allocation x_1 on scenes in S_1 such that for each $s \in S_1$, it holds $x_1(s) = x_i(s')$ for each $s' \in \psi(s)$. Then, it holds

$$\begin{aligned}
\sum_{s \in S_1} \left(\Pi^s \, \tilde{q}(x_1, s) \, \theta_{x_1(s)}\right) &\geq \sum_{s \in S_1} \left(\sum_{s' \in \psi(s)} \Pi^{s'} \right) \tilde{q}(x_1, s) \, \theta_{x_1(s)} = \\
&= \sum_{s \in S_1} \sum_{s' \in \psi(s)} \Pi^{s'} \, \tilde{q}(x_i, s') \, \theta_{x_i(s')} = \\
&= \sum_{s \in S_i} \Pi^s \, \tilde{q}(x_i, s) \, \theta_{x_i(s)}.
\end{aligned}$$

This proves that

$$\max_{x(\cdot)} \sum_{s \in S_1} \left(\Pi^s \, \tilde{q}(x, s) \, \theta_{x(s)}\right) \geq \max_{x(\cdot)} \sum_{s \in S_i} \left(\Pi^s \, \tilde{q}(x, s) \, \theta_{x(s)}\right)$$

for each $i \in \{2, \ldots, k+1\}$.

Since there are no externalities among the scenes in S_1, Algorithm 7.2 returns the optimal allocation x^* to the scenes in S_1 by Theorem 7.1 . Hence, it holds

$$\sum_{s\in S_1}\left(\Pi^s\,\tilde{q}(x^*,s)\,\theta_{x^*(s)}\right)=\max_{x(\cdot)}\sum_{s\in S_1}\left(\Pi^s\,\tilde{q}(x,s)\,\theta_{x(s)}\right)\geq$$
$$\geq\max_{x(\cdot)}\sum_{s\in S_i}\left(\Pi^s\,\tilde{q}(x,s)\,\theta_{x(s)}\right)$$

for each $i\in\{1,\dots,k\}$. This implies that

$$\sum_{s\in S_1}\left(\Pi^s\,\tilde{q}(x^*,s)\,\theta_{x^*(s)}\right)\geq\frac{1}{k+1}\sum_{i\in\{1,\dots,k+1\}}\max_{x(\cdot)}\sum_{s\in S_i}\left(\Pi^s\,\tilde{q}(x,s)\,\theta_{x(s)}\right)$$
$$\geq\frac{1}{k+1}\max_{x(\cdot)}\sum_{i\in\{1,\dots,k+1\}}\sum_{s\in S_i}\left(\Pi^s\,\tilde{q}(x,s)\,\theta_{x(s)}\right)$$
$$=\frac{1}{k+1}\max_{x(\cdot)}\sum_{s\in S}\left(\Pi^s\,\tilde{q}(x,s)\,\theta_{x(s)}\right)$$

where while the second comes from the fact that the externalities are only negative. This concludes the proof.

□

Algorithm 7.2 GREEDY–SI–E

Inputs: set of scenes S, set of ads A, memory length k

1: $S_1\leftarrow\{s\in S:|\boldsymbol{\sigma}^s|\in\{1+j(k+1)\}_{j\in\mathbb{N}}\}$
2: $x^*\leftarrow\textsc{Greedy}(S_1,A)$
3: **return** x^*

Finally, we focus on Myerson's weak monotonicity, and we show that Algorithm 7.2 is weakly monotone.

Proposition 7.2. *Algorithm 7.2 is weakly monotone (in the sense of Myerson).*

Proof. To prove the statement, it is sufficient to notice that Algorithm 7.2 is equivalent to Algorithm 7.1 restricted to the subset of scenes S_1. Then, the monotonicity of the algorithm comes from the monotonicity of Algorithm 7.1. In particular, Algorithm 7.1 is weakly monotone since it is an utility maximizing mechanism. □

Therefore, the resulting mechanism in which the allocation function is given by Algorithm 7.2 and the payments are *à la* Myerson is truthful in dominant strategies.

7.5 META-SD-E: Approximating the General Problem

In this section, we deal with the general ad allocation problem in which there are both externalities and scene-dependent qualities. As Theorems 7.3 and 7.6 show, META-SD-E is an Poly-APX-Hard problem. In particular, Theorem 7.6 rules out the possibility of providing approximation sublinear in k in polynomial time. In this section, we show that the problem admits a polynomial-time algorithm providing a $(\frac{1-1/e}{k+1})$-approximation, thus matching the inapproximability lower bound provided by Theorem 7.6.

Let the matroid $(G, \mathcal{I})$ and the function f be defined as in Section 7.3. We show that we can apply Algorithm 7.3 to find a $\frac{1-1/e}{k+1}$-approximation to the META-SD-E problem. In particular, the algorithm follows the approach of Section 7.4, except that it need to evaluates all the sets of scenes $\{1+j(k+1)\}_{j\in\mathbb{N}}, \{2+j(k+1)\}_{j\in\mathbb{N}}, \ldots, \{k+j(k+1)\}_{j\in\mathbb{N}}$ as the qualities depend on the scenes. Intuitively, the rationale is to enumerate these sets of scenes and, for each one of them, to approximate the optimal allocation that employs only those scenes with the continuous greedy algorithm used for the META-SD-NE problem. This is needed since the qualities depend on the scenes and hence we have to include each scene in at least one of the considered allocations. We denote with CONTINUOUSGREEDY(S, A, f) the continuous greedy algorithm that works with the matroid defined in Section 7.3 that considers only scenes S and ads A, and optimize the monotone submodular function f defined is Section 7.3. Finally, we take the best allocation among those evaluated by the algorithm. The resulting approximation factor combines the approximation factors of both META-SD-NE and META-SI-E problems. The pseudocode is provided in Algorithm 7.3. Formally, we can prove the following result.

Algorithm 7.3 GREEDY–SD–E

Inputs: set of scenes S, set of ads A, memory length k

1: **for** $i \in \{1, \ldots, k\}$ **do**
2: $\quad S_i \leftarrow \{s \in S : |\boldsymbol{\sigma}^s| \in \{i + j(k+1)\}_{j\in\mathbb{N}}\}$
3: $\quad x_i \leftarrow$ CONTINUOUSGREEDY(S_i, A, f)
4: **end for**
5: $i^* \leftarrow \text{argmax}_{i\in\{1,\ldots,k\}} \sum_{s\in S} \left(\Pi^s\, \tilde{q}(x_i, s)\, \theta_{x_i(s)}\right)$
6: **return** x_{i^*}

Theorem 7.8. *Algorithm 7.3 provides a* $\frac{1-1/e}{k+1}$*-approximation to* META-SD-E. *Moreover, it runs in polynomial time.*

Proof. As a first step, we show that the given an i allocation x_i provides a

good approximation of the optimal allocation value considering only scenes S_i. Formally, we show that

$$\sum_{s \in S_i} \left(\Pi^s \, \tilde{q}(x_i, s) \, \theta_{x_i(s)} \right) \geq (1 - \frac{1}{e}) \max_{x(\cdot)} \sum_{s \in S_i} \left(\Pi^s \, \tilde{q}(x, s) \, \theta_{x(s)} \right).$$

To see that, it sufficient to apply Theorem 7.4 to the problem restricted to scenes S_i. The inequality holds since by considering only the scenes in S_i (that have distance larger than the memory k) the problem is equivalent to a problem without externalities. Let x^* be the allocation returned by Algorithm 7.3. Then, it holds

$$\begin{aligned}
&\sum_{s \in S_i} \left(\Pi^s \, \tilde{q}(x^*, s) \, \theta_{x^*(s)} \right) = \\
&= \operatorname*{argmax}_{i \in \{1,\dots,k+1\}} \sum_{s \in S} \left(\Pi^s \, \tilde{q}(x_i, s) \, \theta_{x_i(s)} \right) \geq \\
&\geq \frac{1}{k+1} \sum_{i \in \{1,\dots,k+1\}} \sum_{s \in S} \left(\Pi^s \, \tilde{q}(x_i, s) \, \theta_{x_i(s)} \right) \geq \\
&\geq \frac{1}{k+1} \sum_{i \in \{1,\dots,k+1\}} (1 - \frac{1}{e}) \max_{x(\cdot)} \sum_{s \in S_i} \left(\Pi^s \, \tilde{q}(x, s) \, \theta_{x(s)} \right) \geq \\
&\geq \frac{1}{k+1} (1 - \frac{1}{e}) \max_{x(\cdot)} \sum_{i \in \{1,\dots,k+1\}} \sum_{s \in S_i} \left(\Pi^s \, \tilde{q}(x, s) \, \theta_{x(s)} \right) = \\
&= \frac{1}{k+1} (1 - \frac{1}{e}) \max_{x(\cdot)} \sum_{s \in S} \left(\Pi^s \, \tilde{q}(x_i, s) \, \theta_{x_i(s)} \right),
\end{aligned}$$

where the second inequality comes from the guarantees of the continuous greedy algorithm (Theorem 7.4), while the third comes from the fact that the externalities are only negative. This concludes the proof. □

We conclude by showing that neither the greedy Algorithm 7.1 nor its extension Algorithm 7.2 are weakly monotone for the META-SD-E problem. This result follows from the non-monotonicity of the greedy Algorithm 7.1 in the simpler setting with no externalities and scene-dependent ad qualities.

Proposition 7.3. *Neither Algorithm 7.1 nor Algorithm 7.2 are weakly monotone (in the sense of Myerson) for the* `META-SD-E` *problem.*

Proof. Proposition 7.1 shows that Algorithm 7.1 is not monotone even in the simpler setting with no externalities. We conclude the proof providing a negative result also for Algorithm 7.2. To see that Algorithm 7.2 is not

weakly monotone, it is sufficient to notice that when the memory is $k = 0$ Algorithm 7.2 is equivalent to Algorithm 7.1. □

Remark 7.1. *We can also derive an algorithm similar to Algorithm 7.3 in which we replace the continuous greedy algorithm with the greedy Algorithm 7.1 and obtain a $\frac{1/2}{k+1}$-approximation factor. However, symilarly to the other greedy algorithms, this algorithm is not weakly monotone for the* `META-SD-E` *problem.*

7.6 The Importance of Algorithms Exploiting a User Model for the Metaverse

In this section, we assess the importance of adopting our model in practice. More precisely, under the assumption that the users behave as described by our model, we compare the performance of algorithms disregarding basic user features in the metaverse (*i.e.*, scene-dependent qualities, externalities, sequential traversal of scenes) with the performance of our approximation algorithms. In particular, we show that those algorithms can be arbitrarily inefficient w.r.t. our algorithms in instances that are not knife-edge.

Initially, we focus on the importance of scene-dependent qualities. The following example shows that, when the user behavior depends on the scene and there are no externalities, disregarding such dependence leads to allocations whose value is arbitrarily smaller than the value returned by our approximation algorithms.

Example 7.2. *Consider a setting with two scenes s_1, s_2 where $s_2 \in \rho(s_1)$ and two ads a_1, a_2 with the following parameters:*

$$q_{a_1,s_1} = 0 \qquad q_{a_2,s_1} = 1$$
$$q_{a_1,s_2} = 1 \qquad q_{a_2,s_2} = 0$$
$$\theta_{a_1} = 1 + \epsilon \qquad \theta_{a_2} = 1$$

while $\gamma_{a,a'} = 1$ for every pair a, a', $\pi_{s_1,s_2} = 1 - \delta$, and $k = 0$, where $\delta, \epsilon > 0$ are arbitrarily small. This setting is common whenever the ads are specialized, performing well only in a small number of scenes.

Focus on the case in which we use an algorithm based on our model. In this case, we can resort to Algorithm 7.1 which guarantees at least $1/2$ of the optimal value and returns the allocation $x(s_1) = a_2$ and $x(s_2) = a_1$ with a value of $2 + \epsilon - \delta \approx 2$. This allocation is optimal.

Focus on the case in which we use an algorithm disregarding that the ad qualities depend on the scenes. Such an algorithm necessarily works

with scene-independent ad qualities that are average values of the scene-dependent qualities (in practice, we can imagine that the algorithm estimates these parameters by collecting the samples from all the scenes and then making the empiric average). That is, $q_{a_1} = 1/2 = q_{a_2}$*. In this case, the greedy algorithm would return the allocation* $x(s_1) = a_1$ *and* $x(s_2) = a_2$ *with a value of* 0*. Therefore, such an algorithm has an approximation ratio of* 0*.*

Now, we focus on the importance of externalities. The following example shows that, when the behavior of the user is affected by externalities among the ads and the qualities are scene-independent, disregarding the externalities can lead to an allocation whose value is arbitrarily smaller than the value returned by our approximation algorithms.

Example 7.3. *Consider a setting with* $2n$*, with* $n \in \mathbb{N}$*, scenes* $\{s_1, \ldots, s_n\}$ *where* $s_{i+1} \in \rho(s_i)$ *and* $\pi_{s_i,s_{i+1}} = 1$ *for every* $i < n$*. There are* $2n$ *ads with* $q_a = 1$*,* $\theta_a = 1$*, and* $\gamma_{a,a'} = 0$ *for every other* a'*. The memory length is* $k = 1$*.*

Our greedy Algorithm 7.2 guarantees at least $1/2$ *of the optimal value. In this case, it allocates ads in the odd scenes, providing a value of* n *which is the optimal allocation.*

The greedy algorithm which does not consider the externalities allocates one ad per scene, providing a value of 1 *which corresponds to a* $1/n$*-ratio of the optimal value.*

Finally, we focus on the importance of considering that a user traverses several scenes according to some probability distribution. When disregarding this feature, an algorithm would repeat an allocation problem for every scene independently from the others (this is what happens on the Web). It can be easily seen that Example 7.3 shows that those algorithms would be arbitrarily inefficient when compared with our approximation algorithms as they would always allocate ads to all the scenes and these ads can generate externalities to the ads allocated in the following scenes. A similar result can be obtained by using Example 7.2. To conclude, disregarding every single basic user feature in the metaverse can lead to arbitrarily bad allocations.

Part III

Guaranteeing Properties During the Learning Process

CHAPTER 8

Safe Online Bid Optimization with Return-On-Investment and Budget Constraints subject to Uncertainty

Section 1.3 introduced the problem studied in this chapter, which is structured as follows. Section 8.1 formally states the problem. Section 8.2 introduces a meta-algorithm which will be investigated in the subsequent sections. Section 8.3 studies the optimization problem without uncertainty, whereas Section 8.4 investigates the online learning problem. Finally, Section 8.5 experimentally evaluates of our algorithms in settings generated from real-world data.

Other Related Works. Many works study Internet advertising, both from the *publisher* perspective (*e.g.*, Nisan et al. (2007) design auctions for ads allocation and pricing) and from the *advertiser* perspective (*e.g.*, Feldman et al. (2007) study the budget optimization problem in search advertising). Moreover, many recent works focus on the repeated interaction between the advertisers and the publisher (Abeille et al., 2018; Croissant et al., 2020; Nedelec et al., 2022). Few works deal with ROI constraints, and, to the best of our knowledge, they only focus on the design of auction mechanisms.

In particular, Szymanski and Lee (2006) and Borgs et al. (2007) show that ROI-based bidding heuristics can lead to cyclic behavior and reduce the allocation's efficiency, while Golrezaei et al. (2021b) propose more efficient auctions with ROI constraints. The learning algorithms for daily bid optimization available in the literature address only budget constraints in the restricted case in which the platform allows the advertisers to set a daily budget limit (notice that some platforms such as, *e.g.*, TripAdvisor and Trivago, do not even allow the setting of the daily budget limit). For instance, Zhang et al. (2012) provide an *offline* algorithm that exploits accurate models of the campaigns' performance based on low-level data rarely available to the advertisers. Nuara et al. (2018) propose an *online* learning algorithm that combines combinatorial multi-armed bandit techniques (Chen et al., 2013) with regression by Gaussian Processes (Rasmussen and Williams, 2006). This work provides no guarantees on ROI. More recent works also present pseudo-regret bounds (Nuara et al., 2022) and study subcampaigns interdependencies offline (Nuara et al., 2019). Thomaidou et al. (2014) provide a genetic algorithm for budget optimization of advertising campaigns. Ding et al. (2013) and Trovò et al. (2016) address the bid optimization problem in a single subcampaign scenario when the budget constraint is cumulative over time.

A research field strictly related to our work is learning with safe exploration with constraints subject to uncertainty, and the goal is to guarantee w.h.p. their satisfaction during the entire learning process. The only known results on safe exploration in multi-armed bandits address the case with continuous, convex arm spaces and convex constraints. The learner can converge to the optimal solution in these settings without violating the constraints (Moradipari et al., 2020; Amani et al., 2020). Conversely, the case with discrete and/or non-convex arm spaces or non-convex constraints, such as ours, is unexplored in the literature so far. We remark that some bandit algorithms address uncertain constraints where the goal is their satisfaction on average (Mannor et al., 2009; Cao and Liu, 2018). However, the per-round violation can be arbitrarily large (particularly in the early stages of the learning process), not fitting with our setting as humans could be alarmed and, thus, give up on adopting the algorithm. We also notice that several other works in reinforcement learning (Hans et al., 2008; Pirotta et al., 2013; Garcia and Fernández, 2012) and multi-armed bandit (Galichet et al., 2013; Sui et al., 2015) investigate safe exploration, providing safety guarantees on the revenue provided by the algorithm, but not on the satisfaction w.h.p. of uncertain constraints.

8.1 Problem Formulation

We are given an advertising campaign $\mathcal{C} = \{C_1, \dots, C_N\}$, with $N \in \mathbb{N}$ and where C_j is the j-th subcampaign, and a finite time horizon of $T \in \mathbb{N}$ rounds (each corresponding to one day in our application). In this work, as common in the literature on ad allocation optimization, we refer to a subcampaign as a single ad or a group of homogeneous ads requiring to set the same bid. For every round $t \in \{1, \dots, T\}$ and every subcampaign C_j, an advertiser needs to specify the bid $x_{j,t} \in X_j$, where $X_j \subset \mathbb{R}^+$ is a finite set of values that can be set for subcampaign C_j. For every round $t \in \{1, \dots, T\}$, the goal is to find the values of bids maximizing the overall cumulative expected revenue while keeping the overall ROI above a fixed value $\Lambda \in \mathbb{R}^+$ and the overall budget below a daily value $\beta \in \mathbb{R}^+$. Formally, the resulting constrained optimization problem at round t is as follows:

$$\max_{(x_{1,t},\dots,x_{N,t}) \in X_1 \times \dots \times X_N} \sum_{j=1}^{N} v_j \; n_j(x_{j,t}) \tag{8.1a}$$

$$\text{s.t.} \quad \frac{\sum_{j=1}^{N} v_j \; n_j(x_{j,t})}{\sum_{j=1}^{N} \; c_j(x_{j,t})} \geq \Lambda, \tag{8.1b}$$

$$\sum_{j=1}^{N} c_j(x_{j,t}) \leq \beta, \tag{8.1c}$$

where $n_j(x_{j,t})$ and $c_j(x_{j,t})$ are the expected number of clicks and the expected cost given the bid $x_{j,t}$ for subcampaign C_j, respectively, and v_j is the value per click for subcampaign C_j. Moreover, Constraint (8.1b) is the ROI constraint, forcing the revenue to be at least Λ times the costs, and Constraint (8.1c) keeps the daily spend under a predefined overall budget β.[1]

We focus on the customary case in which $n_j(\cdot)$ and $c_j(\cdot)$ are unknown functions whose values need to be estimated online. Our problem can be naturally modeled as a multi-armed bandit where the available *arms* are the different values of the bid $x_{j,t} \in X_j$ satisfying the combinatorial constraints of the optimization problem.[2] A *super-arm* is an arm profile specifying one bid per subcampaign. A *learning policy* $\mathfrak{U}$ solving such a problem is

[1] In economic literature, it is also used an alternative definition of ROI: $\frac{\sum_{j=1}^{N}[v_j \; n_j(x_{j,t}) - c_j(x_{j,t})]}{\sum_{j=1}^{N} \; c_j(x_{j,t})}$. We can capture this case by substituting the right hand side of Constraint (8.1b) with $\Lambda + 1$.

[2] Here, we assume that the value per click v_j is known. In the case it is unknown, we point an interested reader to Nuara et al. (2018) for details.

an algorithm returning, for every round t, a set of bid $\{\hat{x}_{j,t}\}_{j=1}^{N}$. Policy $\mathfrak{U}$ can only use estimates of the unknown number-of-click and cost functions built during the learning process. Therefore, the solutions returned by policy $\mathfrak{U}$ may not be optimal and/or violate Constraints (8.1b) and (8.1c) when evaluated with the true values. Notice that, even if this setting is closely related to the one studied by Badanidiyuru et al. (2018), the specific non-matroidal nature of the constraints does not allow to cast the bid allocation problem above into the bandit-with-knapsack framework.

We are interested in evaluating learning policies $\mathfrak{U}$ in terms of both loss of revenue (a.k.a. pseudo-regret) and violation of the ROI and budget constraints. The pseudo-regret and safety of a learning policy $\mathfrak{U}$ are defined as follows.

Definition 8.1 (Learning policy pseudo-regret)**.** *Given a learning policy* $\mathfrak{U}$*, we define the* pseudo-regret *as:*

$$R^T(\mathfrak{U}) := T\,G^* - \mathbb{E}\left[\sum_{t=1}^{T}\sum_{j=1}^{N} v_j\; n_j(\hat{x}_{j,t})\right],$$

where $G^* := \sum_{j=1}^{N} v_j\; n_j(x_j^*)$ *is the expected revenue provided by a clairvoyant algorithm, the set of bids* $\left\{x_j^*\right\}_{j=1}^{N}$ *is the optimal clairvoyant solution to the problem in Equations* (8.1a)–(8.1c)*, and the expectation* $\mathbb{E}[\,\cdot\,]$ *is taken w.r.t. the stochasticity of the learning policy* $\mathfrak{U}$*.*

Our goal is the design of algorithms that minimize the pseudo-regret $R^T(\mathfrak{U})$. In particular, we are interested in *no-regret* algorithms guaranteeing a regret that increases sublinearly in T. Now, we focus on the notion of safety.

Definition 8.2 (η-safe learning policy)**.** *Given* $\eta \in (0,\ T]$*, a learning policy* $\mathfrak{U}$ *is* η-safe *if* $\{\hat{x}_{j,t}\}_{j=1}^{N}$*,* i.e., *the expected number of times at least one of the Constraints* (8.1b) *and* (8.1c) *is violated from* $t = 1$ *to* T *is less than* η *or, formally:*

$$\sum_{t=1}^{T} \mathbb{P}\left(\frac{\sum_{j=1}^{N} v_j\; n_j(\hat{x}_{j,t})}{\sum_{j=1}^{N}\; c_j(\hat{x}_{j,t})} < \Lambda \vee \sum_{j=1}^{N} c_j(\hat{x}_{j,t}) > \beta\right) \leq \eta.$$

Our goal is the design of safe algorithms that minimize η. In particular, we are interested in safe algorithms guaranteeing that η increases sublinearly in (or is independent of) T.

Algorithm 8.1 Meta-algorithm

Input: sets of bid values $X_1, \ldots, X_N$, ROI threshold Λ, daily budget β
Initialize the GPs for the number of clicks and costs
for $t \in \{1, \ldots, T\}$ **do**
  for $j \in \{1, \ldots, N\}$ **do**
    for $x \in X_j$ **do**
      Call the estimation subroutine to estimate $\hat{n}_{j,t-1}(x)$, $\hat{\sigma}^n_{j,t-1}(x)$ using the GP on the number of clicks
      Call the estimation subroutine to estimate $\hat{c}_{j,t-1}(x)$, $\hat{\sigma}^c_{j,t-1}(x)$ using the GP on the costs
    end for
  end for
  Compute $\boldsymbol{\mu}$ using the GPs estimates
  Call the optimization subroutine $\mathsf{Opt}(\boldsymbol{\mu}, \Lambda)$ to get a solution $\{\hat{x}_{j,t}\}_{j=1}^N$
  Set the prescribed allocation during round t
  Get revenue $\sum_{j=1}^N v_j\, \tilde{n}_j(\hat{x}_{j,t})$
  Update the GPs using the new information $\tilde{n}_{j,t}(\hat{x}_{j,t})$ and $\tilde{c}_{j,t}(\hat{x}_{j,t})$
end for

8.2 Meta-algorithm

We provide the pseudo-code of our meta-algorithm in Algorithm 8.1, which solves the problem in Equations (8.1a)–(8.1c) online. Algorithm 8.1 is based on three components: Gaussian Processes (GPs) (Rasmussen and Williams, 2006) to model the parameters whose values are unknown (details are provided below), an *estimation subroutine* to generate estimates of the parameters from the GPs, and an *optimization subroutine* to solve the optimization problem once given the estimates.

In Algorithm 8.1, GPs are used to model functions $n_j(\cdot)$ and $c_j(\cdot)$, describing the expected number of clicks and the costs, respectively. The employment of GPs to model these functions provides several advantages w.r.t. other regression techniques, such as the provision of a probability distribution over the possible values of the functions for every bid value $x \in X_j$ relying on a finite set of samples. GPs use the noisy realization of the actual number of clicks $\tilde{n}_{j,h}(\hat{x}_{j,h})$ collected from each subcampaign C_j for every previous round $h \in \{1, \ldots, t-1\}$ to generate, for every bid $x \in X_j$, the estimates for the expected value $\hat{n}_{j,t-1}(x)$ and the standard deviation of the number of clicks $\hat{\sigma}^n_{j,t-1}(x)$. Analogously, using the noisy realizations of the actual cost $\tilde{c}_{j,h}(\hat{x}_{j,h})$, with $h \in \{1, \ldots, t-1\}$, GPs generate, for every bid $x \in X_j$, the estimates for the expected value $\hat{c}_{j,t-1}(x)$ and the standard deviation of the cost $\hat{\sigma}^c_{j,t-1}(x)$. Formally, we compute the above values as

follows:

$$\hat{n}_{j,t-1}(x) := \boldsymbol{k}_{j,t-1}(x)^\top (K_{j,t-1} + \sigma^2 I)^{-1} \boldsymbol{k}_{j,t-1}(x),$$
$$\hat{\sigma}^n_{j,t-1}(x) := k_j(x,x) - \boldsymbol{k}^\top_{j,t-1} (K_{j,t-1} + \sigma^2 I)^{-1} \boldsymbol{k}_{j,t-1}(x),$$
$$\hat{c}_{j,t-1}(x) := \boldsymbol{h}_{j,t-1}(x)^\top (H_{j,t-1} + \sigma^2 I)^{-1} \boldsymbol{h}_{j,t-1}(x),$$
$$\hat{\sigma}^c_{j,t-1}(x) := h_j(x,x) - \boldsymbol{h}^\top_{j,t-1} (H_{j,t-1} + \sigma^2 I)^{-1} \boldsymbol{h}_{j,t-1}(x),$$

where $k_j(\cdot,\cdot)$ and $h_j(\cdot,\cdot)$ are the kernels for the GPs over the number of clicks and costs, respectively, $K_{j,t-1}$ and $H_{j,t-1}$ are the Gram matrix over the training bids for the two GPs, σ^2 is the variance of the noise of the GPs, $\boldsymbol{k}_{j,t-1}(x)$ and $\boldsymbol{h}_{j,t-1}$ are vectors built computing the kernel between the training bids and the current bid x, and I is the identity matrix of order $t-1$. For further details on the use of GPs, we point an interested reader to Rasmussen and Williams (2006). We recall that the asymptotic running time of the GP estimation procedure is $\Theta(\sum_{j=1}^{N} |X_j|\ t^2)$, where t is the number of samples (corresponding to the rounds), and the asymptotic space complexity is $\Theta(Nt^2)$, *i.e.*, the space required to store the Gram matrix. A better, linear dependence on the number of days t can be obtained by using the recursive formula for the GP mean and variance computation (see Chowdhury and Gopalan (2017) for details).

The estimation subroutine returns the vector $\boldsymbol{\mu}$ composed of the estimates generated from the GPs. In the following sections, we investigate two different procedures to compute $\boldsymbol{\mu}$. Then, the vector $\boldsymbol{\mu}$ is given as input to the optimization subroutine, namely $\mathsf{Opt}(\boldsymbol{\mu}, \Lambda)$, solving the problem stated in Equations (8.1a)–(8.1c) and returns the bid strategy $\{\hat{x}_{j,t}\}_{j=1}^{N}$ to play the next round t. Finally, once the strategy has been applied, the revenue $\sum_{j=1}^{N} v_j\ \tilde{n}_j(\hat{x}_{j,t})$ is obtained, and the stochastic realization of the number of clicks $\tilde{n}_{j,t}(\hat{x}_{j,t})$ and costs $\tilde{c}_{j,t}(\hat{x}_{j,t})$ are observed and provided to the GPs to update the models that will be used at round $t+1$. For the sake of presentation, we first describe the optimization subroutine $\mathsf{Opt}(\boldsymbol{\mu}, \Lambda)$ and, then, some estimation subroutines together with the theoretical guarantees provided by Algorithm 8.1 when these subroutines are adopted.

8.3 Optimization Subroutine

At first, we show that, even if all the values of the parameters of the optimization problem are known, the optimal solution cannot be approximated in polynomial time within any strictly positive factor (even depending on the size of the instance), unless $\mathsf{P} = \mathsf{NP}$. We reduce from SUBSET-SUM that is an NP-hard problem. Given a set S of integers $u_i \in \mathbb{N}^+$ and an integer

Algorithm 8.2 $\mathsf{Opt}(\boldsymbol{\mu}, \Lambda)$ subroutine

Input: sets of bid values $X_1, \ldots, X_N$, set of cumulative cost values Y, set of revenue values R, vector $\boldsymbol{\mu}$, ROI threshold Λ

Initialize M empty matrix with dimension $|Y| \times |R|$
Initialize $\mathbf{x}^{y,r} = \mathbf{x}^{y,r}_{\text{next}} = [\ ], \forall y \in Y, r \in R$
$S(y,r) = \bigcup \{x \in X_1 |\ \bar{c}_1(x) \leq y \wedge \underline{w}_1(x) \geq r\}\ \ \forall y \in Y, r \in R$
$\mathbf{x}^{y,r} = \arg\max_{x \in S} \overline{w}_1(x)\ \ \forall y \in Y, r \in R$
$M(y,r) = \max_{x \in S} \overline{w}_1(x)\ \ \forall y \in Y, r \in R$
for $j \in \{2, \ldots, N\}$ **do**
    for $y \in Y$ **do**
        for $r \in R$ **do**
            Update $S(y,r)$ according to Equation (8.2)
            $\mathbf{x}^{y,r}_{\text{next}} = \arg\max_{\mathbf{s} \in S(y,r)} \sum_{i=1}^{j} \overline{w}_i(s_i)$
            $M(y,r) = \max_{\mathbf{s} \in S(y,r)} \sum_{i=1}^{j} \overline{w}_i(s_i)$
        end for
    end for
    $\mathbf{x}^{y,r} = \mathbf{x}^{y,r}_{\text{next}}$
end for
Choose (y^*, r^*) according to Equation (8.3)
Output: $\mathbf{x}^{y^*,r^*}$

$z \in \mathbb{N}^+$, SUBSET-SUM requires to decide whether there is a set $S^* \subseteq S$ with $\sum_{i \in S^*} u_i = z$.[3]

Theorem 8.1 (Inapproximability)**.** *For any $\rho \in (0, 1]$, there is no polynomial-time algorithm returning a ρ-approximation to the problem in Equations* (8.1a)–(8.1c)*, unless* $\mathsf{P} = \mathsf{NP}$.

It is well known that SUBSET-SUM is a weakly NP-hard problem, admitting an exact algorithm whose running time is polynomial in the size of the problem and the magnitude of the data involved rather than the base-two logarithm of their magnitude. The same can be showed for our problem. Indeed, we can design a pseudo-polynomial-time algorithm to find the optimal solution in polynomial time w.r.t. the number of possible values of revenues and costs. In real-world settings, the values of revenue and cost are in limited ranges and rounded to the nearest cent, allowing the problem to be solved in a reasonable time. For simplicity, in the following we assume the discretization of the ranges of the values of the daily cost Y and revenue R be evenly spaced.

The pseudo-code of the $\mathsf{Opt}(\boldsymbol{\mu}, \Lambda)$ subroutine, solving the problem in Equations (8.1a)–(8.1c) with a dynamic programming approach, is provided in Algorithm 8.2. It takes as input the set of the possible bid values X_j

[3] Some of the proofs of this chapter are deferred to Appendix C

for each subcampaign C_j, the set of the possible cumulative cost values Y such that $\max_{y \in Y} y = \beta$, the set of the possible revenue values R, a ROI threshold Λ, and a vector $\boldsymbol{\mu}$ of parameters characterizing the specific instance of the optimization problem that is defined as follows:

$$\boldsymbol{\mu} := \big[\overline{w}_1(x_1), \dots, \overline{w}_N(x_{|X_N|}), \underline{w}_1(x_1), \dots \\ \dots, \underline{w}_N(x_{|X_N|}), -\overline{c}_1(x_1), \dots, -\overline{c}_N(x_{|X_N|})\big],$$

where $w_j(x_j) := v_j\, n_j(x_j)$ denotes the revenue for a subcampaign C_j. We use $\overline{h}$ and $\underline{h}$ to denote potentially different estimated values of a generic function h used by the learning algorithms in the next sections. In particular, if the functions are known beforehand, then it holds $\overline{h} = \underline{h} = h$ for both $h = w_j$ and $h = c_j$. For the sake of clarity, $\overline{w}_j(x)$ is used in the objective function, while $\underline{w}_j(x)$ and $\overline{c}_j(x)$ are used in the constraints. At first, the subroutine initializes a matrix M in which it stores the optimal solution for each combination of values $y \in Y$ and $r \in R$, and it initializes the vectors $\mathbf{x}^{y,r} = \mathbf{x}^{y,r}_{\text{next}} = [\ ], \forall y \in Y, \forall r \in R$ (Lines 1 and 2, respectively). Then, the subroutine generates the set $S(y,r)$ of the bids for subcampaign C_1 (Line 3). More precisely, the set $S(y,r)$ contains only the bids x that induce the overall costs to be lower than or equal to y and the overall revenue to be higher than or equal to r. The bid in $S(y,r)$ that maximizes the revenue calculated with parameters $\overline{w}_j$ is included in the vector $\mathbf{x}^{y,r}$, while the corresponding revenue is stored in the matrix M (Lines 4–5). Then, the subroutine iterates over each subcampaign C_j, with $j \in \{2, \dots, N\}$, all the values $y \in Y$, and all the values $r \in R$ (Lines 9–11). At each iteration, for every pair (y,r), the subroutine stores in $\mathbf{x}^{y,r}$ the optimal set of bids for subcampaigns $C_1, \dots, C_j$ that maximizes the objective function and stores the corresponding optimum value in $M(y,r)$. At every j-th iteration, the computation of the optimal bids is performed by evaluating a set of candidate solutions $S(y,r)$, computed as follows:

$$S(y,r) := \bigcup \Big\{ \mathbf{s} = [\mathbf{x}^{y',r'}, x] \text{ s.t. } y' + \overline{c}_j(x) \le y \wedge \\ r' + \underline{w}_j(x) \ge r \wedge x \in X_j \wedge y' \in Y \wedge r' \in R \Big\}. \tag{8.2}$$

This set is built by combining the optimal bids $\mathbf{x}^{y',r'}$ computed at the $(j-1)$-th iteration with one of the bids $x \in X_j$ available for the j-th subcampaign, such that these combinations satisfy the ROI and budget constraints. Then, the subroutine assigns the element of $S(y,r)$ that maximizes the revenue to $\mathbf{x}^{y,r}_{\text{next}}$ and the corresponding revenue to $M(y,r)$. At the end, the subroutine

computes the optimal pair (y^*, r^*) as follows:

$$(y^*, r^*) = \Big\{ y \in Y, r \in R \text{ s.t. } \frac{r}{y} \geq \Lambda \wedge M(y,r) \geq M(y',r'), \quad \forall y' \in Y, \forall r' \in R \Big\}, \tag{8.3}$$

and the corresponding set of bids $\mathbf{x}^{y^*,r^*}$, containing one bid for each subcampaign. We can state the following:

Theorem 8.2 (Optimality)**.** *The* Opt$(\boldsymbol{\mu}, \Lambda)$ *subroutine returns the optimal solution to the problem in Equations* (8.1a)–(8.1c) *when* $\overline{w}_j(x) = \underline{w}_j(x) = v_j\, n_j(x)$ *and* $\overline{c}_j(x) = c_j(x)$ *for each* $j \in \{1, \ldots, N\}$ *and the values of revenues and costs are in* R *and* Y*, respectively.*

Proof. Since all the possible values for the revenues and costs are taken into account in the subroutine, the elements in $S(y,r)$ satisfy the two inequalities in Equation (8.2) with the equal sign. Therefore, all the elements in $S(y,r)$ would contribute to the computation of the final value of the ROI and budget constraints, *i.e.*, the ones after evaluating all the N subcampaigns, with the same values for revenue and costs, being their overall revenue equal to r and their overall cost equal to y. Notice that Constraint (8.1c) is satisfied as long as it holds $\max(Y) = \beta$. The maximum operator in Line 11 excludes only solutions with the same costs and a lower revenue, therefore, the subroutine excludes only solutions that would never be optimal (and, for this reason, said dominated). The same reasoning holds also for the subcampaign C_1 analysed by the algorithm. Finally, after all the dominated allocations have been discarded, the solution is selected by Equation (8.3), *i.e.*, among all the solutions satisfying the ROI constraints the one with the largest revenue is selected. □

The asymptotic running time of Opt is $\Theta\left(\sum_{j=1}^{N} |X_j|\, |Y|^2\, |R|^2\right)$, where $|X_j|$ is the cardinality of the set of bids X_j, since it cycles over all the subcampaigns and, for each one of them, finds the maximum bids and compute the values in the matrix $S(y,r)$. Moreover, the asymptotic space complexity of the Opt procedure is $\Theta\left(\max_{j=\{1,\ldots,N\}} |X_j|\, |Y|\, |R|\right)$ since it stores the values in the matrix $S(y,r)$ and finds the maximum over the possible bids $x \in X_j$.

8.4 Estimation Subroutine

Initially, we focus on the nature of our learning problem, and we show that no online learning algorithm can provide a sublinear pseudo-regret while

guaranteeing safety.

Theorem 8.3 (Pseudo-regret/safety tradeoff)**.** *For every $\epsilon > 0$ and time horizon T, there is no algorithm with pseudo-regret smaller than $(1/2-\epsilon)\,T$ that violates (in expectation) the constraints less than $(1/2-\epsilon)\,T$ times.*

Proof. In what follows, we provide an impossibility result for the optimization problem in Equations (8.1a)–(8.1c). For the sake of simplicity, our proof is based on the violation of (budget) Constraint (8.1c), but its extension to the violation of (ROI) Constraint (8.1b) is direct.

Initially, we show that an algorithm satisfying the two conditions of the theorem can be used to distinguish between $\mathcal{N}(1,1)$ and $\mathcal{N}(1+\delta,1)$ with an arbitrarily large probability using a number of samples independent from δ. Consider two instances of the bid optimization problem defined as follows. Both instances have a single subcampaign with $x \in \{0,1\}$, $c(0)=0$, $r(0)=0$, $r(1)=1$, $\beta=1$, and $\Lambda=0$. The first instance has cost $c^1(1)=\mathcal{N}(1,1)$, while the second one has cost $c^2(1)=\mathcal{N}(1+\delta,1)$. With the first instance, the algorithm must choose $x=1$ at least $T(1/2+\epsilon)$ times in expectation, otherwise the pseudo-regret would be strictly greater than $T(1/2-\epsilon)$, while, with the second instance, the algorithm must choose $x=1$ at most than $T(1/2-\epsilon)$ times in expectation, otherwise the constraint on the budget would be violated strictly more than $T(1/2-\epsilon)$ times. Standard concentration inequalities imply that, for each $\gamma>0$, there exists a $n(\epsilon,\gamma)$ such that, given $n(\epsilon,\gamma)$ runs of the learning algorithm, with the first instance the algorithm plays $x=1$ strictly more than $Tn(\epsilon,\gamma)/2$ times with probability at least $1-\gamma$, while with the second instance it is played strictly less than $Tn(\epsilon,\gamma)/2$ times with probability at least $1-\gamma$. This entails that the learning algorithm can distinguish with arbitrarily large success probability (independent of δ) between the two instances using (at most) $n(\epsilon,\gamma)T$ samples from one of the normal distributions.

However, the Kullback-Leibler divergence between the two normal distributions is $KL(\mathcal{N}(1,1),\mathcal{N}(1+\delta,1))=\delta^2/2$ and each algorithm needs at least $\Omega(1/\delta^2)$ samples to distinguish between the two distributions with arbitrarily large probability. Since δ can be arbitrarily small, we have a contradiction. Thus, such an algorithm cannot exist. This concludes the proof.[4] □

This impossibility result is crucial in practice, showing that no online learning algorithm can theoretically guarantee both a sublinear regret and

[4]Notice that the theorem can be modified to hold even with instances that satisfy real-world assumptions, *e.g.*, with costs much smaller than the budget. Indeed, we can apply the same reduction in which the costs are arbitrary, *e.g.*, $c(0)=c(1)=q$ with an arbitrary small q and $\beta=1$, while the utilities are $r(0)=0$, $r(1)=\mathcal{N}(1,1)$ or $r(1)=\mathcal{N}(1-\delta,1)$, and the ROI limit is $\Lambda=1/q$.

a sublinear number of violations of the constraints. Therefore, in real-world applications, advertisers have necessarily to accept a tradeoff between the two requirements. The following sections describe three estimation subroutines, each providing theoretical guarantees for a different, relaxed version of the optimization problem. More precisely, in Section 8.4.1, we relax the safety requirement and provide an algorithm, namely GCB, guaranteeing a sublinear regret. In Section 8.4.2, we relax the no-regret requirement and provide an algorithm, namely $\mathsf{GCB}_{\mathtt{safe}}$, guaranteeing safety. In Section 8.4.3, we accept a fixed tolerance (ψ, ϕ) in the safety requirements and provide an algorithm, namely $\mathsf{GCB}_{\mathtt{safe}}(\psi, \phi)$, guaranteeing both a sublinear regret and a sublinear number of violations of the constraints.

8.4.1 Guaranteeing Sublinear Pseudo-regret: GCB

Accabi et al. (2018) propose the GCB algorithm to face general combinatorial bandit problems where the arms are partitioned in subsets and the payoffs of the arms belonging to the same subset are modeled with a GP.[5] To obtain theoretical sublinear guarantees on the regret for our online learning problem, we use a specific definition of $\boldsymbol{\mu}$ vector, making Algorithm 8.1 be an extension of GCB to the case in which the payoffs and constraints are functions whose parameters are modeled by multiple independent GPs. With a slight abuse of terminology, we refer to this extension as GCB.

The GCB algorithm relies on the idea to build the $\boldsymbol{\mu}$ vector so that the parameters corresponding to the reward are statistical upper bounds to the expected values of the random variables, and those corresponding to the costs are statistical lower bounds. The rationale is that this choice satisfies the optimism vs. uncertainty principle. Formally, we have:

$$\overline{w}_j(x) = \underline{w}_j(x) := v_j \left[\hat{n}_{j,t-1}(x) + \sqrt{b_{t-1}} \hat{\sigma}^n_{j,t-1}(x) \right], \tag{8.4}$$

$$\overline{c}_j(x) := \hat{c}_{j,t-1}(x) - \sqrt{b_{t-1}} \hat{\sigma}^c_{j,t-1}(x), \tag{8.5}$$

where $b_t := 2 \ln \left(\frac{\pi^2 N Q T t^2}{3\delta} \right)$ is an uncertainty term used to guarantee the confidence level required by GCB. [6]

In what follows we bound the GCB pseudo-regret in terms of the *maximum information gain* $\gamma_{j,t}$ of the GP modeling the number of clicks of

[5] GCB algorithm was presented at EWRL 2018 without any archival version of the paper.

[6] For the sake of simplicity, we assume that the values of the bounds correspond to values in R and Y, respectively. If the bound values for $\overline{w}_j(x)$ are not in the set R, we need to round them up to the nearest value belonging to R. Instead, if $\underline{c}_j(x)$ are not in the set Y, a rounding down should be performed to the nearest value in Y.

subcampaign C_j at round t, formally defined as:

$$\gamma_{j,t} := \frac{1}{2} \max_{(x_{j,1},\ldots,x_{j,t}),x_{j,h}\in X_j} \left| I_t + \frac{\Phi(x_{j,1},\ldots,x_{j,t})}{\sigma^2} \right|,$$

where I_t is the identity matrix of order t, $\Phi(x_{j,1},\ldots,x_{j,t})$ is the Gram matrix of the GP computed on the vector $(x_{j,1},\ldots,x_{j,t})$, and $\sigma \in \mathbb{R}^+$ is the noise standard deviation. Thanks to this definition, we can state the following.

Theorem 8.4 (GCB pesudo-regret)**.** *Given $\delta \in (0,\ 1)$, GCB applied to the problem in Equations* (8.1a)–(8.1c)*, with probability at least $1-\delta$, suffers from a pseudo-regret of:*

$$R^T(\textit{GCB}) \leq \sqrt{\frac{8Tv_{\max}^2 N^3 b_T}{\ln(1+\sigma^2)} \sum_{j=1}^{N} \gamma_{j,T}} \ ,$$

where $b_t := 2\ln\left(\frac{\pi^2 NQTt^2}{3\delta}\right)$ is an uncertainty term used to guarantee the confidence level required by GCB, $v_{\max} := \max_{j\in\{1,\ldots,N\}} v_j$ is the maximum value per click over all subcampaigns, and $Q := \max_{j\in\{1,\ldots,N\}} |X_j|$ is the maximum number of bids in a subcampaign.

We remark that the upper bound provided in the above theorem is expressed in terms of the maximum information gain $\gamma_{j,T}$ of the GPs over the number of clicks. The problem of bounding $\gamma_{j,T}$ for a generic GP has been already addressed by Srinivas et al. (2010), where the authors present the bounds for the squared exponential kernel $\gamma_{j,T} = \mathcal{O}((\ln T)^2)$ for 1-dimensional GPs. Notice that, thanks to the previous result, the GCB algorithm using squared exponential kernels suffers from a sublinear pseudo-regret since the terms $\gamma_{j,T}$ is bounded by $\mathcal{O}((\ln T)^2)$, and the bound in Theorems 8.4 is $\mathcal{O}(N^{3/2}(\ln T)^{5/2})\sqrt{T})$.

On the other hand, the GCB algorithm violates (in expectation) the constraints a linear number of times in T as stated by the following theorem.

Theorem 8.5 (GCB safety)**.** *Given $\delta \in (0,\ 1)$, GCB applied to the problem in Equations* (8.1a)–(8.1c) *is η-safe where $\eta \geq T - \frac{\delta}{2NQT}$ and, therefore, the number of constraints violations is linear in T.*

Proof. Let us focus on a specific day t. Consider the case in which Constraints (8.1b) and (8.1c) are active, and, therefore, the left side equals the right side: $\sum_{j=1}^{N} \underline{w}_j(x_{j,t}) - \Lambda \sum_{j=1}^{N} \overline{c}_j(x_{j,t}) = 0$ and $\sum_{j=1}^{N} \overline{c}_j(x_{j,t}) = \beta$.

For the sake of simplicity, we focus on the costs $\overline{c}_j(x_{j,t})$, but similar arguments also apply to the revenues $\underline{w}_j(x_{j,t})$. A necessary condition for which the two constraints are valid also for the actual (non-estimated) revenues and costs is that for at least one of the costs it holds $c_j(x_{j,t}) \leq \overline{c}_j(x_{j,t})$. Indeed, if the opposite holds, *i.e.,* $\overline{c}_j(x_{j,t}) < c_j(x_{j,t})$ for each $j \in \{1, \dots, N\}$ and $x_{j,t} \in X_j$, the budget constraint would be violated by the allocation since $\sum_{j=1}^{N} c_j(x_{j,t}) > \sum_{j=1}^{N} \overline{c}_j(x_{j,t}) = \beta$. Since the event $c_j(x_{j,t}) \leq \overline{c}_j(x_{j,t})$ occurs with probability at most $\frac{3\delta}{\pi^2 NQTt^2}$, over the $t \in \mathbb{N}$, formally:

$$\mathbb{P}\left(\frac{\sum_{j=1}^{N} v_j \; n_j(\hat{x}_{j,t})}{\sum_{j=1}^{N} \; c_j(\hat{x}_{j,t})} < \Lambda \vee \sum_{j=1}^{N} c_j(\hat{x}_{j,t}) > \beta\right) \geq 1 - \frac{3\delta}{\pi^2 NQTt^2}.$$

Finally, summing over the time horizon T the probability that the constraints are not violated is at most $\frac{\delta}{2NQT}$, formally:

$$\sum_{t=1}^{T} \mathbb{P}\left(\frac{\sum_{j=1}^{N} v_j \; n_j(\hat{x}_{j,t})}{\sum_{j=1}^{N} \; c_j(\hat{x}_{j,t})} < \Lambda \vee \sum_{j=1}^{N} c_j(\hat{x}_{j,t}) > \beta\right) \geq T - \frac{\delta}{2NQT}.$$

This concludes the proof. □

8.4.2 Guaranteeing Safety: $\mathsf{GCB}_{\mathtt{safe}}$

We propose $\mathsf{GCB}_{\mathtt{safe}}$, a variant of GCB relying on different values to be used in the vector $\boldsymbol{\mu}$. More specifically, we employ optimistic estimates for the parameters used in the objective function and pessimistic estimates for the parameters used in the constraints. Formally, in $\mathsf{GCB}_{\mathtt{safe}}$, we set:

$$\begin{aligned}
\overline{w}_j(x) &:= v_j \left[\hat{n}_{j,t-1}(x) + \sqrt{b_{t-1}}\hat{\sigma}^n_{j,t-1}(x)\right], \\
\underline{w}_j(x) &:= v_j \left[\hat{n}_{j,t-1}(x) - \sqrt{b_{t-1}}\hat{\sigma}^n_{j,t-1}(x)\right], \\
\overline{c}_j(x) &:= \hat{c}_{j,t-1}(x) + \sqrt{b_{t-1}}\hat{\sigma}^c_{j,t-1}(x).
\end{aligned}$$

Furthermore, $\mathsf{GCB}_{\mathtt{safe}}$ needs a default set of bids $\left\{x^{\mathsf{d}}_{j,t}\right\}_{j=1}^{N}$, that is known *a priori* to be feasible for the problem in Equations (8.1a)–(8.1c) with the actual values of the parameters.[7] The pseudo-code of $\mathsf{GCB}_{\mathtt{safe}}$ is provided in Algorithm 8.1 with the above definition of the parameters of vector $\boldsymbol{\mu}$, except that it returns $\{\hat{x}_{j,t}\}_{j=1}^{N} = \left\{x^{\mathsf{d}}_{j,t}\right\}_{j=1}^{N}$ if the optimization problem does not admit any feasible solution with the current estimates. We can show the following.

[7] A trivial default feasible bid allocation is $\left\{x^{\mathsf{d}}_{j,t} = 0\right\}_{j=1}^{N}$.

Theorem 8.6 ($\mathsf{GCB}_{\mathtt{safe}}$ safety)**.** *Given $\delta \in (0,\ 1)$, $\mathit{GCB}_{\mathtt{safe}}$ applied to the problem in Equations* (8.1a)–(8.1c) *is δ-safe and, therefore, the number of constraints violations is constant in T.*

Proof. Let us focus on a specific day t. Constraints (8.1b) and (8.1c) are satisfied by the solution of $\mathsf{Opt}(\boldsymbol{\mu}, \Lambda)$ for the properties of the optimization procedure. Define $\underline{n}_j(x_{j,t}) := \hat{n}_j(x_{j,t}) - \sqrt{b_{t-1}}\hat{\sigma}^n_j(x_{j,t})$. Thanks to the specific construction of the upper bounds, we have that $c_j(x_{j,t}) \leq \overline{c}_j(x_{j,t})$ and $n_j(x_{j,t}) \geq \underline{n}_j(x_{j,t})$, each holding with probability at least $1 - \frac{3\delta}{\pi^2 NQTt^2}$. Therefore, we have:

$$\frac{\sum_{j=1}^N v_j\ n_j(x_{j,t})}{\sum_{j=1}^N\ c_j(x_{j,t})} > \frac{\sum_{j=1}^N v_j\ \underline{n}_j(x_{j,t})}{\sum_{j=1}^N\ \overline{c}_j(x_{j,t})} \geq \Lambda$$

and

$$\sum_{j=1}^N c_j(x_{j,t}) < \sum_{j=1}^N \overline{c}_j(x_{j,t}) \leq \beta.$$

Using a union bound over:

- the two GPs (number of clicks and costs);
- the time horizon T;
- the number of times each bid is chosen in a subcampaign (at most t);
- the number of arms present in each subcampaign ($|X_j|$);
- the number of subcampaigns (N);

we have:

$$\sum_{t=1}^T \mathbb{P}\left(\frac{\sum_{j=1}^N v_j\ n_j(\hat{x}_{j,t})}{\sum_{j=1}^N\ c_j(\hat{x}_{j,t})} < \Lambda \vee \sum_{j=1}^N c_j(\hat{x}_{j,t}) > \beta\right) \leq$$

$$\leq 2\sum_{j=1}^N \sum_{k=1}^{|X_j|} \sum_{h=1}^T \sum_{l=1}^t \frac{3\delta}{\pi^2 NQTl^2} \leq$$

$$\leq 2\sum_{j=1}^N \sum_{k=1}^Q \sum_{h=1}^T \sum_{l=1}^{+\infty} \frac{3\delta}{\pi^2 NQTl^2} = \delta.$$

This concludes the proof. □

The safety property comes at the cost that $\mathsf{GCB}_{\mathtt{safe}}$ may suffer from a much larger pseudo-regret than GCB as stated by the following theorem.

Theorem 8.7 ($\mathsf{GCB}_{\mathtt{safe}}$ pseudo-regret)**.** *Given* $\delta \in (0,\ 1)$*,* $\mathit{GCB}_{\mathtt{safe}}$ *applied to the problem in Equations* (8.1a)–(8.1c) *suffers from a pseudo-regret* $R^T(\mathit{GCB}_{\mathtt{safe}}) = \Theta(T)$.

Proof. At the optimal solution, at least one of the constraints is active, *i.e.,* it has the left-hand side equal to the right-hand side. Assume that the optimal clairvoyant solution $\left\{x_j^*\right\}_{j=1}^N$ to the optimization problem has a value of the ROI Λ_{opt} equal to Λ. We showed in the proof of Theorem 8.6 that for any allocation, with probability at least $1 - \frac{3\delta}{\pi^2 NQTt^2}$, it holds that $\frac{\sum_{j=1}^N v_j\ n_j(x_{j,t})}{\sum_{j=1}^N\ c_j(x_{j,t})} > \frac{\sum_{j=1}^N v_j\ \underline{n}_j(x_{j,t})}{\sum_{j=1}^N\ \overline{c}_j(x_{j,t})}$. This is true also for the optimal clairvoyant solution $\left\{x_j^*\right\}_{j=1}^N$, for which $\Lambda = \frac{\sum_{j=1}^N v_j\ n_j(x^*)}{\sum_{j=1}^N\ c_j(x^*)} > \frac{\sum_{j=1}^N v_j\ \underline{n}_j(x^*)}{\sum_{j=1}^N\ \overline{c}_j(x^*)}$, implying that the values used in the ROI constraint make this allocation not feasible for the $\mathsf{Opt}(\boldsymbol{\mu}, \Lambda)$ procedure. As shown before, this happens with probability at least $1 - \frac{3\delta}{\pi^2 NQTt^2}$ at day t, and $1-\delta$ over the time horizon T. To conclude, with probability $1-\delta$, not depending on the time horizon T, we will not choose the optimal arm during the time horizon and, therefore, the regret of the algorithm cannot be sublinear. Notice that the same line of proof is also holding in the case the budget constraint is active, therefore, the previous result holds for each instance of the problem in Equations (8.1a)–(8.1c). □

8.4.3 Guaranteeing Sublinear Pseudo-regret and Safety with Tolerance: $\mathsf{GCB}_{\mathtt{safe}}(\psi, \phi)$

In what follows, we show that, when a tolerance in the violation of the constraints is accepted, an adaptation of $\mathsf{GCB}_{\mathtt{safe}}$ can be exploited to obtain a sublinear pseudo-regret. Given an instance of the problem in Equations (8.1a)–(8.1c) that we call *original problem*, we build an *auxiliary problem* in which we slightly relax the ROI and budget constraints. Formally, the $\mathsf{GCB}_{\mathtt{safe}}(\psi, \phi)$ is the $\mathsf{GCB}_{\mathtt{safe}}$ applied to the auxiliary problem in which the parameters Λ and β have been substituted with $\Lambda - \psi$ and $\beta + \phi$, respectively. Thanks to the results provided in Section 8.4.2, $\mathsf{GCB}_{\mathtt{safe}}(\psi, \phi)$, w.h.p., does not violate the ROI constraint of the original problem by more than the tolerance ψ and the budget constraint of the original problem by more than the tolerance ϕ. In the following, we distinguish three cases depending on the *a priori* information available to the advertisers. Indeed, the advertisers may know that a constrain is not active at the optimal solution of the given instance thanks to the observation of old data. In these cases, we just need a milder relaxation of the problem than in the general case in which the advertisers has no *a priori* information. At first, we focus on the

case in which we *a priori* know that the budget constraint is not active at the optimal solution. We show the following:

Theorem 8.8 ($\mathsf{GCB}_{\mathtt{safe}}(\psi, 0)$ pseudo-regret and safety with tolerance)**.** *When:*

$$\psi \geq 2\frac{\beta_{opt} + n_{\max}}{\beta_{opt}^2} \sum_{j=1}^{N} v_j \sqrt{2 \ln \left(\frac{\pi^2 N Q T^3}{3\delta'} \right)} \sigma$$

and

$$\beta_{opt} < \beta \frac{\sum_{j=1}^{N} v_j}{\frac{N\ \beta_{opt}\psi}{\beta_{opt}+n_{\max}} + \sum_{j=1}^{N} v_j},$$

where $\delta' \leq \delta$, β_{opt} is the spend at the optimal solution of the original problem, and $n_{\max} := \max_{j,x} n_j(x)$ is the maximum over the sub-campaigns and the admissible bids of the expected number of clicks, $\mathsf{GCB}_{\mathtt{safe}}(\psi, 0)$ provides a pseudo-regret w.r.t. the optimal solution to the original problem of $\mathcal{O}\left(\sqrt{T \sum_{j=1}^{N} \gamma_{j,T}}\right)$ with probability at least $1 - \delta - \frac{\delta'}{QT^2}$, while being δ-safe w.r.t. the constraints of the auxiliary problem.

The above result states that, if we allow a violation of at most ψ of the ROI constraint, the result provided in Theorem 8.1 can be circumvented for a class of instances of the optimization problem. In this case, $\mathsf{GCB}_{\mathtt{safe}}(\psi, 0)$ guarantees sublinear regret and a number of constraints violations that is constant in T.

Notice that the magnitude of the violation ψ increases linearly in the maximum number of clicks n_{max} and $\sum_{j=1}^{N} v_j$, that, in its turn, increases linearly in the number of sub-campaigns N. This suggests that in large instances this value may be large. However, in practice, the maximum number of clicks of a sub-campaign $n_{\max}$ is a sublinear function in the optimal budget β_{opt}, and usually it goes to a constant as the budget spent goes to infinity. Moreover, the number of sub-campaigns N usually depends on the budget, *i.e.*, the budget planned by the business units is linear in the number of sub-campaigns. As a result, β_{opt} is of the same order of $\sum_{j=1}^{N} v_j$, and therefore, since n_{max} is sublinear in β_{opt} and $\sum_{j=1}^{N} v_j$ is of the order of β_{opt}, the final expression of ψ is sub-linear in β_{opt}. This means that the lower bound to ψ to satisfy the assumption needed by Theorem 8.8 goes to zero as β_{opt} increases.

We can derive a similar result in the case in which we *a priori* know that the ROI constraint is not active at the optimal solution. In particular, we state the following.

Theorem 8.9 ($\mathsf{GCB}_{\mathtt{safe}}(0,\phi)$ pseudo-regret and safety with tolerance)**.** *When*

$$\phi \geq 2N\sqrt{2\ln\left(\frac{\pi^2 NQT^3}{3\delta'}\right)}\sigma$$

and

$$\Lambda_{opt} > \Lambda + \frac{(\beta + n_{\max})\phi\sum_{j=1}^{N} v_j}{N\beta^2},$$

where $\delta' \leq \delta$, and $n_{\max} := \max_{j,x} n_j(x)$ is maximum expected number of clicks, $\mathsf{GCB}_{\mathtt{safe}}(0,\phi)$ provides a pseudo-regret w.r.t. the optimal solution to the original problem of $\mathcal{O}\left(\sqrt{T\sum_{j=1}^{N}\gamma_{j,T}}\right)$ with probability at least $1-\delta-\frac{6\delta'}{\pi^2 QT^2}$, while being δ-safe w.r.t. the constraints of the auxiliary problem.

Proof. We show that at a specific day t since the optimal solution of the original problem $\left\{x_j^*\right\}_{j=1}^{N}$ is included in the set of feasible ones, we are in a setting analogous to the one of GCB, in which the regret is sublinear. Let us assume that the upper bounds to all the quantities (number of clicks and costs) holds. This has been shown before to occur with overall probability δ over the whole time horizon T.

First, let us evaluate the probability that the optimal solution is not feasible. This occurs if its bounds are either violating the ROI or budget constraints. From the fact that the ROI of the optimal solution satisfies $\Lambda_{opt} > \Lambda + \frac{(\beta+n_{\max})\phi\sum_{j=1}^{N} v_j}{N\beta^2}$, we have:

$$\begin{aligned}
&\mathbb{P}\left(\frac{\sum_{j=1}^{N} v_j\ \underline{n}_j(x_j^*)}{\sum_{j=1}^{N}\ \overline{c}_j(x_j^*)} < \Lambda\right)\\
&\leq \mathbb{P}\left(\frac{\sum_{j=1}^{N} v_j\ \underline{n}_j(x_j^*)}{\sum_{j=1}^{N}\ \overline{c}_j(x_j^*)} < \Lambda_{opt} - \frac{(\beta+n_{\max})\phi\sum_{j=1}^{N} v_j}{N\beta^2}\right)\\
&= \mathbb{P}\Bigg(\frac{\sum_{j=1}^{N} v_j\ \underline{n}_j(x_j^*)}{\sum_{j=1}^{N}\ \overline{c}_j(x_j^*)} <\\
&< \frac{\sum_{j=1}^{N} v_j\ n_j(x_j^*)}{\sum_{j=1}^{N}\ c_j(x_j^*)} - 2\frac{\beta_{opt}+n_{\max}}{\beta_{opt}^2}\sum_{j=1}^{N} v_j\sqrt{\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\Bigg)\\
&\leq \frac{3\delta'}{\pi^2 QT^3},
\end{aligned}$$

where the derivation uses arguments similar to the ones applied in the proof for the ROI constraint in Theorem 8.8. Summing over the time horizon T ensures that the optimal solution of the original problem $\left\{x_j^*\right\}_{j=1}^N$ is excluded from the feasible solutions at most with probability $\frac{3\delta'}{\pi^2 QT^2}$.

Second, let us evaluate the probability for which the optimal solution of the original problem $\left\{x_j^*\right\}_{j=1}^N$ is excluded due to the budget constraint, formally:

$$
\begin{aligned}
&\mathbb{P}\left(\sum_{j=1}^{N}\overline{c}_j(x_j^*) > \beta + \phi\right) \\
&\leq \mathbb{P}\left(\sum_{j=1}^{N}\overline{c}_j(x_j^*) > \beta + 2N\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\right) \\
&= \mathbb{P}\left(\sum_{j=1}^{N}\overline{c}_j(x_j^*) > \sum_{j=1}^{N} c_j(x_j^*) + 2N\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\right) \\
&\leq \sum_{j=1}^{N}\mathbb{P}\left(\overline{c}_j(x_j^*) > c_j(x_j^*) + 2\sqrt{\ln\frac{12NT^3}{\pi^2\delta'}}\sigma\right) \qquad (8.6) \\
&= \sum_{j=1}^{N}\mathbb{P}\left(\hat{c}_{j,t-1}(x_j^*) - c_j(x_j^*) \geq -\sqrt{b_t}\hat{\sigma}^c_{j,t-1}(x_j^*) + 2\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\right) \\
&\leq \sum_{j=1}^{N}\mathbb{P}\left(\hat{c}_{j,t-1}(x_j^*) - c_j(x_j^*) \geq \sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\hat{\sigma}^c_{j,t-1}(x_j^*)\right) \\
&\leq \sum_{j=1}^{N}\mathbb{P}\left(\frac{\hat{c}_{j,t-1}(x_j^*) - c_j(x_j^*)}{\hat{\sigma}^c_{j,t-1}(x_j^*)} \geq \sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\right) \\
&\leq \sum_{j=1}^{N}\frac{3\delta'}{\pi^2 NQT^3} = \frac{3\delta'}{\pi^2 QT^3}, \qquad (8.7)
\end{aligned}
$$

where we use the fact that $\beta = \beta_{opt}$, and the derivation uses arguments similar to the ones applied in the proof for the budget constraint in Theorem 8.8. Summing over the time horizon T, we get that the optimal solution of the original problem $\left\{x_j^*\right\}_{j=1}^N$ is excluded from the set of the feasible ones with probability at most $\frac{\pi^2\delta'}{6T^2}$. Finally, using a union bound, we have that the optimal solution can be chosen over the time horizon with probability at least $1 - \frac{3\delta'}{\pi^2 QT^2}$.

Notice that here we want to compute the regret of the $\mathsf{GCB}_{\texttt{safe}}$ algorithm w.r.t. $\left\{x_j^*\right\}_{j=1}^N$ which is not optimal for the analysed relaxed problem. Nonetheless, the proof on the pseudo-regret provided in Theorem 8.4 is valid also for suboptimal solutions in the case it is feasible with high probability. This can be trivially shown using the fact that the regret w.r.t. a generic solution cannot be larger than the one computed on the optimal one. Thanks to that, using a union bound over the probability that the bounds hold and that $\left\{x_j^*\right\}_{j=1}^N$ is feasible, we conclude that with probability at least $1-\delta-\frac{6\delta'}{\pi^2 QT^2}$ the regret $\mathsf{GCB}_{\texttt{safe}}$ is of the order of $\mathcal{O}\left(\sqrt{T\sum_{j=1}^N \gamma_{j,T}}\right)$. Finally, thanks to the property of the $\mathsf{GCB}_{\texttt{safe}}$ algorithm shown in Theorem 8.6, the learning policy is δ-safe for the relaxed problem. □

On the satisfaction of the assumption needed by the above theorem, we can produce a consideration similar to that done before for the case in which the budget constraint is not active at the optimal solution. As previously, the lower bound to ϕ to satisfy the assumption needed by Theorem 8.9 goes to zero as β_{opt} increases.

Finally, we focus on the case in which the advertiser has no information on which constrain is active. In this case, we can state the following which generalizes the two results provided above.

Theorem 8.10 ($\mathsf{GCB}_{\texttt{safe}}(\psi,\phi)$ pseudo-regret and safety with tolerance)**.** *Setting*

$$\psi = 2\frac{\beta_{opt}+n_{\max}}{\beta_{opt}^2}\sum_{j=1}^N v_j\sqrt{2\ln\left(\frac{\pi^2 NQT^3}{3\delta'}\right)}\sigma$$

and

$$\phi = 2N\sqrt{2\ln\left(\frac{\pi^2 NQT^3}{3\delta'}\right)}\sigma,$$

where $\delta' \leq \delta$, $\mathsf{GCB}_{\texttt{safe}}(\psi,\phi)$ provides a pseudo-regret w.r.t. the optimal solution to the original problem of $\mathcal{O}\left(\sqrt{T\sum_{j=1}^N \gamma_{j,T}}\right)$ with probability at least $1-\delta-\frac{\delta'}{QT^2}$, while being δ-safe w.r.t. the constraints of the auxiliary problem.

Proof. The proof follows from combining the arguments about the ROI constraint used in Theorem 8.8 and those about the budget constraint used in Theorem 8.9. □

8.5 Experimental Evaluation

We experimentally evaluate our algorithms in terms of pseudo-regret and safety in synthetic settings generated from real-world data. The adoption of synthetic settings allows us to evaluate our algorithms in many different realistic scenarios and, for each of them, to find the optimal clairvoyant solution necessary to measure the algorithms' regret and safety. The real-world dataset is provided by AdsHotel (`https://www.adshotel.com/`), an Italian media agency working in the hotel booking market.[8]

Our experimental activity is structured as follows. In Section 8.5.1, we evaluate how GCB and $\text{GCB}_{\texttt{safe}}$ violate the constraints. In Section 8.5.2, we evaluate how the performances of $\text{GCB}_{\texttt{safe}}(\psi, \phi)$ vary as ψ varies when the budget constraint is known to be active. In Section 8.5.3, we evaluate the performances of all the algorithms when the ROI constraint is known to be active. Finally, in Section 8.5.4, we run our algorithms with multiple, heterogeneous settings and evaluate the average performances.

8.5.1 Experiment #1: evaluating constraint violation with GCB and $\text{GCB}_{\texttt{safe}}$

In this experiment, we show that GCB suffers from significant violations of both ROI and budget constraints even in simple settings, while $\text{GCB}_{\texttt{safe}}$ does not.

Setting We simulate $N = 5$ subcampaigns, with $|X_j| = 201$ bid values evenly spaced in $[0,\ 2]$, $|Y| = 101$ cost values evenly spaced in $[0,\ 100]$, and $|R| = 151$ revenue values evenly spaced in $[0,\ 1200]$. For a generic subcampaign C_j, at every t, the daily number of clicks is returned by function $\tilde{n}_j(x) := \theta_j(1 - e^{-x/\delta_j}) + \xi_j^n$ and the daily cost by function $\tilde{c}_j(x) = \alpha_j(1 - e^{-x/\gamma_j}) + \xi_j^c$, where $\theta_j \in \mathbb{R}^+$ and $\alpha_j \in \mathbb{R}^+$ represent the maximum achievable number of clicks and cost for subcampaign C_j in a single day, $\delta_j \in \mathbb{R}^+$ and $\gamma_j \in \mathbb{R}^+$ characterize how fast the two functions reach a saturation point, and ξ_j^n and ξ_j^c are noise terms drawn from a $\mathcal{N}(0, 1)$ Gaussian distribution (these functions are customarily used in the advertising literature, *e.g.,* by Kong et al. (2018)). We assume a unitary value for each click, *i.e.,* $v_j = 1$ for each $j \in \{1, \dots, N\}$. The values of the parameters of cost and revenue functions of the subcampaigns are specified in Table C.1 reported in C.1.2. We set a daily budget $\beta = 100$, $\Lambda = 10$ in the ROI constraint, and a time horizon

[8] Additional details useful for the complete reproducibility of our results are provided in Appendix C.1.2, while the code used is available at: `https://github.com/oi-tech/safe_bid_opt`.

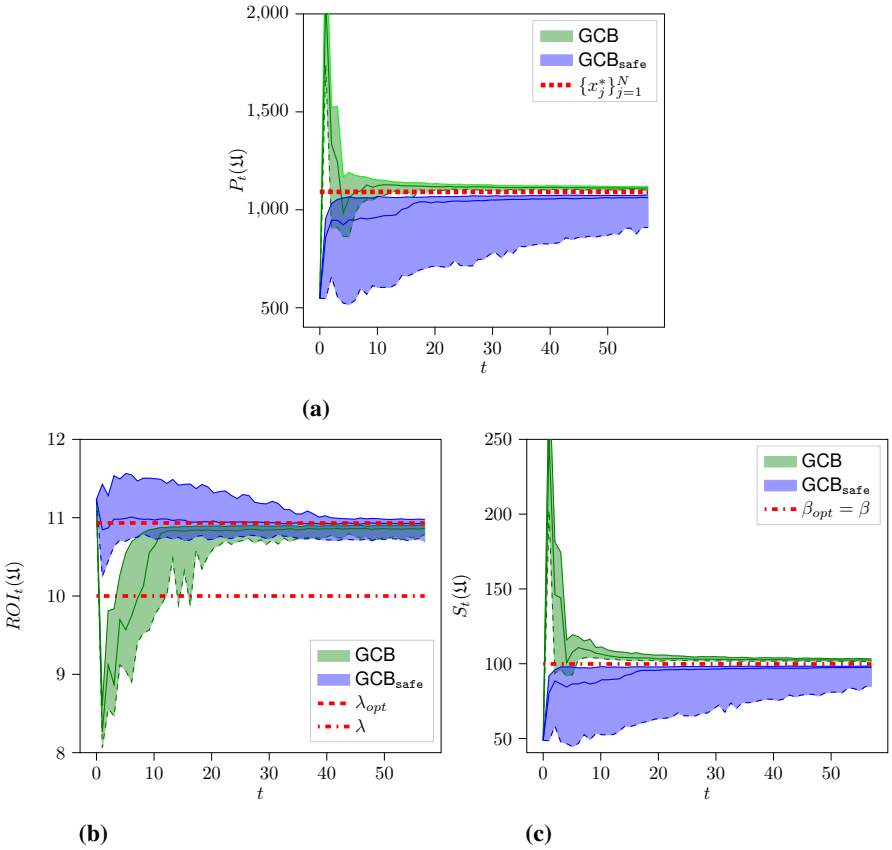


Figure 8.1: *Results of Experiment #1: daily revenue (a), ROI (b), and spend (c) obtained by GCB and GCB$_{\texttt{safe}}$. Dashed lines correspond to the optimal values for the revenue and ROI, while dash-dotted lines correspond to the values of the ROI and budget constraints.*

$T = 60$. The peculiarity of this setting is that, at the optimal solution, the budget constraint is active, while the ROI constraint is not.

For both GCB and GCB$_{\texttt{safe}}$, the kernels for the number of clicks GPs $k(x, x')$ and for the costs GPs $h_j(x, x')$ are squared exponential kernels of the form $\sigma_f^2 \exp\left\{-\frac{(x-x')^2}{l}\right\}$ for every $x, x' \in X_j$, where the parameters $\sigma_f \in \mathbb{R}^+$ and $l \in \mathbb{R}^+$ are estimated from data, as suggested by Rasmussen and Williams (2006). The confidence for the algorithms is $\delta = 0.2$.

Results We evaluate the algorithms in terms of:

- daily revenue: $P_t(\mathfrak{U}) := \sum_{j=1}^N v_j n_j(\hat{x}_{j,t})$;

- daily ROI: $ROI_t(\mathfrak{U}) := \frac{\sum_{j=1}^{N} v_j \; n_j(\hat{x}_{j,t})}{\sum_{j=1}^{N} \; c_j(\hat{x}_{j,t})}$;
- daily spend: $S_t(\mathfrak{U}) := \sum_{j=1}^{N} c_j(\hat{x}_{j,t})$.

We perform 100 independent runs for each algorithm.

In Figure 8.1, for the daily revenue, ROI, and spend achieved by GCB and $\mathsf{GCB}_{\mathtt{safe}}$ at every t, we show the 50_{th} percentile (*i.e.*, the median) with solid lines and the 90_{th} and 10_{th} percentiles with dashed lines surrounding the semi-transparent area. While GCB achieves a larger revenue than $\mathsf{GCB}_{\mathtt{safe}}$, it violates the budget constraint over the entire time horizon and the ROI constraint in the first 7 days in more than 50% of the runs. This happens because, in the optimal solution, the ROI constraint is not active, while the budget constraint is. Conversely, $\mathsf{GCB}_{\mathtt{safe}}$ satisfies the budget and ROI constraints over the time horizon for more than 90% of the runs, and has a slower convergence to the optimal revenue. If we focus on the median revenue, $\mathsf{GCB}_{\mathtt{safe}}$ has a similar behaviour to that of GCB for $t > 15$. This makes $\mathsf{GCB}_{\mathtt{safe}}$ a good choice even in terms of overall revenue. However, it is worth to notice that, in the 10% of the runs, $\mathsf{GCB}_{\mathtt{safe}}$ does not converge to the optimal solution before the end of the learning period. These results confirm our theoretical analysis showing that limiting the exploration to safe regions might lead the algorithm to get a large regret. Furthermore, let us remark that the learning dynamics of $\mathsf{GCB}_{\mathtt{safe}}$ are much smoother than those of GCB, which present, instead, oscillations.

8.5.2 Experiment #2: evaluating $\mathsf{GCB}_{\mathtt{safe}}(\psi, 0)$ when the budget constraint is active

In real-world scenarios, the business goals in terms of volumes-profitability tradeoff are often blurred, and sometimes it can be desirable to slightly violate the constraints (usually, the ROI constraint) in favor of a significant volume increase. However, analyzing and acquiring information about these tradeoff curves requires exploring volumes opportunities by relaxing the constraints. In this experiment, we show how our approach can be adjusted to address this problem in practice.

Setting We use the same setting of Experiment #1, except that we evaluate $\mathsf{GCB}_{\mathtt{safe}}$ and $\mathsf{GCB}_{\mathtt{safe}}(\psi, \phi)$ algorithms. More precisely, we relax the ROI constraint by a tolerance $\psi \in \{0, 0.05, 0.1, 0.15\}$ (while keeping $\phi = 0$). Notice that $\mathsf{GCB}_{\mathtt{safe}}(0, 0)$ corresponds to the use of $\mathsf{GCB}_{\mathtt{safe}}$ in the original problem. As a result, except for the case $\phi = 0$, we allow $\mathsf{GCB}_{\mathtt{safe}}(\psi, \phi)$ to violate the ROI constraint, but, with high probability, the violation is

bounded by at most 0.5%, 1%, 1.5% of Λ, respectively. Instead, we do not introduce any tolerance for the daily budget constraint β.

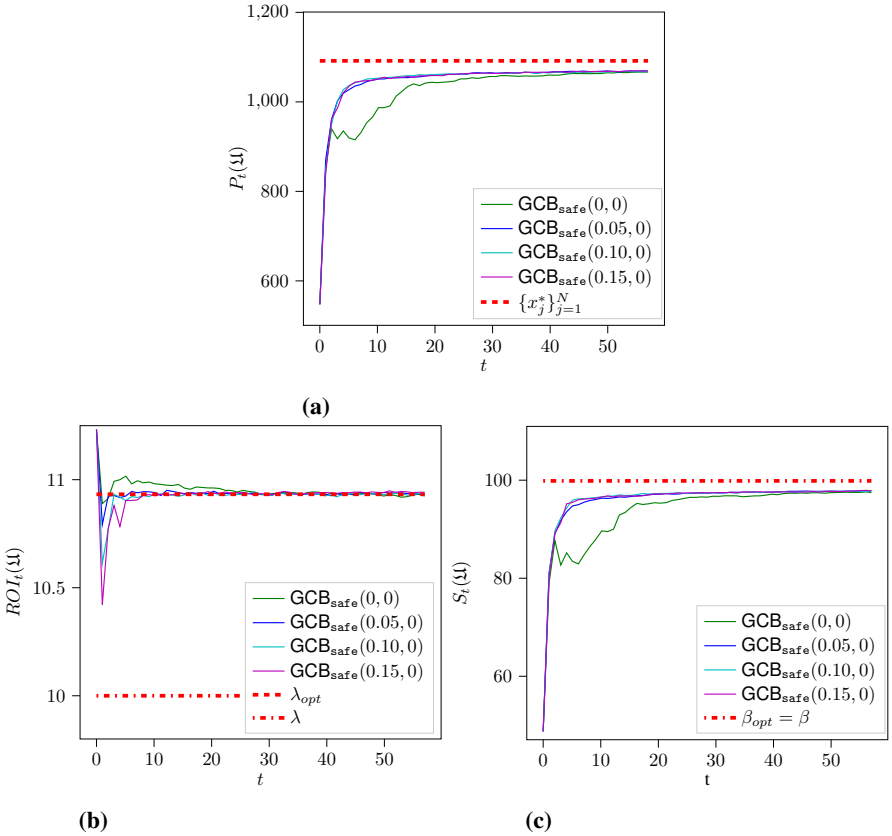


Figure 8.2: *Results of Experiment #2: Median values of the daily revenue (a), ROI (b) and spend (c) obtained by* $\mathsf{GCB}_{\mathtt{safe}}(\psi, 0)$ *with different values of* ψ.

Results In Figures 8.2, we show the median values, on 100 independent runs, of the performance in terms of daily revenue, ROI, and spend of $\mathsf{GCB}_{\mathtt{safe}}(\psi, 0)$ for every value of ψ. The 10% and 90% quantiles are reported in Figure C.2, C.3, C.4, and C.5 in C.1.2. The results show that allowing a small tolerance in the ROI constraint violation, we can improve the exploration and, therefore, lead to faster convergence. We note that if we set a value of $\psi \geq 0.05$, we achieve significantly better performance in the first learning steps ($t < 20$) still maintaining a robust behavior in terms of constraints violation. Most importantly, the ROI constraint is always satisfied by the median and also by the 10% and 90% quantiles. Furthermore, the few violations are concentrated in the early stages of the learning process.

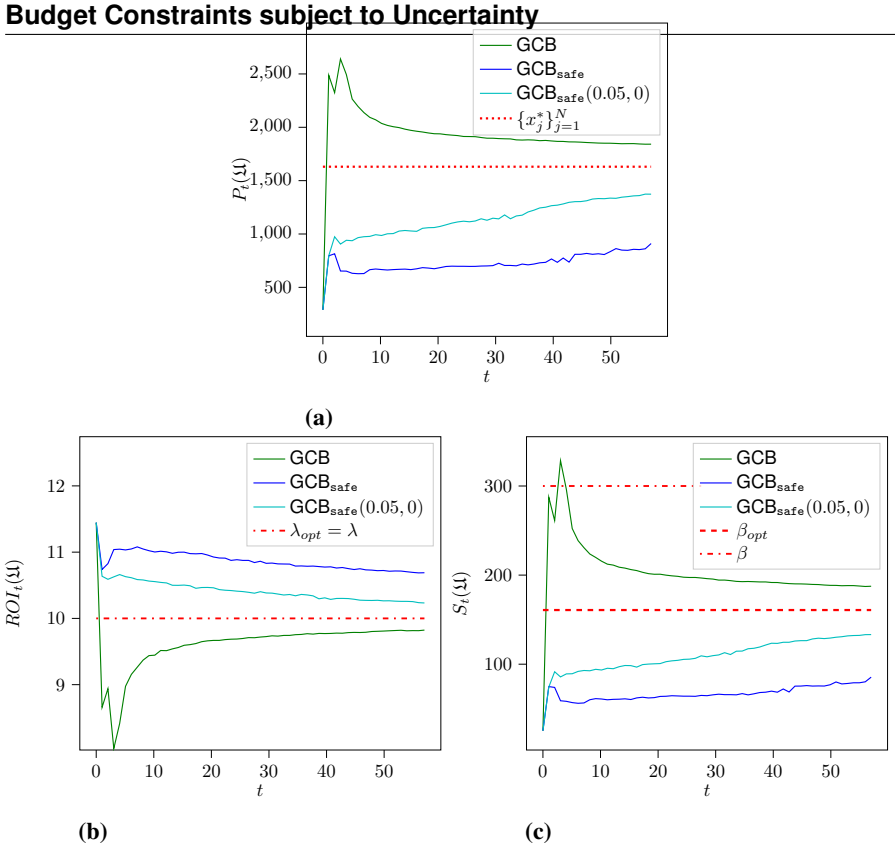


Figure 8.3: *Results of Experiment #3: Median values of the daily revenue (a), ROI (b) and spend (c) of GCB, $\text{GCB}_{\texttt{safe}}$, and $\text{GCB}_{\texttt{safe}}(0.05, 0)$.*

8.5.3 Experiment #3: comparing GCB, $\text{GCB}_{\texttt{safe}}$, and $\text{GCB}_{\texttt{safe}}(\psi, 0)$ when the ROI constraint is active

We study a setting in which the ROI constraint is active at the optimal solution, *i.e.*, $\Lambda = \Lambda_{opt}$, while the budget constraint is not. This means that, at the optimal solution, the advertiser would have an extra budget to spend. However, such budget is not spent, the ROI constraint would be violated otherwise.

Setting The experimental setting is the same of Experiment #1, except that we set the budget constraint as $\beta = 300$. The optimal daily spend is $\beta_{opt} = 161$.

Results In Figure 8.3, we show the median values of the daily revenue, the ROI, and the spend of GCB, $\text{GCB}_{\texttt{safe}}$, $\text{GCB}_{\texttt{safe}}(0.05, 0)$ obtained with 100 independent runs. The 10% and 90% of the quantities provided by GCB,

$\mathsf{GCB}_{\texttt{safe}}$, and $\mathsf{GCB}_{\texttt{safe}}(0.05, 0)$ are reported in Figures C.6, C.7, and C.8 in C.1.2. We notice that, even in this setting, GCB violates the ROI constraint for the entire time horizon, and the budget constraint in $t = 6$ and $t = 7$. However, it achieves a revenue larger than that of the optimal constrained solution. On the other side, $\mathsf{GCB}_{\texttt{safe}}$ and always satisfies both the constraints, but it does not perform enough exploration to quickly converge to the optimal solution. We observe that it is sufficient to allow a tolerance in the ROI constraint violation by slightly perturbing the input value Λ ($\psi = 0.05$, corresponding to a violation of the constraint by at most 0.5%) to make $\mathsf{GCB}_{\texttt{safe}}(\psi, \phi)$ capable of approaching the optimal solution while satisfying both constraints for every $t \in \{0, \dots, T\}$. This suggests that, in real-world applications, $\mathsf{GCB}_{\texttt{safe}}(\psi, \phi)$ with a small tolerance represents an effective solution, providing guarantees on the violation of the constraints while returning high values of revenue.

8.5.4 Experiment #4: comparing GCB, $\mathsf{GCB}_{\texttt{safe}}$, and $\mathsf{GCB}_{\texttt{safe}}(\psi, \phi)$ with multiple, heterogeneous settings

In this experiment we extend the experimental activity we conduct in Experiments #1 and #3 to other multiple, heterogeneous settings.

Setting We simulate $N = 5$ subcampaigns with a daily budget $\beta = 100$, with $|X_j| = 201$ bid values evenly spaced in $[0,\ 2]$, $|Y| = 101$ cost values evenly spaced in $[0,\ 100]$, being the daily budget $\beta = 100$, and $|R|$ evenly spaced revenue values depending on the setting. We generate 10 scenarios that differ in the parameters defining the cost and revenue functions, and in the ROI parameter Λ. Recall that the number-of-click functions coincide with the revenue functions since $v_j = 1$ for each $j \in \{1, \dots, N\}$. Parameters $\alpha_j \in \mathbb{N}^+$ and $\theta_j \in \mathbb{N}^+$ are sampled from discrete uniform distributions $\mathcal{U}\{50, 100\}$ and $\mathcal{U}\{400, 700\}$, respectively. Parameters γ_j and δ_j are sampled from the continuous uniform distributions $\mathcal{U}[0.2, 1.1)$. Finally, parameters Λ are chosen such that the ROI constraint is active at the optimal solution. Table C.1 in C.1.2 specifies the values of such parameters.

Results We compare the algorithms GCB, $\mathsf{GCB}_{\texttt{safe}}$, $\mathsf{GCB}_{\texttt{safe}}(0.05, 0)$, and $\mathsf{GCB}_{\texttt{safe}}(0.10, 0)$ in terms of:

- $W_t := \sum_{h=1}^{t} P_t(\mathfrak{U})$: average cumulative revenue at round t (and the corresponding standard deviation σ_t);
- M_t: median of the cumulative revenue at round t;

		W_T	$W_{T/2}$	σ_T	$\sigma_{T/2}$	M_T	$M_{T/2}$	U_T	$U_{T/2}$	L_T	$L_{T/2}$	V_{ROI}	V_B
Setting #1	GCB	57481	30767	556	376	57497	30811	58081	31239	56758	30288	1.00	0.62
	GCB_{safe}	44419	21549	4766	2474	45348	21972	46783	23163	42287	20324	0.02	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	48028	23524	4902	2487	48626	23831	50388	24827	46307	22506	0.21	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	52327	25859	829	611	52338	25887	53324	26605	51316	25104	0.94	0.00
Setting #2	GCB	63664	35566	1049	679	63701	35573	64984	36524	62249	34675	1.00	0.14
	GCB_{safe}	34675	16290	8541	4448	37028	17647	39594	19473	27748	11141	0.03	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	40962	19564	6013	3122	41823	20152	44468	21698	38640	17645	0.04	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	46694	22099	6382	3112	47749	22433	51564	24776	44099	19929	0.72	0.00
Setting #3	GCB	54845	30213	757	478	54816	30177	55734	30885	54006	29638	1.00	0.25
	GCB_{safe}	35726	16577	8239	4361	38302	18114	40746	19882	27279	8791	0.03	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	38757	18370	8492	4594	41422	19808	43337	21092	30413	12678	0.07	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	42184	19993	9652	5056	44820	21574	47659	23118	36570	14450	0.75	0.00
Setting #4	GCB	71404	37383	351	262	71399	37387	71877	37732	70930	37021	0.98	0.98
	GCB_{safe}	29101	13817	7052	3646	30992	14680	35602	17256	20509	9562	0.00	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	39802	18270	10232	4955	38296	17994	53375	24962	25197	11341	0.01	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	51515	24095	11094	5639	56621	24902	61992	30020	35642	16198	0.56	0.00
Setting #5	GCB	74638	39523	642	392	74693	39529	75405	40049	73756	39063	0.98	0.31
	GCB_{safe}	48956	23230	6715	3486	50021	23838	53644	26266	42946	19287	0.00	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	56205	27003	2578	1742	56554	27211	58839	28802	53278	24987	0.00	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	63411	30207	5636	2916	64364	30665	66764	32212	60519	28260	0.59	0.00
Setting #6	GCB	67118	35775	327	260	67130	35795	67536	36111	66726	35424	0.98	0.98
	GCB_{safe}	14448	7707	6006	3065	15019	8075	18581	9800	6781	3926	0.02	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	14968	7710	6174	2974	15161	8157	20548	10351	7954	3860	0.02	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	34716	15507	16133	7280	37409	16601	55236	25366	9895	5188	0.19	0.00
Setting #7	GCB	63038	35330	873	401	63088	35367	64226	35793	61754	34823	1.00	0.41
	GCB_{safe}	31662	14806	5651	3090	33009	15570	35004	16922	28296	11338	0.04	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	37744	17606	4173	2619	38321	18161	41184	19805	33914	15276	0.03	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	42528	20046	7497	3624	43765	20683	47187	22301	38988	18314	0.70	0.00
Setting #8	GCB	79571	42322	476	375	79581	42317	80073	42743	78969	41913	1.00	0.98
	GCB_{safe}	48046	22478	11779	6000	52094	24180	57321	28024	30655	13338	0.02	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	58450	27477	10296	5605	61404	28845	66902	32883	41196	18222	0.02	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	68252	33255	3436	2417	68886	33857	70758	35377	65394	30696	0.07	0.00
Setting #9	GCB	70280	37363	672	347	70275	37352	71123	37811	69379	36942	1.00	0.34
	GCB_{safe}	40116	18895	5522	3047	40673	19357	43850	21161	37310	17222	0.03	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	51138	23683	3110	2036	50984	23375	54545	26174	47465	21385	0.03	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	63574	29675	3810	3323	64011	30112	66658	32559	60970	27280	0.80	0.00
Setting #10	GCB	80570	41973	435	344	80568	42019	81127	42388	80023	41496	1.00	0.98
	GCB_{safe}	58965	28785	3097	1465	60033	28917	62353	30535	54590	26931	0.02	0.00
	$\text{GCB}_{\text{safe}}(0.05, 0)$	63685	31004	3787	1876	65273	31550	67364	33105	57860	28349	0.02	0.00
	$\text{GCB}_{\text{safe}}(0.10, 0)$	68480	33358	4224	2181	70388	33998	72730	35838	61971	30317	0.65	0.00

Table 8.1: *Results of Experiment #4.*

- U_t: 90-th percentile of the cumulative revenue at round t;
- L_t: 10-th percentile of the cumulative revenue at round t;
- V_{ROI}: the fraction of days in which the ROI constraint is violated;
- V_B: the fraction of days in which the budget constraint is violated;

where W_t, M_t, U_t and L_t are computed over 100 runs.

Table 8.1 reports the algorithms performances at $\lceil T/2 \rceil = 28$ and at the end of the learning process $t = T = 57$. As already observed in the previous experiments, GCB violates the ROI constraint at every round, run, and setting. More surprisingly, GCB violates the budget constraint most of the time (60% on average) even if that constraint is not active at the optimal solution. Interestingly, $\mathsf{GCB}_{\mathtt{safe}}(\psi, 0)$ never violates the budget constraints (for every ψ). As expected, the violation of the ROI constraint is close to zero with $\mathsf{GCB}_{\mathtt{safe}}$, while it increases as ψ increases. In terms of average cumulative revenue, at T, we observe that $\mathsf{GCB}_{\mathtt{safe}}$ gets about 56% of the revenue provided by GCB, while the ratio related to $\mathsf{GCB}_{\mathtt{safe}}(0.05, 0)$ is about 66% and that related to $\mathsf{GCB}_{\mathtt{safe}}(0.10, 0)$ is about 78%. At $T/2$, we the ratios are about 52% for GCB, 61% for $\mathsf{GCB}_{\mathtt{safe}}(0.05, 0)$, and 73% for $\mathsf{GCB}_{\mathtt{safe}}(0.10, 0)$, showing that those ratios increase as T increases. The rationale is that in the early stages of the learning process, safe algorithms learn more slowly than non-safe algorithms. Similar performances can be observed when focusing on the other indices. Summarily, the above results show that our algorithms provide advertisers with a wide spectrum of effective tools to address the revenue/safety tradeoff. A small value of ψ (and ϕ) represents a good tradeoff. By the way, the choice of the specific configuration to adopt in practice depends on the advertiser's aversion to the violation of the constraints.

CHAPTER 9

A Unifying Framework for Online Optimization with Long-Term Constraints

In Section 1.3 we introduced the topic of this chapter. Motivated by the impressive amount of related works in the online convex optimization literature, we provide a summary of the existing results in setting similar to ours. In particular, we underline some of the common assumptions made in the related literature, which we do not consider, hence studying a more general framework. A similar discussion is conducted with respect to the baselines used to measure the performance of the decision maker. In Section 9.2, we present the first best-of-both-world type algorithm for this general class of problems, with no-regret guarantees both in the case in which rewards and constraints are selected according to an unknown, stochastic model, and in the case in which they are selected at each round by an adversary. In Sections 9.3, 9.4, 9.5, 9.6, we analyze the performance of our algorithm in case of stochastic or adversarial constraints and stochastic or adversarial rewards. Finally, in Section 9.7 we show how our framework can be applied in the context of budget-management mechanisms for repeated auctions in order to guarantee long-term constraints that are not packing (e.g., ROI constraints).

Seminal Related Works. The *online convex optimization* (OCO) framework was first proposed in the machine learning literature by Zinkevich (2003), and since then it has significantly expanded becoming widely influential in the learning community (see, *e.g.*, Hazan (2006, 2019); Shalev-Shwartz et al. (2012)). In what follows, we highlight the most relevant works with respect to ours from the literature related to online convex optimization problems with constraints. The analysis and the results are quite different depending on the nature of the constraints, which may be static, *i.e.*, time-invariant, or stochastic/adversarial, *i.e.*, time-variant.

Related Works - Static Constraints. By developing a projection-based *online gradient descent* (OGD) algorithm, Zinkevich (2003) first addressed online convex optimization problems with static constraints. This method guarantees a regret upper bound of $\tilde{O}(T^{1/2})$ for an arbitrary sequence of convex objective functions with bounded subgradients. Hazan et al. (2007) showed that this is a tight bound up to constant factors. When the set defined by the static constraints is complex, the conventional projection-based online algorithms can be difficult to implement due to the potentially high computational cost of carrying out the projection operation. To overcome this difficulty, Mahdavi et al. (2012) propose an efficient algorithm which is an adaptation of OGD achieving a cumulative regret of order $\tilde{O}(T^{1/2})$ and a cumulative constraints violation of $\tilde{O}(T^{3/4})$. These bounds are generalized by Jenatton et al. (2016) who propose an algorithm that achieves a cumulative regret of $\tilde{O}(T^{\max\{\beta,1-\beta\}})$ and a cumulative violation of $\tilde{O}(T^{1-\beta/2})$, where $\beta \in (0,1)$ is a user-defined parameter. Other works, such as, *e.g*, Yuan and Lamperski (2018); Yu and Neely (2020), propose primal-dual algorithms and achieve better bounds by making further assumptions. In particular, Yu and Neely (2020) achieve bounds on the cumulative regret of $\tilde{O}(T^{1/2})$ and on the cumulative violation of $O(1)$ by assuming that the Slater's condition holds (*i.e.*, the existence of a strictly feasible solution). Then, Yuan and Lamperski (2018) achieve a cumulative regret of $O(\log T)$ and a constraint violation of $\tilde{O}(T^{1/2})$ under the assumption that the objective functions are strongly convex. In all the the papers cited above, the regret is computed with respect to the best fixed action in hindsight,that does *not* violate the constraints at each round t. This metric is called *static regret*.

Related Works - Stochastic Constraints. Yu et al. (2017) consider an online convex optimization framework with stochastic constraints, where the objective functions are chosen by an adversary, and the constraint functions are independent and identically distributed (i.i.d.) over time. Yu et al. (2017) provide

a primal-dual proximal gradient algorithm achieving $\tilde{O}(T^{1/2})$ cumulative regret and constraint violation by assuming Slater's condition. Moreover, Wei et al. (2020) provide bounds of the same order by assuming a less stringent version of the Slater's condition. As a performance metric, the latter work use *static regret*, while Yu et al. (2017) employ the same baseline as ours (see Table 9.1).

Related Works - Adversarial Constraints. Various works in the literature address the online learning setting with adversarial reward and constraint functions. This problem was first studied by Mannor et al. (2009) in a two-player game setting. The regret is computed with respect to the best strategy from the set of fixed strategies that satisfy the constraints on average. Mannor et al. (2009) show that in general it is impossible to compete against the best decision in such a set. In particular, they construct a two-player game where there exists a policy for the adversary such that, among the policies of the player that violate sublinearly the constraints, there is no policy that can achieve the no-regret property in terms of maximizing the player's reward. Sun et al. (2017) study a similar problem related to contextual bandits and show that also in their setting the decision maker is unable to compete again this baseline by adapting the result from Mannor et al. (2009) to their setting. To circumvent this issue and provide some guarantees, they rely on a weaker baseline to compute the regret. In particular, they assume that the decision set is rich enough that, in hindsight, there exist a fixed action that satisfies the constraints at each round: they are using the *static regret* as a performance metric. Then, by employing static regret as a baseline, Sun et al. (2017) show that the approaches of Mahdavi et al. (2012) and Jenatton et al. (2016) can be extended to the online learning framework with adversarial sequential constraints. Therefore, they provide an algorithm which is a generalization of that from Mahdavi et al. (2012) achieving sublinear cumulative regret and constraint violations.

Liakopoulos et al. (2019) define a new notion of regret, to overcome the impossibility result from Mannor et al. (2009). They introduce a refined regret metric which compares the agent's incurred losses to those of a K-*benchmark*, which is the best strategy in the hindsight such that, for each time window of length K, the long-term constraints over that window are satisfied. They provide parametric results that are useful to balance the trade-off between regret minimization and long-term residual constraint violation. Moreover, instead of the Slater's condition they consider a less stringent assumption related to the definition of their regret metric.

A recent line of works such as Chen et al. (2017); Chen and Giannakis

(2018) and Cao and Liu (2018) provide some results related to the regret against *dynamic policies*. As expected, comparing against a dynamic baseline require very strong assumptions. Chen et al. (2017) compute a bound on the cumulative dynamic regret which is sublinear in the time horizon T only if the drift of the baseline sequence $\left(i.e., \sum_{t=1}^{T} ||\boldsymbol{x}_{t+1}^* - \boldsymbol{x}_t^*||\right)$ and that of the constraints $\left(i.e., \sum_{t=1}^{T} \max_{\boldsymbol{x}} ||[\boldsymbol{g}_{t+1}(\boldsymbol{x}) - \boldsymbol{g}_t(\boldsymbol{x})]^+||\right)$ are $o(T^{2/3})$. Cao and Liu (2018) consider a bandit feedback setting and, in order to provide sublinear regret and constraint violations, they assume that all the loss functions have uniformly bounded difference (*i.e.*, for each t and $\boldsymbol{x}, \boldsymbol{x}' \in \mathcal{X}$, $|f_t(\boldsymbol{x}) - f_t(\boldsymbol{x}')| \leq M$ for some positive constant M), and that the drift of the baseline sequence is sublinear. In other words, the underlying dynamic optimization problems vary *slowly* over time. Both Chen et al. (2017) and Cao and Liu (2018) need to assume the Slater's condition. Yi et al. (2020) provide similar results in a distributed online convex optimization setting with adversarial constraints. They analyze both the case in which the Slater's condition holds, and the case without this assumption.

Others relevant related works, are those studying online learning problems in which the decision maker has to satisfy supply/budget constraints. In this setting, the decision maker wants to maximize their expected reward without violating a set of m resource constraints. The process stops at time horizon T, or when the total consumption of some resource exceeds its budget. Badanidiyuru et al. (2018) first introduce and solve the *Bandits with Knapsacks* (BwK) framework, in which thay consider bandit feedback, stochastic objective and constraint functions. Other optimal algorithms for Stochastic BwK were proposed by Agrawal and Devanur (2014, 2019) and by Immorlica et al. (2019). The *Adversarial Bandits with Knapsacks* setting was first studied by Immorlica et al. (2019). The authors shows that an appropriate baseline is the best fixed distribution over arms. Achieving no-regret is no longer possible under this baseline and, therefore, they provide no-α-regret guarantees for their algorithm.

We remark that we are able to handle more general constraints than Immorlica et al. (2019), which can deal only with budget constraints. Moreover, we can compete with a baseline stronger than the static regret used by Sun et al. (2017), without needing the strong assumptions required, for instance, by Cao and Liu (2018).

Algorithm	Constr.	Non-convex f_t and g_t	Bound — constant ρ Reward	Bound — constant ρ Violation	Bound — arbitrary ρ Reward	Bound — arbitrary ρ Violation
Yu et al. (2017)	STOC	✗	$\texttt{OPT} - \tilde{O}(T^{1/2})$	$\tilde{O}(T^{1/2})$	—	—
Ours	STOC	✓	$\texttt{OPT} - \tilde{O}(T^{1/2})$	$\tilde{O}(T^{1/2})$	$\texttt{OPT} - \tilde{O}(T^{3/4})$	$\tilde{O}(T^{3/4})$
	ADV	✓	$\frac{\rho}{1+\rho}\texttt{OPT} - \tilde{O}\left(T^{1/2}\right)$	$\tilde{O}(T^{1/2})$	—	—

Table 9.1: *Comparison between the performance of our algorithm and previous work using the same baseline as ours. Bounds for settings that were* not *previously tractable are highlighted in gray.* $\texttt{OPT}$ *is the reward of the baseline.*

9.1 Preliminaries

The decision maker has a non-empty set of available strategies $\mathcal{X}$ (this set may be non-convex, integral, and even non-compact). In each round $t \in [T]$,[1] the decision maker first chooses $\boldsymbol{x}_t \in \mathcal{X}$, and the environment selects a reward function $f_t : \mathcal{X} \to [0, 1]$ and a constraint function $g_t : \mathcal{X} \to [-1, 1]^m$ conditioned on the past history of play up to time $t - 1$ (*i.e.*, the environment chooses f_t and g_t without knowledge of $\boldsymbol{x}_t$). Notice that both f_t and g_t need *not* be convex. The latter specifies a set of m constraints of the form $g_t(\boldsymbol{x}) \leq \mathbf{0}$, with $g_{t,i}(\boldsymbol{x}) \leq 0$ denoting the i-th constraint.[2] In the following, we denote as $\mathcal{F}$, respectively $\mathcal{G}$, the set of all the possible f_t, respectively g_t, functions (*e.g.*, $\mathcal{F}$ and $\mathcal{G}$ may contain all the Lipschitz-continuous functions defined over $\mathcal{X}$). At each round $t \in [T]$, the decision maker can condition their decision on prior feedbacks and on the sequence of prior decisions $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-1}$, but no information about future rewards and constraint functions is available.

9.1.1 Strong duality through strategy mixtures

Next, we define the optimization problem (Problem $\texttt{LP}_{f,g}$) which is used to define the baselines against which we compare the performances of the decision maker. Such a problem involves probabilistic mixtures of strategies in $\mathcal{X}$, which are crucial in order to recover strong duality.[3]

First, we introduce the set of probability measures on the Borel sets of $\mathcal{X}$. We refer to such a set as the set of *strategy mixtures*, denoted as

[1] In this work, we denote by $[x]$ the set $\{1, \ldots, x\}$ of the first x natural numbers.

[2] Focusing on the case $g_t(\boldsymbol{x}) \leq \mathbf{0}$ is w.l.o.g. since any set of constraints can be represented in such a form.

[3] The optimal fixed strategy mixture provides an arguably stronger baseline than the optimal fixed strategy. In stochastic settings, this baseline is related to the best dynamic policy. In particular, if we consider the case in which the observed functions are defined as the average of functions f_t and g_t across the T rounds, then the optimal mixture provides the same utility as the best dynamic policy (see Badanidiyuru et al. (2018) for a similar result).

Ξ. We endow $\mathcal{X}$ with the Lebesgue σ-algebra, and we assume that all the functions in $\mathcal{F}$ and $\mathcal{G}$ are measurable with respect to every probability measure $\boldsymbol{\xi} \in \Xi$. This ensures that the various expectations taken are well-defined, since the functions are assumed to be bounded above, and they are therefore integrable. In the following, for the ease of presentation and with a slight abuse of notation, whenever we write a $\boldsymbol{\xi} \in \Xi$ in place of an $\boldsymbol{x} \in \mathcal{X}$, we mean that we are taking the expectation with respect to the probability measure $\boldsymbol{\xi}$. For instance, given $f \in \mathcal{F}$ and $g \in \mathcal{G}$, we have that $f(\boldsymbol{\xi}) = \mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{\xi}} f(\boldsymbol{x})$ and $g(\boldsymbol{\xi}) = \mathbb{E}_{\boldsymbol{x}\sim\boldsymbol{\xi}} g(\boldsymbol{x})$.

Then, given two functions $f \in \mathcal{F}$ and $g \in \mathcal{G}$, we define the following optimization problem, which chooses the strategy mixture $\boldsymbol{\xi} \in \Xi$ that maximizes the expected reward encoded by f, while guaranteeing that the constraints encoded by g are satisfied in expectation.

$$\texttt{OPT}_{f,g} \coloneqq \begin{cases} \sup\limits_{\boldsymbol{\xi}\in\Xi} & f(\boldsymbol{\xi}) \qquad \text{s.t.} \\ & g(\boldsymbol{\xi}) \leq 0. \end{cases} \tag{$\texttt{LP}_{f,g}$}$$

We denote by $d_g \in [-1, 1]$ the largest possible value for which there exists a strategy mixture $\boldsymbol{\xi} \in \Xi$ satisfying the constraints $g(\boldsymbol{\xi}) \leq 0$ by a margin of at least d_g. Formally,

$$d_g \coloneqq \sup_{\boldsymbol{\xi}\in\Xi} \min_{i\in[m]} -g_i(\boldsymbol{\xi}). \tag{9.1}$$

In order to ensure that $\texttt{OPT}_{f,g}$ is always well defined, we assume that it is always the case that $d_g \geq 0$. Notice that, if $d_g > 0$, then Problem $\texttt{LP}_{f,g}$ satisfies Slater's condition.

In the following, we prove some auxiliary results relating to Problem $\texttt{LP}_{f,g}$ that will be useful in the rest of the chapter. First, we introduce a Lagrangian relaxation of the problem.

Definition 9.1 (Lagrangian Function)**.** *Given two arbitrary functions $f \in \mathcal{F}$ and $g \in \mathcal{G}$, the* Lagrangian function $\mathcal{L}_{f,g} : \Xi \times \mathbb{R}^m_{\geq 0} \to \mathbb{R}$ *of Problem $\texttt{LP}_{f,g}$ is defined as*

$$\mathcal{L}_{f,g}(\boldsymbol{\xi}, \boldsymbol{\lambda}) \coloneqq f(\boldsymbol{\xi}) - \langle \boldsymbol{\lambda}, g(\boldsymbol{\xi}) \rangle.$$

If Problem $\texttt{LP}_{f,g}$ satisfies Slater's condition, then Theorem 1 of Chapter 8.3 in (Luenberger, 1997) readily gives us that strong duality holds even if f and g are arbitrary non-convex functions. Formally:

Corollary 9.1. *Given $f \in \mathcal{F}$ and $g \in \mathcal{G}$ such that $d_g > 0$, it holds*

$$\sup_{\boldsymbol{\xi}\in\Xi} \inf_{\boldsymbol{\lambda}\in\mathbb{R}^m_{\geq 0}} \mathcal{L}_{f,g}(\boldsymbol{\xi}, \boldsymbol{\lambda}) = \inf_{\boldsymbol{\lambda}\in\mathbb{R}^m_{\geq 0}} \sup_{\boldsymbol{\xi}\in\Xi} \mathcal{L}_{f,g}(\boldsymbol{\xi}, \boldsymbol{\lambda}) = \texttt{OPT}_{f,g}.$$

Next, we show that, if $d_g > 0$, then strong duality holds even when we restrict the admissible dual vectors $\boldsymbol{\lambda} \in \mathbb{R}^m_{\geq 0}$ to the set $\mathcal{D}_{d_g}$, where, for any $q \in \mathbb{R}_{>0}$, we let $\mathcal{D}_q \coloneqq \left\{\boldsymbol{\lambda} \in \mathbb{R}^m_{\geq 0} : \|\boldsymbol{\lambda}\|_1 \leq 1/q\right\}$.

Theorem 9.1. *Given $f \in \mathcal{F}$ and $g \in \mathcal{G}$ such that $d_g > 0$, it holds*

$$\sup_{\boldsymbol{\xi} \in \Xi} \inf_{\boldsymbol{\lambda} \in \mathcal{D}_{d_g}} \mathcal{L}_{f,g}(\boldsymbol{\xi}, \boldsymbol{\lambda}) = \inf_{\boldsymbol{\lambda} \in \mathcal{D}_{d_g}} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}_{f,g}(\boldsymbol{\xi}, \boldsymbol{\lambda}) = \texttt{OPT}_{f,g}.$$

Proof. As a first step, we prove that

$$\inf_{\boldsymbol{\lambda} \in \mathcal{D}_{d_g}} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) = \inf_{\boldsymbol{\lambda} \in \mathbb{R}^m_+} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}).$$

Notice that for each $\boldsymbol{\lambda} \in \mathbb{R}^m_+$ such that $\|\boldsymbol{\lambda}\|_1 > 1/d_g$, it holds

$$\sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) \geq \mathcal{L}(\boldsymbol{\xi}^\circ, \boldsymbol{\lambda}) \geq -\langle \boldsymbol{\lambda}^*, g(\boldsymbol{\xi}^\circ) \rangle \geq d_g \|\boldsymbol{\lambda}^*\|_1 > 1,$$

where, with an abuse of notation, $\boldsymbol{\xi}^\circ \in \Xi$ denotes a strictly feasible strategy mixture for Problem $\texttt{LP}_{f,g}$. That is a strategy mixture $\boldsymbol{\xi} \in \Xi$ which is optimal for the problem defining d_g in Equation (9.1), and, thus, it satisfies all the constraints by at least d_g (*i.e.*, it holds $g_i(\boldsymbol{\xi}^\circ) \leq -d_g$ for all $i \in [m]$).[4] Thus, it holds that

$$\inf_{\boldsymbol{\lambda} \in \mathbb{R}^m_+ \setminus \mathcal{D}_{d_g}} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) > 1.$$

Moreover, since

$$\inf_{\boldsymbol{\lambda} \in \mathcal{D}_{d_g}} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) \leq \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \mathbf{0}) \leq 1,$$

we can conclude that

$$\begin{aligned} \inf_{\boldsymbol{\lambda} \in \mathbb{R}^m_+} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) &= \min \left\{ \inf_{\boldsymbol{\lambda} \in \mathcal{D}_{d_g}} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}); \inf_{\boldsymbol{\lambda} \in \mathbb{R}^m_+ \setminus \mathcal{D}_{d_g}} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) \right\} \\ &= \inf_{\boldsymbol{\lambda} \in \mathcal{D}_{d_g}} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}). \end{aligned} \tag{9.2}$$

Then,

$$\begin{aligned} \texttt{OPT}_{f,g} &= \sup_{\boldsymbol{\xi} \in \Xi} \inf_{\boldsymbol{\lambda} \in \mathbb{R}^m_+} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) \\ &\leq \sup_{\boldsymbol{\xi} \in \Xi} \inf_{\boldsymbol{\lambda} \in \mathcal{D}_{d_g}} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) \end{aligned}$$

[4] Notice that $\boldsymbol{\xi}^\circ$ may *not* be well defined when the problem in Equation 9.1 does *not* admit a maximum. In such cases, we can take a $\boldsymbol{\xi}^\circ$ that is arbitrarily "close" to a supremum, so that the result still holds.

$$\begin{aligned}
&\leq \inf_{\boldsymbol{\lambda} \in \mathcal{D}_{d_g}} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) \\
&= \inf_{\boldsymbol{\lambda} \in \mathbb{R}^m_+} \sup_{\boldsymbol{\xi} \in \Xi} \mathcal{L}(\boldsymbol{\xi}, \boldsymbol{\lambda}) \\
&= \texttt{OPT}_{f,g},
\end{aligned}$$

where the first inequality holds since in the right-hand side the $\inf$ is taken over the more restrictive set $\mathcal{D}_{d_g}$, the second one by the *max–min inequality*, while the second-to-last equality holds by Equation (9.2). This concludes the proof. □

9.1.2 Stochastic vs. adversarial: baselines and feasibility

We consider several settings, differing in how functions f_t and g_t are selected, either *stochastically* or *adversarially*. We say that functions f_t (respectively g_t) are selected stochastically, when they are independently drawn according to a given probability measure $\mu_{\mathcal{F}}$ over $\mathcal{F}$ (respectively $\mu_{\mathcal{G}}$ over $\mathcal{G}$). Instead, we say that functions f_t (respectively g_t) are selected adversarially if each f_t (respectively g_t) is chosen by an adversary based on the sequence of prior decisions, namely $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{t-1}$.

Consistently with previous work (see, *e.g.*, (Mannor et al., 2009)), we compare the performance of the decision maker (in terms of reward cumulated over the T rounds) against the baseline $T\,\texttt{OPT}_{\bar{f},\bar{g}}$ (as defined by Problem $\texttt{LP}_{\bar{f},\bar{g}}$), where $\bar{f}$ and $\bar{g}$ are suitably-defined functions. In particular:

- When functions f_t, respectively g_t, are selected stochastically, then we define function $\bar{f}$, respectively $\bar{g}$, so that $\bar{f}(\boldsymbol{x}) \coloneqq \mathbb{E}_{f \sim \mu_{\mathcal{F}}}[f(\boldsymbol{x})]$, respectively $\bar{g}(\boldsymbol{x}) \coloneqq \mathbb{E}_{g \sim \mu_{\mathcal{G}}}[g(\boldsymbol{x})]$.
- When functions f_t, respectively g_t, are selected adversarially, then we define function $\bar{f}$, respectively $\bar{g}$, so that $\bar{f}(\boldsymbol{x}) \coloneqq \frac{1}{T}\sum_{t=1}^{T} f_t(\boldsymbol{x})$, respectively $\bar{g}(\boldsymbol{x}) \coloneqq \frac{1}{T}\sum_{t=1}^{T} g_t(\boldsymbol{x})$.

Intuitively, in the stochastic case, the baseline is instantiated with an expectation of functions taken with respect to the probability measure $\mu_{\mathcal{F}}$ (respectively $\mu_{\mathcal{G}}$). Instead, in the adversarial case, the baseline uses the average of functions f_t (respectively g_t) observed over the T rounds.

Let us remark that, when the set $\mathcal{X}$ is compact convex and functions f_t and g_t are convex, then Problem $\texttt{LP}_{\bar{f},\bar{g}}$ defining our baselines can be equivalently re-written by using strategies $\boldsymbol{x} \in \mathcal{X}$ rather than strategy mixtures $\boldsymbol{\xi} \in \Xi$, since there always exists an optimal solution to Problem $\texttt{LP}_{\bar{f},\bar{g}}$ that places all the probability mass on a single strategy.

Our goal is to design online algorithms for the decision maker that output a sequence of decisions $\boldsymbol{x}_1, \dots, \boldsymbol{x}_T$ such that both the *cumulative regret* with respect to the performance of the baseline, defined as $R^T := T\,\texttt{OPT}_{\bar{f},\bar{g}} - \sum_{t=1}^{T} f_t(\boldsymbol{x}_t)$, and the *cumulative constraints violation*, defined as $V^T := \max_{i\in[m]} \sum_{t=1}^{T} g_{t,i}(\boldsymbol{x}_t)$, grow sublinearly in the number of rounds T.

In conclusion, we introduce a problem-specific parameter that is strictly related to the feasibility of Problem $\texttt{LP}_{\bar{f},\bar{g}}$. We call it the *feasibility parameter* $\rho \in \mathbb{R}$, which is formally defined as follows:

- When functions g_t are selected stochastically:
  $\rho := \sup_{\boldsymbol{\xi}\in\Xi} \min_{i\in[m]} -\bar{g}_i(\boldsymbol{\xi})$.
- When functions g_t are selected adversarially:
  $\rho := \sup_{\boldsymbol{\xi}\in\Xi} \min_{t\in[T]} \min_{i\in[m]} -g_{t,i}(\boldsymbol{\xi})$.

Intuitively, in the stochastic case, ρ is equal to $d_{\bar{g}}$, while in the adversarial case it is computed similarly, but considering the worst case with respect to the functions g_t observed at each round t. Notice that, when $\rho > 0$, Slater's condition is satisfied for Problem $\texttt{LP}_{\bar{f},\bar{g}}$.

In the following, we denote by $\boldsymbol{\xi}^* \in \Xi$ a strategy mixture that is optimal for Problem $\texttt{LP}_{\bar{f},\bar{g}}$. Moreover, we always assume that functions f_t and g_t are such that Problem $\texttt{LP}_{\bar{f},\bar{g}}$ is feasible, and we let $\boldsymbol{\xi}^\circ \in \Xi$ be the *feasible* strategy mixture that is optimal for the problem defining ρ.[5]

9.1.3 Regret minimizers

A *regret minimizer* (RM) for a set $\mathcal{W}$ is an abstract model for a decision maker that repeatedly interacts with a black-box environment. At each t, a RM performs two operations: (i) NEXTELEMENT(), which outputs an element $\boldsymbol{w}_t \in \mathcal{W}$; and (ii) OBSERVEUTILITY(u_t), which updates the internal state of the RM using the feedback received from the environment. This is defined in terms of a utility function $u_t : \mathcal{W} \to [a, b]$ having range $[a, b] \subseteq \mathbb{R}$, with u_t possibly depending adversarially on the sequence of outputs $\boldsymbol{w}_1, \dots, \boldsymbol{w}_{t-1}$. The objective of the RM is to output a sequence $\boldsymbol{w}_1, \dots, \boldsymbol{w}_T$ of points in $\mathcal{W}$ so that its *cumulative regret*, defined as $\sup_{\boldsymbol{w}\in\mathcal{W}} \sum_{t=1}^{T} (u_t(\boldsymbol{w}) - u_t(\boldsymbol{w}_t))$, grows asymptotically sublinearly in T. See (Cesa-Bianchi and Lugosi, 2006) for a review of the various RMs available in the literature.

[5] Notice that $\boldsymbol{\xi}^*$ and $\boldsymbol{\xi}^\circ$ may *not* be well defined in all the cases in which the problem that defines them does *not* admit a maximum. Nevertheless, in such cases, we assume that $\boldsymbol{\xi}^*$ (or $\boldsymbol{\xi}^\circ$) is a strategy mixture arbitrarily "close" to the supremum, so that all of our results continue to hold up to negligible additive approximations that are dominated by other approximation factors, and we can safely ignore them for the ease of exposition.

For the ease of presentation, we introduce the concept of *regret minimizer constructor*, which is a procedure, say $\textsc{Init}(\mathcal{W}, [a, b], \eta)$, that builds a RM on the basis of the three parameters given as input. In particular, the procedure returns a RM instantiated for the set $\mathcal{W}$, working with utility functions having range $[a, b]$, and such that its cumulative regret is guaranteed to grow sublinearly in the time horizon T with probability at least $1 - \eta$.

9.2 A unifying meta-algorithm

In this section, we present our meta-algorithm. Its core idea is to instantiate suitable pairs of RMs, where one is working in the domain $\mathcal{X}$ of primal variables and the other in a suitable subset of the domain $\mathbb{R}^m_+$ of dual variables. At each round $t \in [T]$, the algorithm makes the RMs "play" against each other in a *Lagrangian game*, where the utility functions observed by them are related to the Lagrangian function $\mathcal{L}_{f_t,g_t}(\boldsymbol{x}, \boldsymbol{\lambda})$ of Problem $\texttt{LP}_{f_t,g_t}$.[6]

Algorithm 9.1 provides the pseudo-code of the meta-algorithm, which takes as input: the total number of rounds T, a failure probability $\delta \in (0, 1)$ such that the guarantees provided by the algorithm hold with probability at least $1-\delta$, and a lower bound $\hat{\rho} \geq 0$ on the value of the feasibility parameter ρ.

Algorithm description. The algorithm works in two phases. In the first one, called *play phase*, the algorithm builds a primal RM, called $\mathcal{R}^{\text{P}}_{\text{I}}$, working in the primal domain $\mathcal{X}$ and a dual RM, called $\mathcal{R}^{\text{D}}_{\text{I}}$, operating on the subset $\mathcal{D}_{\tilde{\rho}}$ of the dual domain $\mathbb{R}^m_+$, where $\tilde{\rho}$ is set in Line 1. The algorithm makes the two RMs playing against each other (see the call $\textsc{LagrangianGame}(\mathcal{R}^{\text{P}}_{\text{I}}, \mathcal{R}^{\text{D}}_{\text{I}}, 1)$) until either the cumulative violation V^t incurred by the algorithm exceeds a given threshold (see Line 4, where $M_{\tilde{\rho}}$ is defined in Equation (9.3)) or round T is reached. Then, in the second phase, called *recovery phase*, the algorithm constructs a new pair of primal, dual RMs, with the latter working on the $(m-1)$-dimensional simplex Δ_m. The recovery phase uses the remaining rounds to make these new RMs play against each other, with the primal RM observing modified utility functions that do *not* account for functions f_t (see the call $\textsc{LagrangianGame}(\mathcal{R}^{\text{P}}_{\text{II}}, \mathcal{R}^{\text{D}}_{\text{II}}, 0)$). Intuitively, this is needed in order to ensure that the algorithm plays strategies $\boldsymbol{x}_t$ that satisfy the constraints, thus balancing out the cumulative constraint violation accumulated in the first phase. The pseudo-code describing one "play" between two RMs, called $\mathcal{R}^{\text{P}}$

[6] The idea of having pairs of primal, dual RMs playing a Lagrangian game was originally introduced by Immorlica et al. (2019), restricted to the case of knapsack constraints.

and $\mathcal{R}^{\text{D}}$, is defined by the sub-procedure LAGRANGIANGAME$(\mathcal{R}^{\text{P}}, \mathcal{R}^{\text{D}}, v)$ in Algorithm 9.2. The additional parameter $v \in \{0, 1\}$ is used to control the feedback fed into the primal RM $\mathcal{R}^{\text{P}}$; specifically, if $v = 1$, then $\mathcal{R}^{\text{P}}$ observes a utility function that also accounts for f_t (play phase), otherwise, if $v = 0$, the observed utility function only accounts for the term depending on g_t (recovery phase).

Algorithm 9.1 META-ALGORITHM$(T, \delta, \hat{\rho})$

$\tilde{\rho} \leftarrow \max\left\{\hat{\rho}/2, T^{-1/4}\right\}, \eta \leftarrow \delta/3, t \leftarrow 1$
▷ Phase I: Play
$\mathcal{R}^{\text{P}}_{\text{I}} \leftarrow \text{INIT}^{\text{P}}\left(\mathcal{X}, \left[-1/\tilde{\rho}, 1 + 1/\tilde{\rho}\right], \eta\right)$
$\mathcal{R}^{\text{D}}_{\text{I}} \leftarrow \text{INIT}^{\text{D}}\left(\mathcal{D}_{\tilde{\rho}}, \left[-1/\tilde{\rho}, 1/\tilde{\rho}\right], 0\right)$
while $V^t \leq (T - t)\tilde{\rho} + M_{\tilde{\rho}} - 1 \wedge t \leq T$ **do**
    $\boldsymbol{x}_t \leftarrow \text{LAGRANGIANGAME}(\mathcal{R}^{\text{P}}_{\text{I}}, \mathcal{R}^{\text{D}}_{\text{I}}, 1)$
    $t \leftarrow t + 1$
end while
$T_1 \leftarrow t - 1$
▷ Phase II: Recovery
$\mathcal{R}^{\text{P}}_{\text{II}} \leftarrow \text{INIT}^{\text{P}}\left(\mathcal{X}, [-1, 1], \eta\right)$
$\mathcal{R}^{\text{D}}_{\text{II}} \leftarrow \text{INIT}^{\text{D}}\left(\Delta_m, [-1, 1], 0\right)$
while $t \leq T$ **do**
    $\boldsymbol{x}_t \leftarrow \text{LAGRANGIANGAME}(\mathcal{R}^{\text{P}}_{\text{II}}, \mathcal{R}^{\text{D}}_{\text{II}}, 0)$
    $t \leftarrow t + 1$
end while

Algorithm 9.2 LAGRANGIANGAME$(\mathcal{R}^{\text{P}}, \mathcal{R}^{\text{D}}, v)$

$\boldsymbol{x}_t \leftarrow \mathcal{R}^{\text{P}}.\text{NEXTELEMENT}()$
$\boldsymbol{\lambda}_t \leftarrow \mathcal{R}^{\text{D}}.\text{NEXTELEMENT}()$
Play $\boldsymbol{x}_t$ and get f_t and g_t ▷ Full f.
Play $\boldsymbol{x}_t$ and get $f_t(\boldsymbol{x}_t)$ and $g_t(\boldsymbol{x}_t)$ ▷ Bandit f.
▷ Primal RM update
Let $u^{\text{P}}_t : \boldsymbol{x} \mapsto v f_t(\boldsymbol{x}) - \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}) \rangle$ ▷ Full f.
$u^{\text{P}}_t(\boldsymbol{x}_t) \leftarrow v f_t(\boldsymbol{x}_t) - \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t) \rangle$ ▷ Bandit f.
$\mathcal{R}^{\text{P}}.\text{OBSERVEUTILITY}(u^{\text{P}}_t)$ ▷ Full f.
$\mathcal{R}^{\text{P}}.\text{OBSERVEUTILITY}(u^{\text{P}}_t(\boldsymbol{x}_t))$ ▷ Bandit f.
▷ Dual RM update
Let $u^{\text{D}}_t : \boldsymbol{\lambda} \mapsto -\langle \boldsymbol{\lambda}, g_t(\boldsymbol{x}) \rangle$
$\mathcal{R}^{\text{D}}.\text{OBSERVEUTILITY}(u^{\text{D}}_t)$

Regret minimizer constructors. Algorithm 9.1 also needs access to two suitably-defined regret minimizer constructors, denoted by $\text{INIT}^{\text{P}}(\mathcal{W}, [a, b], \eta)$ and

$\textsc{Init}^{\mathrm{D}}(\mathcal{W}, [a, b], \eta)$, where the former is used to build RMs working in the primal domain and the latter for those operating on the dual one. Their actual implementation depends on the specific problem at hand. In the following, we let $\mathcal{E}^{\mathrm{P}}_{t,\eta}$ be the regret upper bound (on $t \in [T]$ rounds) for primal RMs $\mathcal{R}^{\mathrm{P}}$ dealing with utility functions having range $[0, 1]$, as returned by the call $\textsc{Init}^{\mathrm{P}}(\mathcal{X}, [0, 1], \eta)$. Notice that, when the range is $[a, b]$, the same RM can be adopted by first normalizing utility values, so that the resulting regret upper bound is $(b - a)\mathcal{E}^{\mathrm{P}}_{t,\eta}$. As for dual RMs $\mathcal{R}^{\mathrm{D}}$, we let $\mathcal{E}^{\mathrm{D}}_t$ be the regret upper bound (on round $t \in [T]$) provided by the RM defined for the set Δ_m, while $\mathcal{E}^{\mathrm{D}}_t/\tilde{\rho}$ is the upper bound for the dual RM instantiated on the set $\mathcal{D}_{\tilde{\rho}}$. Notice that, since dual RMs always have full feedback, we can safely assume that the regret bounds $\mathcal{E}^{\mathrm{D}}_t$ hold deterministically. We also assume that RMs provide bounds that increase with the number of rounds, *i.e.*, such that $\mathcal{E}^{\mathrm{P}}_{t,\eta} \leq \mathcal{E}^{\mathrm{P}}_{t',\eta}$ and $\mathcal{E}^{\mathrm{D}}_t \leq \mathcal{E}^{\mathrm{D}}_{t'}$ for all $t \leq t'$.

How to construct RMs. $\textsc{Init}^{\mathrm{D}}$ can be implemented by using *online mirror descent* (OMD) with domain Δ_m (or $\mathcal{D}_1$) and a negative entropy regularizer. Since the utility function u^{D}_t is linear in $\boldsymbol{\lambda}$, we get a regret bound for the primal RM of $\mathcal{E}^{\mathrm{D}}_T = O(\sqrt{T \log(m)})$ (see, *e.g.*, Beck and Teboulle (2003); Nemirovskij and Yudin (1983)). The design of $\textsc{Init}^{\mathrm{P}}$ depends on the structure of $\mathcal{X}$ and functions f_t and g_t. For instance, in convex settings with full feedback we can employ OMD (Hazan, 2019), while with bandit feedback we can use (Bubeck et al., 2017). Finally, for non-convex functions we can employ, *e.g.*, the RMs in (Suggala and Netrapalli, 2020). All these RMs guarantee $\tilde{O}(\sqrt{T})$ regret.

How to get away with no knowledge of ρ. In Section 9.6, we show that a lower bound $\hat{\rho}$ is *not* necessary when functions g_t are selected stochastically. Indeed, it is sufficient to add a preliminary phase to Algorithm 9.1, which is used to infer a suitable lower bound on ρ from experience. In order to do this, only $\sqrt{T}$ rounds are needed, so that the bounds of Algorithm 9.1 are *not* compromised. When functions g_t are chosen adversarially, it is easy to see that it is impossible to compute a lower bound on the feasibility parameter ρ by only using the first rounds. For instance, think of a setting in which ρ is very large when only considering the first rounds, while it becomes small during later rounds.

Remark 9.1 (Dependence on the lower bound $\hat{\rho}$)**.** *Algorithm 9.1 can take as input any $\hat{\rho} \geq 0$. However, since our regret bounds include a factor $1/\tilde{\rho}$, by choosing the trivial lower bound $\hat{\rho} = 0$ we incur in a regret of*

$\tilde{O}(\sqrt{T}/\tilde{\rho}) = \tilde{O}(T^{3/4})$. *In order to obtain optimal bounds, we would like to have* $\tilde{\rho} = \Omega(\rho)$.

Remark 9.2 (Dependence on the feasibility parameter ρ)**.** *We choose to include the dependence on the feasibility parameter* ρ *in the order of convergence of the algorithm. As customary, the goal is devising bounds in the form* poly(instance) $\cdot\, h(T)$*, where the first term is a polynomial function of the parameters defining the problem instance, and* $h(T) = o(T)$*. Therefore, we cannot include a factor* $1/\rho$ *in the regret bounds if* ρ *can be arbitrarily small. Even from a practical standpoint, when* ρ *is too small a* $1/\rho$ *regret bound is too large to be significant. For those reasons, we set* $\tilde{\rho}$ *in Algorithm 9.1 to be the maximum between the feasibility parameter lower bound* $\hat{\rho}$ *and* $T^{-1/4}$*. The value* $T^{-1/4}$ *has been chosen so as to minimize the maximum between the cumulative regret and the cumulative constraint violation when the lower bound on the feasibility parameter* $\hat{\rho}$ *is too small.*

9.3 Analysis with stochastic constraints and adversarial rewards

We start by analyzing the performance of our meta-algorithm (Algorithm 9.1) when the reward and constraint functions are selected stochastically and adversarially, respectively.

Given $t \in [T]$ and $\eta \in (0, 1)$, we let $\mathcal{E}_{t,\eta} \coloneqq \sqrt{8t \log(18mt^2/\eta)}$ be the value bounding differences between expectations and empirical means of constraint functions, obtained by applying the Azuma-Hoeffding inequality, and holding with probability at least $1 - \eta$. Given $\gamma \in (0, 1)$, we also let

$$M_\gamma \coloneqq \frac{2}{\gamma}\sqrt{T} + \left(2 + \frac{3}{\gamma}\right) \mathcal{E}_{t,\eta} + \left(1 + \frac{2}{\gamma}\right) \mathcal{E}^{\text{P}}_{t,\eta} + \frac{1}{\gamma}\mathcal{E}^{\text{D}}_{t}, \tag{9.3}$$

which is a recurring term related to the maximum violation that Algorithm 9.1 accepts in play phase.

First, we introduce a useful event $\mathbf{E}$ that encompasses all the cases in which Algorithm 9.1 successfully terminates. Then, Lemma 9.1 shows that such an event holds with probability at least $1 - \delta$. In particular, $\mathbf{E}$ holds when the regret bounds of $\mathcal{R}^{\text{P}}_{\text{I}}$ and $\mathcal{R}^{\text{P}}_{\text{II}}$ hold, and, additionally, the differences between expectations and empirical means of constraint functions are bounded as desired.

Definition 9.2. *We denote with* $\mathbf{E}$ *the event in which Algorithm 9.1 satisfies the following conditions (recall that* $\eta = \delta/3$*): (i) the regret incurred by* $\mathcal{R}^{P}_{\text{I}}$ *after* T_1 *rounds is upper bounded by* $\mathcal{E}^{P}_{T_1,\eta}$*; (ii) the regret cumulated by* $\mathcal{R}^{P}_{\text{II}}$

after the remaining $T - T_1$ rounds is upper bounded by $\mathcal{E}^{P}_{T-T_1,\eta}$; and (iii) for every pair of rounds $t, t' \in [T] : t \leq t'$ and resource $i \in [m]$ it holds:

- $\left|\sum_{\tau=t}^{t'} g_{\tau,i}(\boldsymbol{x}_\tau) - \sum_{\tau=t}^{t'} \bar{g}_i(\boldsymbol{x}_\tau)\right| \leq \mathcal{E}_{t'-t,\eta}$,
- $\left|\sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau g_{\tau,i}(\boldsymbol{x}_\tau) - \sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau \bar{g}_i(\boldsymbol{x}_\tau)\right| \leq \mathcal{E}_{t'-t,\eta} \max_{\tau\in[T]:t\leq\tau\leq t'} ||\boldsymbol{\lambda}_\tau||_1$,
- $\left|\sum_{\tau=t}^{t'} g_{\tau,i}(\boldsymbol{\xi}) - \sum_{\tau=t}^{t'} \bar{g}_i(\boldsymbol{\xi})\right| \leq \mathcal{E}_{t'-t,\eta}$, *for* $\boldsymbol{\xi} \in \{\boldsymbol{\xi}^*, \boldsymbol{\xi}^\circ\}$,
- $\left|\sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau g_{\tau,i}(\boldsymbol{\xi}) - \sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau \bar{g}_i(\boldsymbol{\xi})\right| \leq \mathcal{E}_{t'-t,\eta} \max_{\tau\in[T]:t\leq\tau\leq t'} ||\boldsymbol{\lambda}_\tau||_1$, *for* $\boldsymbol{\xi} \in \{\boldsymbol{\xi}^*, \boldsymbol{\xi}^\circ\}$.

Lemma 9.1. *After running Algorithm 9.1, the event* $\mathbf{E}$ *holds with probability at least* $1 - \delta$.

Proof. Given a desired failure probability $\delta \in (0, 1)$, recall that $\eta = \delta/3$ and set $\varepsilon = \eta/18mT^2$. Consider the following inequalities in which the differences between expectations and empirical means of constraint functions are bounded:

$$\left|\sum_{\tau=t}^{t'} g_{\tau,i}(\boldsymbol{x}_\tau) - \sum_{\tau=t}^{t'} \bar{g}_i(\boldsymbol{x}_\tau)\right| > 2\sqrt{2(t'-t)\ln\frac{1}{\varepsilon}}, \tag{9.4}$$

$$\left|\sum_{\tau=t}^{t'} g_{\tau,i}(\boldsymbol{\xi}^\circ) - \sum_{\tau=t}^{t'} \bar{g}_i(\boldsymbol{\xi}^\circ)\right| > 2\sqrt{2(t'-t)\ln\frac{1}{\varepsilon}}, \tag{9.5}$$

$$\left|\sum_{\tau=t}^{t'} g_{\tau,i}(\boldsymbol{\xi}^*) - \sum_{\tau=t}^{t'} \bar{g}_i(\boldsymbol{\xi}^*)\right| > 2\sqrt{2(t'-t)\ln\frac{1}{\varepsilon}}, \tag{9.6}$$

$$\left|\sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau g_{\tau,i}(\boldsymbol{x}_\tau) - \sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau \bar{g}_i(\boldsymbol{x}_\tau)\right| > 2 \max_{\tau\in[T]:t\leq\tau\leq t'} ||\boldsymbol{\lambda}_\tau||_1 \sqrt{2(t'-t)\ln\frac{1}{\varepsilon}}, \tag{9.7}$$

$$\left|\sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau\, g_{\tau,i}(\boldsymbol{\xi}^*) - \sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau\, \bar{g}_i(\boldsymbol{\xi}^*)\right| > 2 \max_{\tau\in[T]:t\leq\tau\leq t'} ||\boldsymbol{\lambda}_\tau||_1 \sqrt{2(t'-t)\ln\frac{1}{\varepsilon}},$$

$$\left|\sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau\, g_{\tau,i}(\boldsymbol{\xi}^\circ) - \sum_{\tau=t}^{t'} \boldsymbol{\lambda}_\tau\, \bar{g}_i(\boldsymbol{\xi}^\circ)\right| \leq 2 \max_{\tau\in[T]:t\leq\tau\leq t'} ||\boldsymbol{\lambda}_\tau||_1 \sqrt{2(t'-t)\ln\frac{1}{\varepsilon}}.$$

By applying Azuma-Hoeffding inequality to each martingale difference sequence, we get that each inequality holds with probability at most 2ε. We denote by $\mathbf{E}_\eta$ the event in which Equations (9.4), (9.5), (9.6), and (9.7) are satisfied for all $t, t' \in [T]$ with $t < t'$, and for all $i \in [m]$. By a union bound

that takes into account the six events above, the m constraints, and all the possible time intervals from t to t', which are at most T^2, we get:

$$\mathbb{P}(\mathbf{E}_\eta) \geq 1 - 6mT^2(2\varepsilon) = 1 - 12mT^2\varepsilon = 1 - \frac{2}{3}\eta \geq 1 - \eta.$$

Therefore, event $\mathbf{E}_\eta$ holds with probability at least $1 - \eta$. Moreover, let us recall that:

$$\mathcal{E}_{t'-t,\eta} = \sqrt{8(t'-t)\ln\frac{18mT^2}{\eta}} = 2\sqrt{2(t'-t)\ln\frac{1}{\varepsilon}}.$$

Now, consider event $\mathbf{E}$ in which Algorithms 9.1 satisfies the following conditions: (i) the regret incurred by $\mathcal{R}_{\text{I}}^{\text{P}}$ after T_1 rounds is upper bounded by $\mathcal{E}_{T_1,\eta}^{\text{P}}$; (ii) the regret cumulated by $\mathcal{R}_{\text{II}}^{\text{P}}$ after the remaining $T - T_1$ rounds is upper bounded by $\mathcal{E}_{T-T_1,\eta}^{\text{P}}$; and (iii) event $\mathbf{E}_\eta$ holds. Recall that each one of the conditions (i), (ii) and (iii) holds with probability at least $1 - \eta$; hence, by a union bound we get:

$$\mathbb{P}(\mathbf{E}) \geq 1 - 3\eta = 1 - \delta.$$

This concludes the proof. □

Next, we lower bound the cumulative reward obtained by Algorithm 9.1 during the play phase. Intuitively, we show that, if the cumulative constraints violation is large, then the decisions $\boldsymbol{x}_t$ in the first T_1 rounds provide a per-round reward much higher than that achievable by $\boldsymbol{\xi}^*$. This allows us to employ the following recovery phase to decrease constraints violation cumulated in the play phase, while also ensuring that the cumulative regret stays low at the end of the algorithm. Formally:

Lemma 9.2. *If event* $\mathbf{E}$ *holds, then after round* T_1 *of Algorithm 9.1 the following inequality holds:* $\sum_{t=1}^{T_1} f_t(\boldsymbol{x}_t) \geq \sum_{t=1}^{T_1} f_t(\boldsymbol{\xi}^*) + (T - T_1) - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1,\eta} - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T_1,\eta}^{\text{P}} - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1}^{\text{D}}$.

Proof. By the no-regret property of the primal regret minimizer, we have that:

$$\begin{aligned}
&\sum_{t=1}^{T_1}\Big(f_t(\boldsymbol{x}_t) - \langle\boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle\Big) \geq \\
&\geq \sum_{t=1}^{T_1}\Big(f_t(\boldsymbol{\xi}^*) - \langle\boldsymbol{\lambda}_t, g_t(\boldsymbol{\xi}^*)\rangle\Big) - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T_1,\eta}^{\text{P}}. \qquad (9.8)
\end{aligned}$$

Let $i^\star \in \operatorname{argmax}_{i\in[m]} \sum_{t=1}^{T_1} g_{t,i}(\boldsymbol{x}_t)$ be one of the "most violated" constraints. We prove that:

$$\sum_{t=1}^{T_1} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle \geq (T - T_1) - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1}. \tag{9.9}$$

To do that, we consider the following two cases.

Case $T_1 = T$. We get:

$$\sum_{t=1}^{T_1} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle \geq \sum_{t=1}^{T_1} \langle \mathbf{0}, g_t(\boldsymbol{x}_t)\rangle - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1} = (T - T_1) - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1}.$$

Case $T_1 < T$. By the condition in Line 4 of Algorithm 9.1, we have that $\sum_{t=1}^{T_1} g_{t,i^\star}(\boldsymbol{x}_t) \geq (T - T_1)\tilde{\rho}$. Thus, we have that:

$$\sum_{t=1}^{T_1} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle \geq \sum_{t=1}^{T_1} \frac{1}{\tilde{\rho}} g_{t,i^\star}(\boldsymbol{x}_t) - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1} \geq (T - T_1) - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1},$$

where the first inequality follows from the no-regret property of the dual regret minimizer and the second one from the fact that $\sum_{t=1}^{T_1} g_{t,i^\star}(\boldsymbol{x}_t) \geq (T - T_1)\tilde{\rho}$ when $T_1 < T$.

Now, by using Equation (9.9), we can provide a lower bound on the cumulative reward obtained by Algorithm 9.1 during the play phase. We have that:

$$\begin{aligned}
&\sum_{t=1}^{T_1} f_t(\boldsymbol{x}_t) \geq \sum_{t=1}^{T_1} \Big(f_t(\boldsymbol{\xi}^*) - \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{\xi}^*)\rangle + \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle \Big) - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T_1,\eta}, \\
&\geq \sum_{t=1}^{T_1} \Big(f_t(\boldsymbol{\xi}^*) - \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{\xi}^*)\rangle \Big) + (T - T_1) - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1} \geq \\
&\geq \sum_{t=1}^{T_1} \Big(f_t(\boldsymbol{\xi}^*) - \langle \boldsymbol{\lambda}_t, \bar{g}(\boldsymbol{\xi}^*)\rangle \Big) + (T - T_1) - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \\
&\quad - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1} - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1,\eta} \geq \\
&\geq \sum_{t=1}^{T_1} f_t(\boldsymbol{\xi}^*) + (T - T_1) - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1} - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1,\eta},
\end{aligned}$$

where the first inequality holds by Equation (9.8), the second one by Equation (9.9), the third one follows from the fact that the event $\mathbf{E}$ holds, while the last one from the fact that $\bar{g}(\boldsymbol{\xi}^*) \leq 0$ by definition. □

In the recovery phase, the only goal of Algorithm 9.1 is to decrease constraints violation. In the following Lemma 9.3, we show that, at each round of the recovery phase, the algorithm is "close" to satisfying (in expectation) all the constraints by at least ρ. Formally:

Lemma 9.3. *If event* $\mathbf{E}$ *holds, then after Algorithm 9.1 halts, the following holds for every* $i \in [m]$*:* $\sum_{t=T_1+1}^{T} g_{t,i}(\boldsymbol{x}_t) \leq -(T - T_1)\rho + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}^{\mathrm{D}}_{T-T_1} + \mathcal{E}_{T-T_1,\eta}$.

Proof. Let $i^\star \in \operatorname{argmax}_{i\in[m]} \sum_{t=T_1+1}^{T} g_{t,i}(\boldsymbol{x}_t)$ be one of the "most violated" constraints. Then,

$$\begin{aligned}
(T - T_1)\rho &\leq -\sum_{t=T_1+1}^{T} \langle \boldsymbol{\lambda}_t, \bar{g}(\boldsymbol{\xi}^\circ)\rangle \\
&\leq -\sum_{t=T_1+1}^{T} \langle \boldsymbol{\lambda}_t g_t(\boldsymbol{\xi}^\circ)\rangle + \mathcal{E}_{T-T_1,\eta} \\
&\leq -\sum_{t=T_1+1}^{T} \langle \boldsymbol{\lambda}_t g_t(\boldsymbol{x}_t)\rangle + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}_{T-T_1,\eta} \\
&\leq -\sum_{t=T_1+1}^{T} g_{t,i^\star}(\boldsymbol{x}_t) + \mathcal{E}^{\mathrm{D}}_{T-T_1} + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}_{T-T_1,\eta},
\end{aligned}$$

where the first inequality comes from the definition of ρ, the second one from the fact that event $\mathbf{E}$ holds, the third one from the no-regret property of the primal regret minimizer, and the last one from the no-regret property of the dual regret minimizer. Hence,

$$\sum_{t=T_1+1}^{T} g_{t,i^\star}(\boldsymbol{x}_t) \leq -(T - T_1)\rho - \mathcal{E}^{\mathrm{D}}_{T-T_1} + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}_{T-T_1,\eta}. \quad (9.10)$$

It follows from the definition of $i^\star$ that, if Equation (9.10) holds for $i^\star$, then, it holds for every $i \in [m]$. This concludes the proof. □

Now, we are ready to present the two main results of this section. First, we provide a bound on the cumulative regret and constraints violation when the lower bound $\hat{\rho}$ is sufficiently large.

Condition 9.1. *It holds that* $\hat{\rho} \geq 2T^{-1/4}$.

Notice that, under Condition 9.1, $\tilde{\rho} = \hat{\rho}/2$. This gives us the following result:

Theorem 9.2. *Suppose that functions* f_t *and* g_t *are selected adversarially and stochastically, respectively. If Condition 9.1 is satisfied, then, with probability at least* $1-\delta$*, Algorithm 9.1 provides* $R^T \leq \frac{1}{\tilde{\rho}}\mathcal{E}_{T,\eta} + \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T,\eta} + \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T}$ *and* $V^T \leq M_{\tilde{\rho}} + 2\mathcal{E}^{\mathrm{P}}_{T,\eta} + \mathcal{E}^{\mathrm{D}}_{T} + \mathcal{E}_{T,\eta}$.

Proof. By Lemma 9.1, event $\mathbf{E}$ holds with probability at least $1-\delta$. In the rest of the proof, we assume the event $\mathbf{E}$ holds, providing a bound that holds with probability at least $1-\delta$.

We first provide an upper bound on the cumulative regret. By Lemma 9.2, we have:

$$\sum_{t=1}^{T_1} f_t(\boldsymbol{x}_t) \geq \sum_{t=1}^{T_1} f_t(\boldsymbol{\xi}^*) + (T - T_1) - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1,\eta} - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1}. \tag{9.11}$$

Hence, it holds:

$$\begin{aligned}
\sum_{t=1}^{T} f_t(\boldsymbol{x}_t) &\geq \sum_{t=1}^{T_1} f_t(\boldsymbol{x}_t) \\
&\geq \sum_{t=1}^{T_1} f_t(\boldsymbol{\xi}^*) + (T - T_1) - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1,\eta} - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1} \\
&\geq \sum_{t=1}^{T} f_t(\boldsymbol{\xi}^*) - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1,\eta} - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1} \\
&\geq \sum_{t=1}^{T} f_t(\boldsymbol{\xi}^*) - \frac{1}{\tilde{\rho}}\mathcal{E}_{T,\eta} - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T,\eta} - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T},
\end{aligned}$$

where the second inequality holds by Equation (9.11) and the third one by $\sum_{t=T_1+1}^{T} f_t(\boldsymbol{\xi}^*) \leq T - T_1$, which follows from the fact that the range of f_t is $[0,1]$.

By recalling that $\boldsymbol{\xi}^* \in \Xi$ is defined as an optimal solution to Problem $\mathtt{LP}_{\bar{f},\bar{g}}$ and $R^T = T\,\mathtt{OPT}_{\bar{f},\bar{g}} - \sum_{t=1}^{T} f_t(\boldsymbol{x}_t)$, the following bound on the cumulative regret holds:

$$R^T = \sum_{t=1}^{T} f_t(\boldsymbol{\xi}^*) - \sum_{t=1}^{T} f_t(\boldsymbol{x}_t) \leq \frac{1}{\tilde{\rho}}\mathcal{E}_{T,\eta} + \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T,\eta} + \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T}.$$

Next, we provide an upper bound on the cumulative constraints violation. By Lemma 9.3, for every $i \in [m]$, we have that:

$$\sum_{t=T_1+1}^{T} g_{t,i}(\boldsymbol{x}_t) \leq -(T-T_1)\rho + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}^{\mathrm{D}}_{T-T_1} + \mathcal{E}_{T-T_1,\eta}. \quad (9.12)$$

Hence, for every $i \in [m]$, it holds

$$\begin{aligned}
&\sum_{t=1}^{T} g_{t,i}(\boldsymbol{x}_t) = \sum_{t=1}^{T_1} g_{t,i}(\boldsymbol{x}_t) + \sum_{t=T_1+1}^{T} g_{t,i}(\boldsymbol{x}_t) \leq \\
&\leq (T-T_1)\tilde{\rho} + M_{\tilde{\rho}} - (T-T_1)\rho + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}^{\mathrm{D}}_{T-T_1} + \mathcal{E}_{T-T_1,\eta} \leq \\
&\leq M_{\tilde{\rho}} + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}^{\mathrm{D}}_{T-T_1} + \mathcal{E}_{T-T_1,\eta} \leq \\
&\leq M_{\tilde{\rho}} + 2\mathcal{E}^{\mathrm{P}}_{T,\eta} + \mathcal{E}^{\mathrm{D}}_{T} + \mathcal{E}_{T,\eta}.
\end{aligned}$$

The first inequality follows from Equation (9.12) and by the condition in Line 4 of Algorithm 9.1, which ensures $\sum_{t=1}^{T_1} g_{t,i}(\boldsymbol{x}_t) \leq (T-T_1)\tilde{\rho} + M_{\tilde{\rho}}$ for every $i \in [m]$. Moreover, the second inequality follows from $\tilde{\rho} \leq \rho$, since Condition 9.1 holds. Let $i^\star \in \operatorname{argmax}_{i\in[m]} \sum_{t=1}^{T} g_{t,i}(\boldsymbol{x}_t)$ be one of the most violated constraints. By recalling that $V^T = \max_{i\in[m]} \sum_{t=1}^{T} g_{t,i}(\boldsymbol{x}_t)$, the following bound on the cumulative constraints violation holds:

$$V^T = \sum_{t=1}^{T} g_{t,i^\star}(\boldsymbol{x}_t) \leq M_{\tilde{\rho}} + 2\mathcal{E}^{\mathrm{P}}_{T,\eta} + \mathcal{E}^{\mathrm{D}}_{T} + \mathcal{E}_{T,\eta}.$$

This concludes the proof. □

Finally, we also prove that even if Condition 9.1 is *not* satisfied, *i.e.*, the lower bound $\hat{\rho}$ is *not* sufficiently large, the following holds:

Theorem 9.3. *Suppose that functions f_t and g_t are selected adversarially and stochastically, respectively. Algorithm 9.1 guarantees that the following bounds hold with probability at least $1-\delta$: $R_T \leq T^{1/4}\mathcal{E}_{T,\eta} + \left(1 + 2T^{1/4}\right)\mathcal{E}^{\mathrm{P}}_{T,\eta} + T^{1/4}\mathcal{E}^{\mathrm{D}}_{T}$ and $V_T \leq T^{3/4} + M_{T^{-1/4}} + 2\mathcal{E}^{\mathrm{P}}_{T,\eta} + \mathcal{E}^{\mathrm{D}}_{T} + \mathcal{E}_{T,\eta}$.*

Proof. If $\hat{\rho} \geq 2T^{-1/4}$, the claim follows by Theorem 9.2. Thus, we prove the statement for the case $\tilde{\rho} = T^{-1/4}$. First, we provide an upper bound on the cumulative regret. By Lemma 9.1, we have that event the $\mathbf{E}$ holds with probability at least $1-\delta$. In the rest of the proof, we assume that the event $\mathbf{E}$ holds, and provide a bound that holds with probability at least $1-\delta$. We

have:

$$\begin{aligned}\sum_{t=1}^{T} f_t(\boldsymbol{x}_t) &\geq \sum_{t=1}^{T_1} f_t(\boldsymbol{x}_t) \\ &\geq \sum_{t=1}^{T_1} f_t(\boldsymbol{\xi}^*) + (T - T_1) - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1,\eta} - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1} \\ &\geq \sum_{t=1}^{T} f_t(\boldsymbol{\xi}^*) - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1,\eta} - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T_1} \\ &\geq \sum_{t=1}^{T} f_t(\boldsymbol{\xi}^*) - \frac{1}{\tilde{\rho}}\mathcal{E}_{T,\eta} - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{T,\eta} - \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{T} \\ &\geq \sum_{t=1}^{T} f_t(\boldsymbol{\xi}^*) - T^{1/4}\mathcal{E}_{T,\eta} - \left(1 + 2T^{1/4}\right)\mathcal{E}^{\mathrm{P}}_{T,\eta} - T^{1/4}\mathcal{E}^{\mathrm{D}}_{T}.\end{aligned}$$

These steps are similar to those used to prove the regret bound in Theorem 9.2 (see the proof of Theorem 9.2 for further details). By recalling that $\boldsymbol{\xi}^*$ is an optimal solution to Problem $\texttt{LP}_{\bar{f},\bar{g}}$ and $R^T = T\,\texttt{OPT}_{\bar{f},\bar{g}} - \sum_{t=1}^{T} f_t(\boldsymbol{x}_t)$, the following bound on the cumulative regret holds:

$$R^T = \sum_{t=1}^{T} f_t(\boldsymbol{\xi}^*) - \sum_{t=1}^{T} f_t(\boldsymbol{x}_t) \leq T^{1/4}\mathcal{E}_{T,\eta} + \left(1 + 2T^{1/4}\right)\mathcal{E}^{\mathrm{P}}_{T,\eta} + T^{1/4}\mathcal{E}^{\mathrm{D}}_{T}.$$

Next, we provide an upper bound on the cumulative constraints violation. For every $i \in [m]$, the following holds

$$\begin{aligned}\sum_{t=T_1+1}^{T} g_{t,i}(\boldsymbol{x}_t) &\leq -(T - T_1)\rho + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}^{\mathrm{D}}_{T-T_1} + \mathcal{E}_{T-T_1,\eta} \\ &\leq 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}^{\mathrm{D}}_{T-T_1} + \mathcal{E}_{T-T_1,\eta}, \qquad (9.13)\end{aligned}$$

where the first inequality follows from Lemma 9.3, while the second one from $\rho \geq 0$. Hence, for every $i \in [m]$, it holds

$$\begin{aligned}\sum_{t=1}^{T} g_{t,i}(\boldsymbol{x}_t) &= \sum_{t=1}^{T_1} g_{t,i}(\boldsymbol{x}_t) + \sum_{t=T_1+1}^{T} g_{t,i}(\boldsymbol{x}_t) \\ &\leq (T - T_1)\tilde{\rho} + M_{\tilde{\rho}} + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}^{\mathrm{D}}_{T-T_1} + \mathcal{E}_{T-T_1,\eta} \\ &\leq (T - T_1)T^{-1/4} + M_{T^{-1/4}} + 2\mathcal{E}^{\mathrm{P}}_{T-T_1,\eta} + \mathcal{E}^{\mathrm{D}}_{T-T_1} + \mathcal{E}_{T-T_1,\eta}\end{aligned}$$

$$\leq T^{3/4} + M_{T^{-1/4}} + 2\mathcal{E}^{\mathtt{P}}_{T-T_1,\eta} + \mathcal{E}^{\mathtt{D}}_{T-T_1} + \mathcal{E}_{T-T_1,\eta}$$
$$\leq T^{3/4} + M_{T^{-1/4}} + 2\mathcal{E}^{\mathtt{P}}_{T,\eta} + \mathcal{E}^{\mathtt{D}}_{T} + \mathcal{E}_{T,\eta}.$$

The first inequality follows from Equation (9.13) and from the condition in Line 4 of Algorithm 9.1, which ensures that $\sum_{t=1}^{T_1} g_{t,i}(\boldsymbol{x}_t) \leq (T-T_1)\tilde{\rho}+M_{\tilde{\rho}}$ for every $i \in [m]$. Moreover, the second inequality follows from $\tilde{\rho} = T^{-1/4}$. Thus, by letting $i^\star \in \operatorname{argmax}_{i\in[m]} \sum_{t=1}^{T} g_{t,i}(\boldsymbol{x}_t)$, and by recalling that $V^T = \max_{i\in[m]} \sum_{t=1}^{T} g_{t,i}(\boldsymbol{x}_t)$, the following bound on the cumulative constraints violation holds:

$$V^T = \sum_{t=1}^{T} g_{t,i^\star}(\boldsymbol{x}_t) \leq T^{3/4} + M_{T^{-1/4}} + 2\mathcal{E}^{\mathtt{P}}_{T,\eta} + \mathcal{E}^{\mathtt{D}}_{T} + \mathcal{E}_{T,\eta}.$$

This concludes the proof. □

Remark 9.3. *Notice that, by using primal and dual RMs whose regret bounds are of the order of $\tilde{O}(\sqrt{T})$, Theorem 9.2 allows us to recover $\tilde{O}(\sqrt{T}/\hat{\rho})$ regret and $\tilde{O}(\sqrt{T}/\hat{\rho})$ constraints violation for the case in which Condition 9.1 holds. Theorem 9.3 still provides $\tilde{O}(T^{3/4})$ regret and constraints violation when the condition is not met, which is necessary the case when $\rho = 0$.*

9.4 Analysis with stochastic constraints and stochastic rewards

In this section, we focus on the case in which both reward and constraint functions are selected stochastically. In this setting, we are able to show that Algorithm 9.1 never enters the recovery phase. As we argue in Section 9.7, this is an important property for budget-management applications, since it is related to the round in which the budget is fully depleted.

In order to prove our result, we extend the event $\mathbf{E}$ to capture also the Azuma-Hoeffding bounds for the reward functions, which are stochastic in this setting.[7] The core idea that we exploit to prove our result is that we can think of the two RMs as if they are playing a stochastic repeated zero-sum game, which is the repeated Lagrangian game whose functions are sampled according to the probability measures $\mu_{\mathcal{F}}$ and $\mu_{\mathcal{G}}$. By Theorem 9.1, strong duality holds, and the game has an equilibrium. Hence, it is possible to show that the per-round utility of the primal RM is close to the value of the game, which is $\mathtt{OPT}_{\bar{f},\bar{g}}$. At the same time, it is possible to show that, if the

[7] Accounting for the martingale difference sequences $f_t(\boldsymbol{x}_t) - \bar{f}(\boldsymbol{x}_t)$ and $f_t(\boldsymbol{\xi}^*) - \bar{f}(\boldsymbol{\xi}^*)$.

cumulative constraints violation becomes large during the play phase (and, thus, $T_1 < T$), then the per-round utility of the primal RM is below $\texttt{OPT}_{\bar{f},\bar{g}}$, reaching a contradiction that proves the following Theorem 9.4.

First, we provide a preliminary result on the value of the Lagrangian game when primal and dual players are constrained to specific sets of strategies.

Lemma 9.4. *Let $f \in \mathcal{F}$ and $g \in \mathcal{G}$ be such that $d_g > 0$. Moreover, given any $\epsilon > 0$, let $\Xi_{\epsilon,g} \coloneqq \left\{\boldsymbol{\xi} \in \Xi : \max_{i\in[m]} g_i(\boldsymbol{\xi}) \geq \epsilon\right\}$. The following holds:*

$$\sup_{\boldsymbol{\xi}\in\Xi_\epsilon} \inf_{\boldsymbol{\lambda}\in\mathcal{D}_{d_g/2}} \mathcal{L}_{f,g}(\boldsymbol{\xi},\boldsymbol{\lambda}) \leq \texttt{OPT}_{f,g} - \frac{\epsilon}{d_g}.$$

Proof. Let $\boldsymbol{\xi} \in \Xi_{\epsilon,g}$ and $i^\star \in \operatorname{argmax}_{i\in[m]} g_i(\boldsymbol{\xi})$. Then,

$$\begin{aligned}
\inf_{\boldsymbol{\lambda}\in\mathcal{D}_{d_g/2}} \Big\{f(\boldsymbol{\xi}) - \langle\boldsymbol{\lambda}, g(\boldsymbol{\xi})\rangle\Big\} &= f(\boldsymbol{\xi}) - \frac{2}{d_g}\, g_{i^\star}(\boldsymbol{\xi}) \\
&= \inf_{\boldsymbol{\lambda}\in\mathcal{D}_{d_g}} \Big\{f(\boldsymbol{\xi}) - \langle\boldsymbol{\lambda}, g(\boldsymbol{\xi})\rangle\Big\} - \frac{1}{d_g} g_{i^\star}(\boldsymbol{\xi}) \\
&\leq \sup_{\boldsymbol{\xi}\in\Xi} \inf_{\boldsymbol{\lambda}\in\mathcal{D}_{d_g}} \mathcal{L}_{f,g}(\boldsymbol{\xi},\boldsymbol{\lambda}) - \frac{1}{d_g} g_{i^\star}(\boldsymbol{\xi}) \\
&\leq \texttt{OPT}_{f,g} - \frac{1}{d_g} g_{i^\star}(\boldsymbol{\xi}) \\
&\leq \texttt{OPT}_{f,g} - \frac{\epsilon}{d_g},
\end{aligned}$$

where the second inequality follows from Theorem 9.1, while the last one holds by the definition of $\Xi_{\epsilon,g}$ and $i^\star$. □

Next, we introduce a new event that extends $\mathbf{E}$ by considering also the (stochastic) sequence of reward functions f_t. Formally, the event is defined as follows.

Definition 9.3. *We denote with $\bar{\mathbf{E}}$ the event in which Algorithm 9.1 satisfies the following conditions (recall that $\eta = \delta/3$): (i) event $\mathbf{E}$ holds; (ii) for every pair of rounds $t, t' \in [T] : t \leq t'$ it holds:*

- $|\sum_{\tau=t}^{t'} f_\tau(\boldsymbol{x}_\tau) - \sum_{\tau=t}^{t'} \bar{f}(\boldsymbol{x}_\tau)| \leq \mathcal{E}_{t'-t,\eta}$,
- $|\sum_{\tau=t}^{t'} f_\tau(\boldsymbol{\xi}^*) - \sum_{\tau=t}^{t'} \bar{f}(\boldsymbol{\xi}^*)| \leq \mathcal{E}_{t'-t,\eta}$.

Lemma 9.5. *After running Algorithm 9.1, the event $\bar{\mathbf{E}}$ holds with probability at least $1 - \delta$.*

Proof. Given a desired failure probability $\delta \in (0,1)$, recall that $\eta = \delta/3$ and set $\varepsilon = \eta/12mT^2$. Consider the following inequalities in which the differences between expectations and empirical means of reward functions are bounded:

$$\left|\sum_{\tau=t}^{t'} f_\tau(\boldsymbol{x}_\tau) - \sum_{\tau=t}^{t'} \bar{f}(\boldsymbol{x}_\tau)\right| > 2\sqrt{2(t'-t)\ln\frac{1}{\varepsilon}}, \tag{9.14}$$

$$\left|\sum_{\tau=t}^{t'} f_\tau(\boldsymbol{\xi}^*) - \sum_{\tau=t}^{t'} \bar{f}(\boldsymbol{\xi}^*)\right| > 2\sqrt{2(t'-t)\ln\frac{1}{\varepsilon}}. \tag{9.15}$$

By applying the Azuma-Hoeffding inequality to each martingale difference sequence, we get that each inequality holds with probability at most 2ε. We denote by $\bar{\mathbf{E}}_\eta$ the event in which Equations (9.14) and (9.15) hold for every $t \leq t' \in [T] : t < t'$ and event $\mathbf{E}_\eta$ holds (see the proof of Lemma 9.1 for the definition of event $\mathbf{E}_\eta$). By a union bound, we have that:

$$\mathbb{P}\big(\bar{\mathbf{E}}_\eta\big) \geq 1 - 2\varepsilon(4mT^2 + 2T^2) \geq 1 - \eta.$$

Therefore, event $\bar{\mathbf{E}}_\eta$ holds with probability at least $1-\eta$. Moreover, let us recall that:

$$\mathcal{E}_{t'-t,\eta} = \sqrt{8(t'-t)\ln\frac{12mT^2}{\eta}} = 2\sqrt{2(t'-t)\ln\frac{1}{\varepsilon}}.$$

Now, consider the event $\bar{\mathbf{E}}$ in which Algorithm 9.1 satisfies the following conditions: (i) the regret incurred by $\mathcal{R}_{\mathrm{I}}^{\mathrm{P}}$ after T_1 rounds is upper bounded by $\mathcal{E}_{T_1,\eta}^{\mathrm{P}}$; (ii) the regret cumulated by $\mathcal{R}_{\mathrm{II}}^{\mathrm{P}}$ after the remaining $T - T_1$ rounds is upper bounded by $\mathcal{E}_{T-T_1,\eta}^{\mathrm{P}}$; and (iii) event $\bar{\mathbf{E}}_\eta$ holds. Recall that each one of the conditions (i), (ii) and (iii) holds with probability at least $1-\eta$; hence, by a union bound we get:

$$\mathbb{P}\big(\bar{\mathbf{E}}\big) \geq 1 - 3\eta = 1 - \delta.$$

This concludes the proof. □

As a first step, we prove that the primal regret minimizer gets a per-round utility that is close to the value $\mathtt{OPT}_{\bar{f},\bar{g}}$. Formally:

Lemma 9.6. *If the event $\bar{\mathbf{E}}$ holds, then, for every round $\tau \in [T_1]$ the following inequality holds:*

$$\sum_{t=1}^{\tau} \mathcal{L}_{f_t,g_t}(\boldsymbol{x}_t, \boldsymbol{\lambda}_t) \geq \tau\, \mathtt{OPT}_{\bar{f},\bar{g}}^{\mathtt{LP}} - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta}^{\mathrm{P}} - \left(1 + \frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta}.$$

Proof. Let $\boldsymbol{\xi}^\star$ be an optimal solution to Problem $\texttt{LP}_{\bar{f},\bar{g}}$, and let $\bar{\boldsymbol{\lambda}} = \frac{1}{\tau}\sum_{t=1}^{\tau}\boldsymbol{\lambda}_t$. Then, it holds

$$\begin{aligned}
\sum_{t=1}^{\tau}\mathcal{L}_{f_t,g_t}(\boldsymbol{x}_t,\boldsymbol{\lambda}_t) &\geq \sum_{t=1}^{\tau}\mathcal{L}_{f_t,g_t}(\boldsymbol{\xi}^*,\boldsymbol{\lambda}_t) - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{\tau,\eta} \\
&\geq \sum_{t=1}^{\tau}\mathcal{L}_{\bar{f},\bar{g}}(\boldsymbol{\xi}^*,\boldsymbol{\lambda}_t) - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{\tau,\eta} - \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \\
&= \sum_{t=1}^{\tau}\mathcal{L}_{\bar{f},\bar{g}}(\boldsymbol{\xi}^*,\bar{\boldsymbol{\lambda}}) - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{\tau,\eta} - \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \\
&\geq \tau \inf_{\boldsymbol{\lambda}\in\mathcal{D}_{\tilde{\rho}}}\mathcal{L}_{\bar{f},\bar{g}}(\boldsymbol{\xi}^*,\boldsymbol{\lambda}) - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{\tau,\eta} - \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \\
&= \tau \sup_{\boldsymbol{\xi}\in\Xi}\inf_{\boldsymbol{\lambda}\in\mathcal{D}_{\tilde{\rho}}}\mathcal{L}_{\bar{f},\bar{g}}(\boldsymbol{\xi},\boldsymbol{\lambda}) - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{\tau,\eta} - \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \\
&= \tau\texttt{OPT}^{\texttt{LP}}_{\bar{f},\bar{g}} - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}^{\mathrm{P}}_{\tau,\eta} - \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta},
\end{aligned}$$

where the first inequality follows from the no-regret property of the primal regret minimizer, the second one from the definition of the event $\bar{\mathbf{E}}$, and the third one from the definition of $\boldsymbol{\xi}^*$. Moreover, the first equality follows from the fact that $\bar{g}$ is independent from t. This concludes the proof. □

Now, we show that the dual regret minimizer gets a per-round utility that is close to the value $\texttt{OPT}_{\bar{f},\bar{g}}$. Moreover, the attained utility increases by an additive factor proportional to the primal violation. This can be proved only in the setting with stochastic reward functions. Indeed, in this setting the primal and dual regret minimizers are playing a stochastic repeated zero-sum game that converges to an equilibrium. Notice that this is *not* true when the reward functions are adversarial.

Lemma 9.7. *If event $\bar{\mathbf{E}}$ holds and Condition 9.1 is satisfied, then for each $\tau \in [T_1]$ and each $i \in [m]$*

$$\sum_{t=1}^{\tau}\mathcal{L}_{f_t,g_t}(\boldsymbol{\xi}_t,\boldsymbol{\lambda}_t) \leq \tau\texttt{OPT}^{\texttt{LP}}_{\bar{f},\bar{g}} + \frac{1}{\tilde{\rho}}\mathcal{E}^{\mathrm{D}}_{\tau} + \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} - \sum_{t=1}^{\tau}g_{t,i}(\boldsymbol{x}_t).$$

Proof. In the following, let $\boldsymbol{\lambda}^* \in \arg\min_{\boldsymbol{\lambda}\in\mathcal{D}_{\tilde{\rho}}}\sum_{t=1}^{\tau}\mathcal{L}_{\bar{f},\bar{g}}(\boldsymbol{\xi}_t,\boldsymbol{\lambda})$, let $\epsilon \coloneqq \frac{\max_{i\in[m]}\sum_{t=1}^{\tau}g_{t,i}(\boldsymbol{x}_t)-\mathcal{E}_{\tau,\eta}}{\tau}$, and $\bar{\boldsymbol{\xi}} \coloneqq \frac{1}{\tau}\sum_{t=1}^{\tau}\boldsymbol{\xi}_t$, where $\boldsymbol{\xi}_t \in \Xi$ denotes the strategy mixture that plays deterministically $\boldsymbol{x}_t$. Moreover, let us define the set $\Xi_{\epsilon,\bar{g}} \coloneqq \{\boldsymbol{\xi}\in\Xi : \max_{i\in[m]}\bar{g}_i(\boldsymbol{\xi}) \geq \epsilon\}$.

As a first step, we prove that $\bar{\boldsymbol{\xi}} \in \Xi_{\epsilon,\bar{g}}$. In particular, since the event $\bar{\mathbf{E}}$ holds, we have that

$$\max_{i\in[m]} \bar{g}_i(\bar{\boldsymbol{\xi}}) \geq \frac{\sum_{t=1}^{\tau} \max_{i\in[m]} g_i(\bar{\boldsymbol{\xi}}) - \mathcal{E}_{\tau,\eta}}{\tau} = \epsilon$$

For every $\tau \in [T_1]$, we have:

$$\sum_{t=1}^{\tau} \mathcal{L}_{f_t,g_t}(\boldsymbol{x}_t, \boldsymbol{\lambda}_t) \leq \sum_{t=1}^{\tau} \mathcal{L}_{f_t,g_t}(\boldsymbol{x}_t, \boldsymbol{\lambda}^*) + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} \tag{9.16a}$$

$$\leq \sum_{t=1}^{\tau} \mathcal{L}_{\bar{f},\bar{g}}(\boldsymbol{x}_t, \boldsymbol{\lambda}^*) + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \tag{9.16b}$$

$$\leq \inf_{\boldsymbol{\lambda}\in\mathcal{D}_{\tilde{\rho}}} \sum_{t=1}^{\tau} \mathcal{L}_{\bar{f},\bar{g}}(\boldsymbol{x}_t, \boldsymbol{\lambda}) + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \tag{9.16c}$$

$$= \tau \inf_{\boldsymbol{\lambda}\in\mathcal{D}_{\tilde{\rho}}} \mathcal{L}_{\bar{f},\bar{g}}(\bar{\boldsymbol{\xi}}, \boldsymbol{\lambda}) + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \tag{9.16d}$$

$$= \tau \inf_{\boldsymbol{\lambda}\in\mathcal{D}_{\rho/2}} \mathcal{L}_{\bar{f},\bar{g}}(\bar{\boldsymbol{\xi}}, \boldsymbol{\lambda}) + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \tag{9.16e}$$

$$\leq \tau \sup_{\boldsymbol{\xi}\in\Xi_{\epsilon,\bar{g}}} \inf_{\lambda\in\mathcal{D}_{\rho/2}} \mathcal{L}_{\bar{f},\bar{g}}(\boldsymbol{\xi}, \boldsymbol{\lambda}) + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \tag{9.16f}$$

$$\leq \tau \sup_{\boldsymbol{\xi}\in\Xi} \inf_{\boldsymbol{\lambda}\in\mathcal{D}_{\rho}} \left(\mathcal{L}_{\bar{f},\bar{g}}(\boldsymbol{\xi}, \boldsymbol{\lambda}) - \frac{\epsilon}{\rho}\right) + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \tag{9.16g}$$

$$= \tau\left(\mathtt{OPT}_{\bar{f},\bar{g}} - \frac{\epsilon}{\rho}\right) + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \tag{9.16h}$$

$$= \tau\mathtt{OPT}_{\bar{f},\bar{g}} - \tau\frac{\max_{i'\in[m]} \sum_{t=1}^{\tau} g_{t,i'}(\boldsymbol{x}_t) - \mathcal{E}_{\tau,\eta}}{\tau\rho} + \\ + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \tag{9.16i}$$

$$\leq \tau\mathtt{OPT}_{\bar{f},\bar{g}} + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} - \max_{i'\in[m]} \frac{\sum_{t=1}^{\tau} g_{t,i'}(2\boldsymbol{x}_t)}{\rho} \tag{9.16j}$$

$$\leq \tau\mathtt{OPT}_{\bar{f},\bar{g}} + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\mathrm{D}} + \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} - \max_{i'\in[m]} \sum_{t=1}^{\tau} g_{t,i'}(\boldsymbol{x}_t) \tag{9.16k}$$

$$\leq \tau \texttt{OPT}_{\bar{f},\bar{g}} + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\texttt{D}} + \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} - \sum_{t=1}^{\tau} g_{t,i}(\boldsymbol{x}_t) \forall i \in [m], \tag{9.16l}$$

where Equation (9.16a) is given by the no-regret property of the dual regret minimizer, and Equation (9.16b) by the definition of the event $\bar{\mathbf{E}}$, which holds by assumption. Moreover, Equation (9.16d) follows from the fact that $\bar{f}$ and $\bar{g}$ are independent from t, Equation (9.16e) follows from $\tilde{\rho} = \hat{\rho}/2 \leq \rho/2$, and Equation (9.16f) from $\bar{\boldsymbol{\xi}} \in \Xi_{\epsilon,\bar{g}}$. Finally, Equation (9.16g) follows from Lemma 9.4, Equation (9.16i) by definition of ϵ, and Equation (9.16j) by $\tilde{\rho} \leq \rho$. □

Now, we are ready to prove the main result of this section.

Theorem 9.4. *Suppose that functions f_t and g_t are selected stochastically. With probability at least $1-\delta$, Algorithm 9.1 never enters the recovery phase, namely $T_1 = T$.*

Proof. We prove the statement of the theorem by considering two cases.

Case "Condition 9.1 holds". By Lemma 9.1, event $\mathbf{E}$ holds with probability at least $1 - \delta$. In the rest of the proof, we assume that the event $\mathbf{E}$ holds, and we provide a bound that holds with probability at least $1 - \delta$. For every $\tau \in [T_1]$, we have:

$$\begin{aligned}
\sum_{t=1}^{\tau} g_t(\boldsymbol{x}_t) &\leq \tau \texttt{OPT}_{\bar{f},\bar{g}} - \sum_{t=1}^{\tau} \mathcal{L}_{f_t,g_t}(\boldsymbol{x}_t, \boldsymbol{\lambda}_t) + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\texttt{D}} + \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \\
&\leq \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta}^{\texttt{P}} + \left(1 + \frac{1}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\texttt{D}} + \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} \\
&= \left(2 + \frac{3}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} + \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta}^{\texttt{P}} + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\texttt{D}} \\
&\leq \frac{2}{\tilde{\rho}}\sqrt{T} - 1 + \left(2 + \frac{3}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta} + \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{\tau,\eta}^{\texttt{P}} + \frac{1}{\tilde{\rho}}\mathcal{E}_{\tau}^{\texttt{D}} \\
&= M_{\tilde{\rho}} - 1,
\end{aligned}$$

where the first inequality follows from Lemma 9.6, the second one from Lemma 9.7, the third one from the fact that $\frac{2}{\tilde{\rho}}\sqrt{T} - 1 \geq 0$, being $\tilde{\rho} \leq 1$, and the last equation follows from the definition of $M_{\tilde{\rho}}$. This implies that the algorithm never enters the recovery phase when Condition 9.1 holds.

Case "Condition 9.1 does *not* hold". By Lemma 9.1, event $\mathbf{E}$ holds with probability at least $1-\delta$. In the rest of the proof, we assume that the event $\mathbf{E}$ holds, and we provide a bound that holds with probability at least $1-\delta$. Suppose by contradiction that $T_1 < T$. This implies that a constraint $i \in [m]$ is violated by at least $M_{T^{-1/4}} - 1$. Let $i^\star \in \operatorname{argmax}_{i\in[m]} \sum_{t=1}^{T_1} g_{t,i}(\boldsymbol{x}_t)$ be one of the most violated constraints during the play phase. Then, we have:

$$\begin{aligned}
\sum_{t=1}^{T_1} \mathcal{L}_{f_t,g_t}(\boldsymbol{x}_t, \boldsymbol{\lambda}_t) &= \sum_{t=1}^{T_1} \Big(f(\boldsymbol{x}_t) - \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t) \rangle \Big) \\
&\leq T_1 - \sum_{t=1}^{T_1} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t) \rangle \\
&\leq T_1 - \sum_{t=1}^{T_1} \frac{1}{T^{-1/4}} g_{t,i^\star}(\boldsymbol{x}_t) + T^{1/4} \mathcal{E}^{\mathrm{D}}_{T_1} \\
&\leq T_1 - T^{1/4}(M_{T^{-1/4}} - 1) + T^{1/4} \mathcal{E}^{\mathrm{D}}_{\tau_1} \\
&< -\left(1 + \frac{2}{T^{-1/4}}\right) \mathcal{E}^{\mathrm{P}}_{\tau,\eta} - \frac{1}{T^{-1/4}} \mathcal{E}_{\tau,\eta},
\end{aligned}$$

where the second inequality follows from the no-regret property of the dual regret minimizer and the fact that, when Condition 9.1 does *not* hold, $\tilde{\rho} = T^{-1/4}$. The last inequality follows from the definition of $M_{T^{-1/4}}$. Then, the result above allows us to reach the desired contradiction when compared with the following one. In particular, for every $\tau \in [T_1]$, we have:

$$\begin{aligned}
\sum_{t=1}^{\tau} \mathcal{L}_{f_t,g_t}(\boldsymbol{x}_t, \boldsymbol{\lambda}_t) &\geq \sum_{t=1}^{\tau} \mathcal{L}_{f_t,g_t}(\boldsymbol{\xi}^\circ, \boldsymbol{\lambda}_t) - \left(1 + \frac{2}{T^{-1/4}}\right) \mathcal{E}^{\mathrm{P}}_{\tau,\eta} \\
&\geq \sum_{t=1}^{\tau} \mathcal{L}_{f_t,\bar{g}}(\boldsymbol{\xi}^\circ, \boldsymbol{\lambda}_t) - \frac{1}{T^{-1/4}} \mathcal{E}_{\tau,\eta} - \left(1 + \frac{2}{T^{-1/4}}\right) \mathcal{E}^{\mathrm{P}}_{\tau,\eta} \\
&\geq -\frac{1}{T^{-1/4}} \mathcal{E}_{\tau,\eta} - \left(1 + \frac{2}{T^{-1/4}}\right) \mathcal{E}^{\mathrm{P}}_{\tau,\eta},
\end{aligned}$$

where the first inequality follows from the no-regret property of the primal regret minimizer, the second one follows from the fact that event $\mathbf{E}$ holds, and the third one from the feasibility of $\boldsymbol{\xi}^\circ$. □

Notice that regret bounds analogous to the one in Theorems 9.2 and 9.3 also hold in the case in which both reward and constraint functions are selected stochastically.

9.5 Analysis with adversarial constraints

In this section, we study settings in which the constraint functions g_t are selected adversarially. As shown by Mannor et al. (2009), it is impossible to obtain sublinear cumulative regret and constraints violation when using our baseline, *i.e.*, the best fixed strategy mixture $\boldsymbol{\xi}^*$ satisfying (in expectation) the long-tern constraints. However, we show that it is possible to achieve a $\rho/(1+\rho)$ fraction of the cumulative reward obtained by always playing $\boldsymbol{\xi}^*$, while guaranteeing sublinear constraints violation. The dependence of the approximation factor on the feasibility parameter ρ is similar to the dependence on the per-round budget in problems with budget constraints (see the related works in Chapter 9 for more details). Moreover, as we discuss later in Section 9.7, when restricted to the case of budget constraints and adversarial reward/cost functions, our approximation factor matches the state-of-the-art bounds provided by Castiglioni et al. (2022a).

As a first step to prove our result, we provide a lower bound on the cumulative reward of the primal RM during the play phase. In particular, we show that it achieves at least a $\rho/(1+\rho)$ fraction of the value obtained by the optimal solution in the first T_1 rounds.

Lemma 9.8. *If Condition* 9.1 *is satisfied, then, with probability at least* $1-\eta$*, at round* T_1 *of Algorithm 9.1 it holds that:*

$$\sum_{t=1}^{T_1} f_t(\boldsymbol{x}_t) \geq \frac{\rho}{1+\rho} \sum_{t=1}^{T_1} f_t(\boldsymbol{\xi}^*) + (T - T_1) - \left(1 + \frac{2}{\tilde{\rho}}\right) \mathcal{E}^P_{T_1,\eta} - \frac{1}{\tilde{\rho}} \mathcal{E}^D_{\tau_1}.$$

Proof. Let $\bar{\boldsymbol{\xi}} \in \Xi$ be a strategy mixture obtained by playing with probability $1/(1+\rho)$ the mixture $\boldsymbol{\xi}^\circ$ and with the remaining probability $\rho/(1+\rho)$ an optimal mixture $\boldsymbol{\xi}^*$. Notice that the probabilities are well defined, since $\rho \geq 0$. Then, for every $t \in [T]$ and $i \in [m]$, it holds:

$$\frac{1}{1+\rho} g_{t,i}(\boldsymbol{\xi}^\circ) + \frac{\rho}{1+\rho} g_{t,i}(\boldsymbol{\xi}^*) \leq -\frac{\rho}{1+\rho} + \frac{\rho}{1+\rho} = 0$$

where the inequality follows from the fact that $g_{t,i}(\boldsymbol{\xi}^\circ) \leq -\rho$ and $g_{t,i}(\boldsymbol{\xi}^*) \leq 1$. Therefore, for every $t \in [T]$ and $i \in [m]$, it holds that $g_t(\bar{\boldsymbol{\xi}}) \leq 0$. Assume that the regret bounds of the regret minimizers hold. Notice that this happens with probability at least $1-\eta$. Then, by the no-regret property of the primal regret minimizer, we have that

$$\sum_{t=1}^{T_1} \mathcal{L}_{f_t,g_t}(\boldsymbol{x}_t, \boldsymbol{\lambda}_t) \geq \sum_{t=1}^{T_1} \mathcal{L}_{f_t,g_t}(\bar{\boldsymbol{\xi}}, \boldsymbol{\lambda}_t) - \left(1 + \frac{2}{\tilde{\rho}}\right) \mathcal{E}^{\mathrm{P}}_{T_1,\eta}. \tag{9.17}$$

Let $i^\star \in \operatorname{argmax}_{i\in[m]} \sum_{t=1}^{T_1} g_{t,i}(\boldsymbol{x}_t)$ be one of the most violated constraints during the play phase. Next, we prove that

$$\sum_{t=1}^{T_1} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle \geq (T - T_1) - \frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{T_1}.$$

We consider two cases. If $T_1 = T$, then

$$\sum_{t=1}^{T_1} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle \geq \sum_{t=1}^{T_1} \langle \mathbf{0}, g_t(\bar{\boldsymbol{\xi}})\rangle - \frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{T_1} = -\frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{\tau_2} = (T - T_1) - \frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{T_1}.$$

Otherwise, we have that $\sum_{t=1}^{T_1} g_{t,i^\star}(\boldsymbol{x}_t) \geq \tilde{\rho}(T - T_1)$ and

$$\sum_{t=1}^{T_1} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle \geq \left(\sum_{t=1}^{T_1} \frac{1}{\tilde{\rho}} g_{t,i^\star}(\boldsymbol{x}_t) \right) - \frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{T_1} \geq (T - T_1) - \frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{T_1}. \tag{9.18}$$

Thus,

$$\begin{aligned}
\sum_{t=1}^{T_1} f_t(\boldsymbol{x}_t) &\geq \sum_{t=1}^{T_1} \Big(f_t(\bar{\boldsymbol{\xi}}) - \langle \boldsymbol{\lambda}_t, g_t(\bar{\boldsymbol{\xi}})\rangle + \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle \Big) - \left(1 + \frac{2}{\tilde{\rho}}\right) \mathcal{E}^{\mathrm{P}}_{T_1,\eta} \\
&\geq \sum_{t=1}^{T_1} \Big(f_t(\bar{\boldsymbol{\xi}}) - \langle \boldsymbol{\lambda}_t, g_t(\bar{\boldsymbol{\xi}})\rangle \Big) + (T - T_1) - \left(1 + \frac{2}{\tilde{\rho}}\right) \mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{\tau_2} \\
&\geq \sum_{t=1}^{T_1} f_t(\bar{\boldsymbol{\xi}}) + (T - T_1) - \left(1 + \frac{2}{\tilde{\rho}}\right) \mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{\tau_2} \\
&\geq \sum_{t=1}^{T_1} \left(\frac{1}{1+\rho} f_t(\boldsymbol{\xi}^\circ) + \frac{\rho}{1+\rho} f_t(\boldsymbol{\xi}^*) \right) + (T - T_1) - \left(1 + \frac{2}{\tilde{\rho}}\right) \mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \\
&\quad - \frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{\tau_2} \\
&\geq \frac{\rho}{1+\rho} \sum_{t=1}^{T_1} f_t(\boldsymbol{\xi}^*) + (T - T_1) - \left(1 + \frac{2}{\tilde{\rho}}\right) \mathcal{E}^{\mathrm{P}}_{T_1,\eta} - \frac{1}{\tilde{\rho}} \mathcal{E}^{\mathrm{D}}_{T_1},
\end{aligned}$$

where the first inequality follows from Equation (9.17), the second one from Equation 9.18, the third one from the fact that for each $t \in [T]$ it holds $g_t(\bar{\boldsymbol{\xi}}) \leq 0$, while the fourth inequality follows from the definition of $\bar{\boldsymbol{\xi}}$. This concludes the proof. □

Notice that, for a small T_1, we have a large lower bound on the cumulative reward. Intuitively, this means that when the play phase is short, the primal regret minimizer accumulated so much regret in the play phase that the recovery phase can be addressed without worrying about the reward.

As a second step, we provide an upper bound on the cumulative constraints violation during the recovery phase. In particular, we show that the constraints are satisfied by at least ρ at each round up to a term related to the regret of $\mathcal{R}_{\text{II}}^{\text{P}}$ and $\mathcal{R}_{\text{II}}^{\text{D}}$.

Lemma 9.9. *With probability at least $1-\eta$, when Algorithm 9.1 halts it holds that for each $i \in [m]$:*

$$\sum_{t=T_1+1}^{T} g_{t,i}(\boldsymbol{x}_t) \leq -(T-T_1)\rho + \mathcal{E}_{T-T_1}^{D} + 2\mathcal{E}_{T-T_1,\eta}^{P}.$$

Proof. Let $i^\star$ be one of the most violated constraints:

$$i^\star \in \operatorname*{argmax}_{i\in[m]} \sum_{t=T_1+1}^{T} g_{t,i}(\boldsymbol{x}_t).$$

Then, we have that:

$$\begin{aligned}
(T-\tau)\rho &\leq -\sum_{t=T_1+1}^{T} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{\xi}^\circ)\rangle \\
&\leq -\sum_{t=T_1+1}^{T} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle + 2\mathcal{E}_{T-T_1,\eta}^{\text{P}} \\
&\leq -\sum_{t=T_1+1}^{T} g_{t,i^\star}(\boldsymbol{x}_t) + \mathcal{E}_{T-T_1}^{\text{D}} + 2\mathcal{E}_{T-T_1,\eta}^{\text{P}},
\end{aligned}$$

where the first inequality follows from the definition of $\boldsymbol{\xi}^\circ$ and the fact that it is always feasible of at least ρ, the second one follows from the assumption that the primal regret minimizer satisfies the regret bound, and the last inequality from the guarantee on the regret of the dual regret minimizer. We conclude the proof by noticing that the regret bound of the primal regret minimizer holds with probability at least $1-\eta$. □

Now, we can provide our bounds for adversarial constraints.

Theorem 9.5. *Suppose that functions f_t and g_t are selected adversarially. If Condition 9.1 is satisfied, then, with probability at least $1-\frac{2}{3}\delta$, Algorithm*

9.1 guarantees that the following holds: $\sum_{t=1}^{T} f_t(\boldsymbol{x}_t) \geq \frac{\rho}{1+\rho} \sum_{t=1}^{T} \texttt{OPT}_{\bar{f},\bar{g}} - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T,\eta}^{\texttt{P}} - \frac{1}{\tilde{\rho}}\mathcal{E}_{T}^{\texttt{D}}$ *and* $V^T \leq M_{\tilde{\rho}} + 2\mathcal{E}_{T,\eta}^{\texttt{P}} + \mathcal{E}_{T}^{\texttt{D}}$.

Proof. In the following, we assume that both Lemma 9.8 and Lemma 9.9. By an union bound, this holds with probability $1 - 2\eta = 1 - \frac{2}{3}\delta$. Then, it holds

$$\begin{aligned}
\sum_{t=1}^{T} f_t(\boldsymbol{x}_t) &\geq \sum_{t=1}^{T_1} f_t(\boldsymbol{x}_t) \\
&\geq \sum_{t=1}^{T_1} \frac{\rho}{1+\rho} f_t(\boldsymbol{\xi}^\star) + (T - T_1) - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T_1,\eta}^{\texttt{P}} - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1}^{\texttt{D}} \\
&\geq \frac{\rho}{1+\rho} \sum_{t=1}^{T} f_t(\boldsymbol{\xi}^\star) - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T_1,\eta}^{\texttt{P}} - \frac{1}{\tilde{\rho}}\mathcal{E}_{T_1}^{\texttt{D}} \\
&\geq \frac{\rho}{1+\rho} \sum_{t=1}^{T} f_t(\boldsymbol{\xi}^\star) - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T,\eta}^{\texttt{P}} - \frac{1}{\tilde{\rho}}\mathcal{E}_{T}^{\texttt{D}},
\end{aligned}$$

where the second inequality comes from Lemma 9.8. This proves the bound on the regret.

By Lemma 9.9, for each $i \in [m]$,

$$\begin{aligned}
\sum_{t=1}^{T} g_{t,i}(\boldsymbol{x}_t) &= \sum_{t=1}^{T_1} g_{t,i}(\boldsymbol{x}_t) + \sum_{t=T_1+1}^{T} g_{t,i}(\boldsymbol{x}_t) \\
&\leq (T - T_1)\tilde{\rho} + M_{\tilde{\rho}} - (T - T_1)\rho + \mathcal{E}_{T-T_1}^{\texttt{D}} + 2\mathcal{E}_{T-T_1,\eta}^{\texttt{P}} \\
&\leq M_{\tilde{\rho}} + \mathcal{E}_{T-T_1}^{\texttt{D}} + 2\mathcal{E}_{T-T_1,\eta}^{\texttt{P}} \\
&\leq M_{\tilde{\rho}} + \mathcal{E}_{T}^{\texttt{D}} + 2\mathcal{E}_{T,\eta}^{\texttt{P}},
\end{aligned}$$

where the second inequality comes from $\tilde{\rho} \leq \rho$.

□

A similar result can be also derived for the case of stochastic rewards and adversarial constraints.

Corollary 9.2. *Suppose functions* f_t *and* g_t *are selected stochastically and adversarially, respectively. If Condition 9.1 is satisfied, then, with probability at least* $1-\delta$*, Algorithm 9.1 provides* $\sum_{t=1}^{T} f_t(\boldsymbol{x}_t) \geq \frac{\rho}{1+\rho} \sum_{t=1}^{T} \texttt{OPT}_{\bar{f},\bar{g}} - \left(1 + \frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T,\eta}^{\texttt{P}} - \frac{1}{\tilde{\rho}}\mathcal{E}_{T}^{\texttt{D}} - 2\mathcal{E}_{T,\eta}$ *and* $V^T \leq M_{\tilde{\rho}} + 2\mathcal{E}_{T,\eta}^{P} + \mathcal{E}_{T}^{D} + \mathcal{E}_{T,\eta}$.

Proof. It is easy to see that Theorem 9.5 can be extended to consider the definition of $\boldsymbol{\xi}^\star$ for stochastic rewards. formally, it holds $\sum_{t=1}^T f_t(x_t) \geq \frac{\rho}{1+\rho}\sum_{t=1}^T f_t(\boldsymbol{\xi}^\star) - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T,\eta}^{\mathtt{P}} - \frac{1}{\tilde{\rho}}\mathcal{E}_T^{\mathtt{D}}$. Consider the two martingale difference sequences $\sum_{t=1}^T f_t(\boldsymbol{x}_t) - \bar{f}(\boldsymbol{x}_t)$ and $\sum_t f_t(\boldsymbol{\xi}^\star) - \bar{f}(\boldsymbol{\xi}^\star)$. We can apply Azuma-Hoeffding inequality to prove that, with probability at least $1-\eta$, it holds $\sum_t |f_t(\boldsymbol{x}_t) - \bar{f}(\boldsymbol{x}_t)| \leq \mathcal{E}_{T,\eta}$ and $\sum_t |f_t(\boldsymbol{\xi}^\star) - \bar{f}(\boldsymbol{\xi}^\star)| \leq \mathcal{E}_{T,\eta}$. Then,

$$\begin{aligned}\sum_{t=1}^T \bar{f}(\boldsymbol{x}_t) &\geq \sum_{t=1}^T f_t(\boldsymbol{x}_t) - \mathcal{E}_{T,\eta} \\ &\geq \frac{\rho}{1+\rho}\sum_{t=1}^T f_t(\boldsymbol{\xi}^\star) - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T,\eta}^{\mathtt{P}} - \frac{1}{\tilde{\rho}}\mathcal{E}_T^{\mathtt{D}} - \mathcal{E}_{T,\eta} \\ &\geq \frac{\rho}{1+\rho}\sum_{t=1}^T \bar{f}(\boldsymbol{\xi}^\star) - \left(1+\frac{2}{\tilde{\rho}}\right)\mathcal{E}_{T,\eta}^{\mathtt{P}} - \frac{1}{\tilde{\rho}}\mathcal{E}_T^{\mathtt{D}} - 2\mathcal{E}_{T,\eta},\end{aligned}$$

proving the statement. □

Remark 9.4. *By using primal and dual RMs whose regret bounds are of the order of $\tilde{O}(\sqrt{T})$, Theorem 9.5 and Corollary 9.2 allows us to recover $\sum_{t=1}^T f_t(\boldsymbol{x}_t) \geq \frac{\rho}{1+\rho}\sum_{t=1}^T \mathtt{OPT}_{\bar{f},\bar{g}} - \tilde{O}(\sqrt{T}/\hat{\rho})$, and $\tilde{O}(\sqrt{T}/\hat{\rho})$ constraints violation for the case in which Condition 9.1 holds.*

9.6 How to get away with no knowledge about the feasibility parameter

We show how to extend Algorithm 9.1 in order to deal with settings in which a lower bound on the feasibility parameter ρ is *not* known. Indeed, we propose an algorithm (Algorithm 9.3) that directly runs Algorithm 9.1, by first devoting a given number $T_0 < T$ of rounds to inferring a suitable lower bound $\hat{\rho}$ on the feasibility parameter ρ. Ideally, we would like to have $\hat{\rho} = \Omega(\rho)$, so that, we recover bounds of the order $\tilde{O}(\sqrt{T}/\rho)$. In particular, we show that we can run Algorithm 9.3 with $T_0 = T^{1/2}$ in order to recover an approximation of ρ that has an additive approximation error of the order $T^{1/4}$. This is sufficient to get $\hat{\rho} = \Omega(\rho)$, since a good approximation of ρ is only needed when $\rho \geq T^{1/4}$.[8]

Let us remark that our approach only works when constraints functions g_t are selected stochastically. When these are chosen adversarially, it is

[8] Notice that Algorithm 9.3 is *not* an explore and exploit algorithm. Indeed, it uses the exploration rounds only to have a rough estimate of ρ.

easy to see that it is impossible to compute a lower bound on the feasibility parameter ρ by only using the first rounds. For instance, think of a setting in which ρ is very large by only considering the first rounds, while it becomes small during later rounds.

Algorithm 9.3 META-ALGORITHM(T, T_0, δ)

$\mathcal{R}^{\text{P}} \leftarrow \text{INIT}^{\text{P}}\left(\mathcal{X}, \big[-1, 1\big], \delta\right)$
$\mathcal{R}^{\text{D}} \leftarrow \text{INIT}^{\text{D}}\left(\Delta_m, [-1, 1], 0\right)$
$t \leftarrow 1$
while $t \leq T_0$: **do**
    $\boldsymbol{x}_t \leftarrow \text{LAGRANGIANGAME}(\mathcal{R}^{\text{P}}, \mathcal{R}^{\text{D}}, 0)$
    $t \leftarrow t + 1$
end while
$\hat{\rho} \leftarrow -\frac{1}{T_0}\left(\max_{i\in[m]} \sum_{t=1}^{T_0} g_{t,i}(\boldsymbol{x}_t) + \mathcal{E}_{T_0,\delta}\right)$
Run Algorithm 9.1 with $T - T_0$, δ, and $\hat{\rho}$ as inputs

In order to exploit the guarantees of Algorithm 9.1 presented in the previous sections, it is enough to show that, after the first T_0 rounds of Algorithm 9.3, $\hat{\rho} \leq \rho$ holds with high probability.

Lemma 9.10. *By setting $T_0 = \sqrt{T}$, after T_0 rounds of Algorithm 9.3 we have that $\hat{\rho} \leq \rho$ with probability at least $1 - \delta$.*

Proof. By Azuma-Hoeffding inequality, we have that with probability at least $1 - \delta$, for each $i \in [m]$ it holds $\left| \sum_{t=1}^{T_0} g_{t,i}(\boldsymbol{x}_t) - \bar{g}_i(\boldsymbol{x}_t) \right|$. Hence,

$$\begin{aligned} -\max_{i\in[m]} \sum_{t=1}^{T_0} g_t(\boldsymbol{x}_t) &\leq -\max_{i\in[m]} \sum_{t=1}^{T_0} \bar{g}(\boldsymbol{x}_t) + \mathcal{E}_{T_0,\delta} \\ &\leq T_0 \bar{g}(\boldsymbol{\xi}^\circ) + \mathcal{E}_{T_0,\delta} \\ &= T_0 \rho + \mathcal{E}_{T_0,\delta}, \end{aligned}$$

where the second and third inequality follow from the definition of $\boldsymbol{\xi}^\circ$. Then,

$$\begin{aligned} \hat{\rho} &= -\frac{1}{T_0}\left(\max_{i\in[m]} \sum_{t=1}^{T_0} g_{t,i}(\boldsymbol{x}_t) + \mathcal{E}_{T_0,\delta}\right) \\ &\leq \frac{1}{T_0}\left(T_0 \rho + \mathcal{E}_{T_0,\delta} - \mathcal{E}_{T_0,\delta}\right) \\ &= \rho. \end{aligned}$$

This concludes the proof. □

To recover a good estimate of ρ, we need the value of ρ to be sufficiently large. Formally, we consider the following condition.[9]

Condition 9.2. *It holds that* $\rho \geq \frac{2}{T_0}\left(2\mathcal{E}_{T_0,\delta} + 2\mathcal{E}^{P}_{T_0,\delta} + \mathcal{E}^{D}_{T_0}\right)$.

Remark 9.5. *Notice that, by using primal and dual RMs whose regret bounds are of the order* $\tilde{O}(\sqrt{T})$*, and setting* $T_0 = \sqrt{T}$ *Condition 9.2 is satisfied when* $\rho = \omega(T^{-1/4})$.

Next, we show that $\hat{\rho} = \Omega(\rho)$, which allows us to exploit the guarantees proved for Algorithm 9.1 in order to provide analogous ones for Algorithm 9.3. Formally:

Lemma 9.11. *By setting* $T_0 = \sqrt{T}$*, and assuming that Condition 9.2 is satisfied, after* T_0 *rounds of Algorithm 9.3 we have that* $\hat{\rho} \geq \rho/2$ *with probability at least* $1 - 2\delta$.

Proof. First, notice that with probability $1-\delta$, the primal regret minimizer has regret bounded by $\mathcal{E}^{\mathrm{P}}_{T_0,\delta}$. Moreover, by the Azuma-Hoeffding inequality, it holds $\left|\sum_{t=1}^{T_0} \boldsymbol{\lambda}_t g_t(\boldsymbol{\xi}^\circ) - \boldsymbol{\lambda}_t \bar{g}(\boldsymbol{\xi}^\circ)\right| \leq \mathcal{E}_{T_0,\delta}$ with probability $1-\delta$. Consider the case in which both the conditions hold. This happens with probability at least $1-2\delta$ by a union bound.

Then,

$$\begin{aligned}
-\max_{i\in[m]} \sum_{t=1}^{T_0} g_t(\boldsymbol{x}_t) &\geq -\sum_{t=1}^{T_0} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{x}_t)\rangle - \mathcal{E}^{\mathrm{D}}_{T_0} \\
&\geq -\sum_{t=1}^{T_0} \langle \boldsymbol{\lambda}_t, g_t(\boldsymbol{\xi}^\circ)\rangle - \mathcal{E}^{\mathrm{D}}_{T_0} - 2\mathcal{E}^{\mathrm{P}}_{T_0,\delta} \\
&\geq -\sum_{t=1}^{T_0} \langle \boldsymbol{\lambda}_t, \bar{g}(\boldsymbol{\xi}^\circ)\rangle - \mathcal{E}^{\mathrm{D}}_{T_0} - 2\mathcal{E}^{\mathrm{P}}_{T_0,\delta} - \mathcal{E}_{T_0,\delta} \\
&\geq T_0\rho - \mathcal{E}^{\mathrm{D}}_{T_0} - 2\mathcal{E}^{\mathrm{P}}_{T_0,\delta} - \mathcal{E}_{T_0,\delta}.
\end{aligned}$$

Hence,

$$\begin{aligned}
\hat{\rho} &= -\frac{1}{T_0}\left(\max_{i\in[m]} \sum_{t=1}^{T_0} g_{t,i}(\boldsymbol{x}_t) + \mathcal{E}_{T_0,\delta}\right) \\
&\geq \frac{1}{T_0}\left(T_0\rho - \mathcal{E}^{\mathrm{D}}_{T_0} - 2\mathcal{E}^{\mathrm{P}}_{T_0,\delta} - \mathcal{E}_{T_0,\delta} - \mathcal{E}_{T_0,\delta}\right)
\end{aligned}$$

[9] Notice that even if ρ does not satisfy the condition, $\hat{\rho}$ is a lower bound on ρ. This is sufficient to guarantee that the results in Theorem 9.3 and Theorem 9.4 hold.

$$\geq \rho/2 + \frac{1}{T_0}\left(T_0\rho/2 - \mathcal{E}^{\mathrm{D}}_{T_0} - 2\mathcal{E}^{\mathrm{P}}_{T_0,\delta} - 2\mathcal{E}_{T_0,\delta}\right)$$
$$\geq \rho/2,$$

where the last inequality comes from Condition 9.2. This concludes the proof. □

By applying the results of the previous sections on the guarantees of Algorithm 9.1, and by using primal and dual RMs whose regret bounds are of the order $\tilde{O}(\sqrt{T})$, we get $\tilde{O}(\sqrt{T}/\rho)$ and $\tilde{O}(\sqrt{T}/\rho)$ regret and violation bounds, respectively, when the functions g_t are selected stochastically.

9.7 Applications to repeated auctions settings

Internet advertising platforms usually operationalize large auction markets by using *proxy bidders* that place bids in repeated auctions on the advertisers' behalf. A proxy-bidder selects bids according to a *budget-pacing mechanism*, which manages the usage of the advertisers' budget over time Agarwal et al. (2014); Conitzer et al. (2021); Balseiro et al. (2021a). In this section, we discuss the application of our framework to budget-management in auctions, arguing that it can deal with more general constraints on ad slots allocation with respect to what is currently achievable with multiplicative pacing algorithms, which manage only *knapsack constraints*.

We consider the problem faced by a bidder who takes part in a sequence of repeated auctions. We focus on the case of *second-price* and *first-price* auctions, since they are the *de facto* standard in large Internet advertising platforms. At each round $t \in [T]$, the bidder observes their valuation v_t from a finite set of n_{v} possible valuations $\mathcal{V} \subset [0,1]$. Such valuation models targeting preferences of the advertiser. Then, the bidder chooses a bid $b_t \in \mathcal{B}$, where $\mathcal{B} \subset [0,1]$ is a finite set of n_{b} possible bids such that $0 \in \mathcal{B}$ (*i.e.*, the bidder is allowed to skip items without incurring in any cost). The utility of the bidder depends on the largest among competing bids, denoted by β_t. In particular, the utility is computed as $f_t(b_t) = (v_t - c_t(b_t))\mathbf{1}_{\{b_t \geq \beta_t\}}$, where the cost c_t is such that $c_t(b_t) = \mathbf{1}_{\{b_t \geq \beta_t\}}$ in second-price auctions, and $c_t(b_t) = b_t \mathbf{1}_{\{b_t \geq \beta_t\}}$ for first-price ones. Finally, the bidder has a target *per-round* budget of $\rho > 0$, which yields an overall budget $B \coloneqq \rho T$ that limits the total spending over the T rounds. In the case of budget-constrained bidding, a strictly feasible solution can be easily achieved by always bidding 0. Using the target per-round budget $\rho = B/T$ we can write the budget constraint as $\sum_{t\in[T]} g_t(b_t) \leq 0$, with $g_t(b) = c_t(b) - \rho$ for any $b \in \mathcal{B}$. Notice

that, in this setting, we have the same feasibility parameter ρ for both the stochastic and the adversarial case.

As a benchmark to evaluate the algorithm, we consider the best feasible static policy $\pi : \mathcal{V} \to \mathcal{B}$. The set of static policies can be represented by $\mathcal{X} \coloneqq \mathcal{B}^{n_{\mathrm{v}}}$, where a vector $\boldsymbol{b} \in \mathcal{B}^{n_{\mathrm{v}}}$ encodes the policy's bids for each possible valuation. To apply our framework to this problem, it is sufficient to design a primal regret minimizer constructor (recall that, in order to design dual RMs, we can employ OMD). This can be implemented by instantiating a regret minimizer EXP3.P (Auer et al., 2002b) for each possible valuation in $\mathcal{V}$. Given a failure probability $\nu \in (0, 1)$, each RM guarantees a regret bound $O(\sqrt{T n_{\mathrm{b}} \log(n_{\mathrm{b}}/\nu)})$ with probability at least $1 - \nu$. Thus, given a desired failure probability $\eta \in (0, 1)$, by setting $\nu = \eta/n_{\mathrm{v}}$ we get that, with probability at least $1 - \eta$, the bounds of all the RMs hold. Hence, by a union bound, we get that the regret of a primal RM is $\mathcal{E}^{\mathrm{P}}_{T,\eta} = O(n_{\mathrm{v}}\sqrt{T n_{\mathrm{b}} \log(n_{\mathrm{b}} n_{\mathrm{v}}/\eta)})$.

Guaranteed budget completion in the stochastic case. The crux of budget-pacing mechanisms is ensuring that the advertisers' budget is not depleted too early (thereby missing potentially valuable future advertising opportunities), while being fully depleted within the planned duration of the campaign. Theorem 9.4 shows that, when inputs are generated according to some stochastic model, Algorithm 9.1 never enters the recovery phase. This is crucial in the context of budget-pacing mechanisms, because whenever the algorithm enters the recovery phase it will converge to always bid 0 in order to mitigate constraints violation. Therefore, the bidder could miss out on potentially valuable items. Moreover, if the platform wanted to guarantee that the bidder does not spend more than the budget B, it would be enough to set a *virtual budget* of $B - \tilde{O}(T^{1/2})$ to compensate for the potential constraints violation. Finally, we argue that, in large-scale markets, an individual bidder has almost no impact on prices, and, thus, stochastic behavior of costs is a reasonable assumption.

Adversarial case. Theorem 6.1 of Castiglioni et al. (2022a) shows how to construct an algorithm that provides a ρ fraction of the optimal utility for problems with budget constraints and adversarial inputs. The ratio $\rho/(1+\rho)$ obtained in Theorem 9.5 matches such result. The latter assumes that rewards and costs are in $[0, 1]$, and, thus, $g_t \in [-\rho, 1-\rho]$ (as they only model budget constraints). However, in our case we have $g_t \in [-1, 1]$. By normalizing the former range to match with ours, we get $g_t \in [-\rho/(1-\rho), 1]$. Therefore, the feasibility parameter would be $\rho' = \rho/(1-\rho)$. By rewriting our guarantees as a function of ρ, we get $\rho'/(1+\rho') = \rho$, which is the same guarantee

of Castiglioni et al. (2022a).

Handling ROI constraints. Traditional budget-pacing mechanisms (see, *e.g.*, Balseiro and Gur (2019); Balseiro et al. (2020)) are based on primal-dual algorithms that are near optimal in settings with knapsack constraints only, and they cannot be generalized to deal with other types of long-term constraints. However, there are many real-world situations in which guaranteeing other types of constraints is crucial for practical applications (see, *e.g.*, Golrezaei et al. (2021b,a)).

One example is the case of *return on investment* (ROI) constraints Auerbach et al. (2008); Golrezaei et al. (2021b); Li et al. (2020). [10] The recent work by Golrezaei et al. (2021a) presents a threshold-based algorithms for repeated *second-price* auctions under budget and ROI constraints. Our framework allows advertisers to reach a target ROI while keeping budget expenditures under control also in the setting of repeated *first-price* auctions, which is a frequent setting in practice.[11] In particular, given a target ROI $\omega \geq 0$ and the largest among competing bids β_t, we define the ROI constraints as

$$g_t(b_t) = \left(\omega - \frac{v_t}{b_t}\right)\mathbf{1}_{\{b_t \geq \beta_t\}} le 0.$$

Then, it is enough to instantiate the framework with the same setup of Section 9.7, that is, EXP3.P (Auer et al., 2002b) for each of the possible valuations in $\mathcal{V}$, and OMD equipped with negative-entropy regularizer for the dual RM. Therefore, we immediately get that the cumulative violations of the budget and ROI constraints are upper bounded by $\tilde{O}(T^{1/2})$. This holds both in the fully stochastic and in the fully adversarial setting under the assumption of having a strictly feasible solution, which is reasonable since it is enough to have a sufficiently *small* bid in the set of available bids $\mathcal{B}$. We observe that always bidding such a *small* bid is sufficient to satisfy the ROI constraints but will penalize the cumulative rewards obtained by the advertiser.

Future research direction: fairness constraints. Consider the setting in which each item appearing at time t is characterized by one or more of n_c categories according to the vector $\boldsymbol{e}_t \in [0,1]^{n_c}$. A bidder may have distributional preferences over such categories, such as ensuring that at least a certain

[10] This is a frequent advertising objective in large Internet advertising platform. See, *e.g.*, `https://tinyurl.com/c86rezhd` and `https://tinyurl.com/mr49vz8a`.

[11] For example, in 2019 Google announced a shift to first-price auctions for its AdManager exchange. See `https://tinyurl.com/chv5nxys`.

fraction of impressions is allocated to each category. This is the case, for example, of advertisers who need to perform online outreach to a population of users while achieving a distribution over different demographics *close* to that of the real underlying population. For example, Gelauff et al. (2020) provide an interesting field study about running advertising campaigns for Participatory Budgeting elections. In Participatory Budgeting elections, community members are asked to vote between various public projects in order to allocate a total budget. The election organizer may use online advertising to try to promote the initiative, and in doing so the goal is to reach a "demographic mix" comparable to that of the local population. Surprisingly, Gelauff et al. (2020) show that advertisers currently have to resort to complex segmentation strategies through subcampaigns in order to achieve that goal.

Two recent works propose to achieve such distributional preferences within budget-pacing mechanisms by embedding them into a concave regularization term in the advertiser's objective Balseiro et al. (2021b); Celli et al. (2022). Such frameworks specifically consider the case of repeated second-price auctions, and can directly handle only packing constraints. Encoding distributional preferences via a regularization term in the objective implies that they cannot provide any formal guarantee w.r.t. how *close* the realized distribution of impressions is to the target, despite showing promising performance in practice.

Differently from previous work, our framework can *explicitly* handle distributional constraints within second- and first-price auction frameworks. Let vector $\hat{\boldsymbol{e}} \in [0,1]^{n_{\mathrm{c}}}$ be such that $\hat{\boldsymbol{e}}_j$ is the fraction of impressions that we want to be allocated to users of category j. Then, for each category $j \in [n_{\mathrm{c}}]$, we could enforce the following type of constraints

$$g_{t,j}(b_t) \coloneqq \hat{\boldsymbol{e}}_j - \boldsymbol{e}_{t,j}\mathbf{1}_{\{b_t \geq \beta_t\}} \leq 0.$$

Assuming the existence of a strictly feasible bidding strategy, our framework guarantees that, for each category j,

$$\hat{\boldsymbol{e}}_j - \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{e}_{t,j}\mathbf{1}_{\{b_t \geq \beta_t\}} \leq \tilde{O}(T^{-1/2}),$$

$$\hat{\boldsymbol{e}}_j - \frac{1}{T}\sum_{t=1}^{T} \boldsymbol{e}_{t,j}\mathbf{1}_{\{b_t \geq \beta_t\}} \leq \tilde{O}(T^{-1/2}),$$

which guarantees that, in the limit, the difference between the average distribution of impressions and the target thresholds is vanishing.

The main question which still needs to be answered in order to apply our framework in the case of fairness constraints is whether we can motivate the existence of a strictly feasible solution. One reasonable requirement is to constrain the target vector $\hat{e}$ to be a point in the full-dimensional simplex with dimension n_c. On top of that, the advertiser would need a way to "buy what's necessary" in order to match the distributional constraints. This desideratum could be achieved, for example, by introducing *buyout options* for advertisers, in the spirit of Gallien and Gupta (2007) (*i.e.*, when the advertiser needs impression from a certain category, they always have the option of bidding the buyout value to be sure to win the relevant items). Therefore, assuming the population of users is large enough, an advertiser could achieve a strictly feasible solution by bidding according to the fixed strategy mixture recommending to bid the buyout option for each category j with a probability greater than or equal to $\hat{e}_j$.

The model we described is clearly a simplification of real budget-pacing systems. Moreover, the practical implications of introducing buyout options should be further investigated, in order to understand if they constitute a viable solution both for the platform and advertisers. Finally, we leave as interesting future research directions the problem of studying the general setting (with arbitrary sets $\mathcal{V}$ and $\mathcal{B}$), and that of providing an empirical evaluation of the above techniques on real-world data.

CHAPTER 10

Conclusions

Research on digital markets has considerably expanded in recent years, and these progresses have increased the impact of Web platforms on the performance of sellers and advertisers. New AI tools have been introduced in e-commerce platforms to boost sales while providing meaningful guarantees to sellers. The existing literature on mechanism design and online learning provides efficient solutions in theory and practice for standard pricing and advertising settings. However, such techniques cannot be applied in a direct way to novel and complex scenarios such as the ones we consider in this thesis. We show that it is often possible to exploit the structure of these new scenarios to find efficient solutions and recover good approximations of the results obtained in standard settings.

First, we provide efficient mechanisms and algorithms for the general problem of selling items. In this part of the thesis, the main challenges we address are finding pricing strategies for perishable items and dealing with possible delays of the seller's reward. As a future research direction, it would be interesting to relax some of the assumptions made in our analysis, in order to study more general scenarios which could be closer to real-world settings. For instance, in Chapter 3 we provide distribution-free posted-price mechanisms in order to sell a unique item within a finite time period. We

evaluate our mechanisms in terms of competitive ratio, measuring the worst-case ratio between their revenue and that of an optimal mechanism that knows the distribution of valuations. In particular, we prove that both mechanisms achieve a competitive ratio that is constant with respect to the actual valuation when the distribution of the valuations has a monotone hazard rate. This shows that our mechanisms are robust even in non-stationary markets subject to arbitrary distribution changes preserving the same support. In future, we will investigate hybrid settings in which our robust mechanisms can be combined with machine learning tools. For instance, data could be used to learn a class of distributions, and we could design a mechanism robust with respect to all the distributions of that class. Another interesting direction could be considering the general problem of selling multiple units of multiple items in a sequential way when valuations may be discounted. Chapter 4 introduces the novel TP-MAB setting, which generalizes the delayed-feedback bandit setting with bounded delay. It could be worth studying how to deal with general and possibly unbounded delays in the time in which seller's rewards appear. The TP-MAB setting with unbounded delay could model recommendation problem in online advertising setting. Suppose that an advertiser at each time instant has to choose which ad to display on a Web page. Each ad impression could produce a number of conversions, which correspond to a reward distributed over time. However, some conversions could be indefinitely delayed over time. Another possible research direction, could be extending the analysis of how arms' cumulative rewards are distributed across multiple rounds. For instance, in some scenarios, there might be additional information available about how the cumulative reward is partitioned over the rounds. It may be reasonable to assume that a significant portion of the cumulative reward is observed during the initial rounds, and that subsequent rounds exhibit a diminishing exponential decline in observed partial reward. In these more general scenarios, it is worth investigating how the α-smoothness property could be generalized and then exploited to develop no-regret algorithms.

In the second part of the thesis, we provide new mechanisms for cutting-edge advertising scenarios originated by recent innovations in advertising platforms. We design new types of ad auctions, and measure their performance in new scenarios such as the one in which an additional price parameter is displayed in the ads, and the metaverse. During the analysis of the first scenario, the externalities introduce by the display of prices with ads are modeled through a metric called *quality*. We design the quality of an ad as a function depending on the price displayed with that ad and the minimum price among all displayed ads. An interesting research direction concerns

the analysis of PoA and PoS and the design of allocation algorithms when the quality functions satisfy specific properties, such as, e.g., smoothness. An alternative approach to design the quality function is to consider the scenario where the quality functions depend on the prices associated with all the displayed ads. Subsequently, in the second part of the chapter, we provide bidding strategies for a group of colluding advertisers coordinated by a common media agency, who participate in the same ad auction. Colluders aim at maximizing their cumulative utility by coordinating their bids while playing against some external bidders. In the future, it would be interesting to further increase the complexity of this scenario by considering adversarial external agents. In this scenario, the colluders would face the problem of learning a bidding strategy which would have to be robust with respect to an adversarial environment. Then, it would be interesting to evaluate both theoretically and empirically the outcomes to which this system converges from a global perspective. A further extension could be considering coalitions of different size, and the presence of more than one coalition.

Finally, the third part of this thesis studies the problem faced by a constrained agent that has to learn effective bidding strategies. In Chapter 9, we provide a general framework and a best-of both-world algorithm to address this problem. In the future, it would be interesting to apply our framework to real-world problems in which constraints represent, for instance, fairness requirements that the platform needs to implement. It would be interesting to study if, in such specific settings, better guarantees can be provided with respect to those of the general framework. Moreover, it would be valuable to explore an intermediate framework between the stochastic and adversarial settings, and see which type of guarantees could be achieved. Such setting could be modeled, for instance, by fixing an underlying distribution and assuming that, at each round, the adversary chooses the constraints distributions which are not too distant from the underlying one. This could be expressed,for instance, by setting a bound on the total variation between the two distributions. An alternative approach for implementing this intermediate setting might involve using a smoothed adversary who chooses constraints distributions that are not excessively spiky. Furthermore, a broader area of research that could be explored involves investigating the interaction among multiple constrained agents. To tackle this issue, the first step would be establishing an appropriate definition of equilibrium, followed by analyzing the convergence towards it. Finally, it is worth mentioning some recent works from the literature related to the problem presented in Chapter 9. Castiglioni et al. (2023) investigate further the problem in the scenario where the feasibility parameter ρ is unknown beforehand and relax

the assumption of having one strictly feasible solution for each round in the adversarial setting. Fikioris and Tardos (2023) study a scenario that is not exactly stochastic but is also not worst-case. They call the problem Approximately Stationary Bandit with Knapsack and introduce a condition that parameterizes how close to stochastic or adversarial an instance is. The direction of their work is towards bridging the gap between the no-regret guarantees achievable in the stochastic setting and the competitive ratio guarantees attainable in the adversarial setting. While their work only focuses on resource constraints, it would be intriguing to explore the extension of these ideas to general constraints.

Bibliography

M. Abeille, C. Calauzènes, N. E. Karoui, T. Nedelec, and V. Perchet. Explicit shading strategies for repeated truthful auctions. *arXiv preprint arXiv:1805.00256*, 2018.

G. M. Accabi, F. Trovò, A. Nuara, N. Gatti, and M. Restelli. When gaussian processes meet combinatorial bandits: Gcb. In *EWRL*, pages 1–11, 2018.

M. Adamczyk, A. Borodin, D. Ferraioli, B. D. Keijzer, and S. Leonardi. Sequential posted-price mechanisms with correlated valuations. *ACM Transactions on Economics and Computation (TEAC)*, 5(4):1–39, 2017.

D. Agarwal, S. Ghosh, K. Wei, and S. You. Budget pacing for targeted online advertisements at linkedin. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1613–1619, 2014.

G. Aggarwal, J. Feldman, S. Muthukrishnan, and M. Pál. Sponsored search auctions with markovian users. In *WINE*, pages 621–628, 2008.

S. Agrawal and N. R. Devanur. Bandits with concave rewards and convex knapsacks. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 989–1006. ACM, 2014.

S. Agrawal and N. R. Devanur. Bandits with global convex constraints and objective. *Operations Research*, 67(5):1486–1502, 2019.

S. Amani, M. Alizadeh, and C. Thrampoulidis. Regret bound for safe gaussian process bandit optimization. In *L4DC*, pages 158–159, 2020.

D. Amodei, C. Olah, J. Steinhardt, P. Christiano, J. Schulman, and D. Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

S. Arya and Y. Yang. Randomized allocation with nonparametric estimation for contextual multi-armed bandits with delayed rewards. *Statistics & Probability Letters*, 164:108818, 2020.

I. Ashlagi, D. Monderer, and M. Tennenholtz. Mediators in position auctions. *Games and Economic Behavior*, 67:2–21, 2009.

P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002a.

P. Auer, N. Cesa-Bianchi, Y. Freund, and R. E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM journal on computing*, 32(1):48–77, 2002b.

J. Auerbach, J. Galenson, and M. Sundararajan. An empirical analysis of return on investment maximization in sponsored search auctions. In *Proceedings of the 2nd International Workshop on Data Mining and Audience Intelligence for Advertising*, pages 1–9, 2008.

M. Babaioff, S. Dughmi, R. Kleinberg, and A. Slivkins. Dynamic pricing with limited supply. *ACM Transactions on Economics and Computation (TEAC)*, 3(1):1–26, 2015.

M. Babaioff, L. Blumrosen, S. Dughmi, and Y. Singer. Posting prices with unknown distributions. *ACM Transactions on Economics and Computation (TEAC)*, 5(2):1–20, 2017.

Y. Bachrach. Honor among thieves: collusion in multi-unit auctions. In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems*, pages 617–624, 2010.

A. Badanidiyuru, R. Kleinberg, and A. Slivkins. Bandits with knapsacks. *Journal of the ACM (JACM)*, 65(3):1–55, 2018.

S. Balseiro, H. Lu, and V. Mirrokni. The best of many worlds: Dual mirror descent for online allocation problems. *arXiv preprint arXiv:2011.10124*, 2020.

S. Balseiro, A. Kim, M. Mahdian, and V. Mirrokni. Budget-management strategies in repeated auctions. *Operations Research*, 2021a.

S. Balseiro, H. Lu, and V. Mirrokni. Regularized online allocation problems: Fairness and beyond. In *International Conference on Machine Learning*, pages 630–639. PMLR, 2021b.

S. R. Balseiro and Y. Gur. Learning in repeated auctions with budgets: Regret minimization and equilibrium. *Management Science*, 65(9):3952–3968, 2019.

R. E. Barlow and A. W. Marshall. Bounds for distributions with monotone hazard rate, ii. *Annals of Mathematical Statistics*, 35(3):1258–1274, 09 1964.

N. Basilico, A. Celli, G. De Nittis, and N. Gatti. Team-maxmin equilibrium: efficiency bounds and algorithms. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

A. Beck and M. Teboulle. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3):167–175, 2003.

A. Bernstein, S. Mannor, and N. Shimkin. Online classification with specificity constraints. *Advances in Neural Information Processing Systems*, 23, 2010.

I. Bistritz, Z. Zhou, X. Chen, N. Bambos, and J. Blanchet. Exp3 learning in adversarial bandits with delayed feedback. *NeurIPS*, 2019.

C. Borgs, J. Chayes, N. Immorlica, K. Jain, O. Etesami, and M. Mahdian. Dynamics of bid optimization in online advertisement auctions. In *WWW*, pages 531–540, 2007.

A. Borodin and R. El-Yaniv. *Online computation and competitive analysis*. Cambridge University Press, 2005.

B. Brost, R. Mehrotra, and T. Jehan. The music streaming sessions dataset. In *WWW*. ACM, 2019.

S. Bubeck and N. Cesa-Bianchi. Regret analysis of stochastic and nonstochastic multi-armed bandit problems. *Foundations and Trends in Machine Learning*, 5(1):1–122, 2012.

S. Bubeck and A. Slivkins. The best of both worlds: Stochastic and adversarial bandits. In *Conference on Learning Theory*, pages 42–1. JMLR Workshop and Conference Proceedings, 2012.

S. Bubeck, Y. T. Lee, and R. Eldan. Kernel-based methods for bandit convex optimization. In *Proceedings of the 49th Annual ACM SIGACT Symposium on Theory of Computing*, page 72–85, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450345286.

G. Calinescu, C. Chekuri, M. Pál, and J. Vondrák. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011.

X. Cao and K. R. Liu. Online convex optimization with time-varying constraints and bandit feedback. *IEEE Transactions on automatic control*, 64(7):2665–2680, 2018.

I. Caragiannis, C. Kaklamanis, P. Kanellopoulos, and M. Kyropoulou. On the efficiency of equilibria in Generalized Second Price auctions. In *EC*, 2011.

I. Caragiannis, C. Kaklamanis, P. Kanellopoulos, M. Kyropoulou, B. Lucier, R. Paes Leme, and E. Tardos. Bounding the inefficiency of outcomes in generalized second price auctions. *Journal of Economic Theory*, 156:343–388, 2015.

M. Cary, A. Das, B. Edelman, I. Giotis, K. Heimerl, A. R. Karlin, C. Mathieu, and M. Schwarz. Greedy bidding strategies for keyword auctions. In *Proceedings of the 8th ACM Conference on Electronic Commerce*, pages 262–271, 2007.

M. Castiglioni, A. Celli, and C. Kroer. Online learning with knapsacks: the best of both worlds. In *International Conference on Machine Learning, ICML 2022*, pages 2767–2783, 2022a.

M. Castiglioni, A. Celli, A. Marchesi, G. Romano, and N. Gatti. A unifying framework for online optimization with long-term constraints, 2022b.

M. Castiglioni, D. Ferraioli, N. Gatti, A. Marchesi, and G. Romano. Efficiency of ad auctions with price displaying. 2022c.

M. Castiglioni, A. Nuara, G. Romano, G. Spadaro, F. Trovò, and N. Gatti. Safe online bid optimization with return-on-investment and budget constraints subject to uncertainty. *arXiv preprint arXiv:2201.07139*, 2022d.

M. Castiglioni, A. Celli, and C. Kroer. Online bidding in repeated non-truthful auctions under budget and roi constraints. *arXiv preprint arXiv:2302.01203*, 2023.

S. Cayci, A. Eryilmaz, and R. Srikant. Learning to control renewal processes with bandit feedback. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(2):1–32, 2019.

L. Cella and N. Cesa-Bianchi. Stochastic bandits with delay-dependent payoffs. In *AISTATS*, pages 1168–1177, 2020.

A. Celli and N. Gatti. Computational results for extensive-form adversarial team games. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

A. Celli, R. Colini-Baldeschi, C. Kroer, and E. Sodomka. The parity ray regularizer for pacing in auction markets. In *Proceedings of the ACM Web Conference 2022*, pages 162–172, 2022.

N. Cesa-Bianchi and G. Lugosi. *Prediction, learning, and games*. Cambridge University Press, 2006.

N. Cesa-Bianchi, C. Gentile, and Y. Mansour. Nonstochastic bandits with composite anonymous feedback. In *COLT*, pages 750–773, 2018.

S. Chawla, J. D. Hartline, D. L. Malec, and B. Sivan. Multi-parameter mechanism design and sequential posted pricing. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pages 311–320, 2010.

T. Chen and G. B. Giannakis. Bandit convex optimization for scalable and dynamic iot management. *IEEE Internet of Things Journal*, 6(1):1276–1286, 2018.

T. Chen, Q. Ling, and G. B. Giannakis. An online convex optimization approach to proactive network resource allocation. *IEEE Transactions on Signal Processing*, 65(24):6350–6364, 2017.

W. Chen, Y. Wang, and Y. Yuan. Combinatorial multi-armed bandit: General framework and applications. In *ICML*, pages 151–159, 2013.

Y. Chen and V. F. Farias. Robust dynamic pricing with strategic customers. *Mathematics of Operations Research*, 43(4):1119–1142, 2018.

S. Chowdhury and A. Gopalan. On kernelized multi-armed bandits. In *ICML*, pages 844–853, 2017.

M. Chui, J. Manyika, M. Miremadi, N. Henke, R. Chung, P. Nel, and S. Malhotra. Notes from the AI frontier insights from hundreds of use cases. *McKinsey Global Institute*, 2018.

V. Conitzer, C. Kroer, E. Sodomka, and N. E. Stier-Moses. Multiplicative pacing equilibria in auction markets. *Operations Research*, 2021.

J. Correa, P. Foncea, R. Hoeksma, T. Oosterwijk, and T. Vredeveld. Posted price mechanisms for a random stream of customers. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, pages 169–186, 2017.

L. Croissant, M. Abeille, and C. Calauzènes. Real-time optimisation for online learning in auctions. In *International Conference on Machine Learning*, pages 2217–2226. PMLR, 2020.

F. Decarolis and G. Rovigatti. Online Auctions and Digital Marketing Agencies. Working Papers 17-08, NET Institute, Sept. 2017. URL `https://ideas.repec.org/p/net/wpaper/1708.html`.

F. Decarolis, M. Goldmanis, and A. Penta. Marketing Agencies and Collusive Bidding in Online Ad Auctions. *Management Science*, 66(10):4433–4454, 2020.

C. Deng and S. Pekec. Money for nothing: exploiting negative externalities. In *ACM EC*, pages 361–370, 2011.

T. Desautels, A. Krause, and J. W. Burdick. Parallelizing exploration-exploitation tradeoffs in gaussian process bandit optimization. *Journal of Machine Learning Research*, 15(119):4053–4103, 2014.

N. R. Devanur and S. M. Kakade. The price of truthfulness for pay-per-click auctions. In *ACM EC*, pages 99–106, 2009.

E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1): 269–271, 1959.

W. Ding, T. Qin, X.-D. Zhang, and T. Liu. Multi-armed bandit with budget constraint and variable costs. In *AAAI*, pages 232–238, 2013.

M. Dudik, D. Hsu, S. Kale, N. Karampatziakis, J. Langford, L. Reyzin, and T. Zhang. Efficient optimal learning for contextual bandits. *arXiv preprint arXiv:1106.2369*, 2011.

L. Einav, C. Farronato, J. Levin, and N. Sundaresan. Auctions versus posted prices in online markets. *Journal of Political Economy*, 126(1):178–215, 2018.

G. Farina and N. Gatti. Ad auctions and cascade model: GSP inefficiency and algorithms. In *AAAI*, pages 489–495, 2016.

G. Farina and N. Gatti. Adopting the cascade model in ad auctions: Efficiency bounds and truthful algorithmic mechanisms. *J. Artif. Intell. Res.*, 59:265–310, 2017.

G. Farina, A. Celli, N. Gatti, and T. Sandholm. Ex ante coordination and collusion in zero-sum multi-player extensive-form games. *Advances in Neural Information Processing Systems*, 31, 2018.

U. Feige. A threshold of ln n for approximating set cover. *Journal of the ACM (JACM)*, 45(4): 634–652, 1998.

J. Feldman, S. Muthukrishnan, M. Pal, and C. Stein. Budget optimization in search-based advertising auctions. In *ACM EC*, pages 40–49, 2007.

G. Fikioris and É. Tardos. Approximately stationary bandits with knapsacks. *arXiv preprint arXiv:2302.14686*, 2023.

D. Fotakis, P. Krysta, and O. Telelis. Externalities among advertisers in sponsored search. In *SAGT*, pages 105–116, 2011.

N. Galichet, M. Sebag, and O. Teytaud. Exploration vs exploitation vs safety: Risk-aware multi-armed bandits. In *ACML*, pages 245–260, 2013.

J. Gallien and S. Gupta. Temporary and permanent buyout prices in online auctions. *Management Science*, 53(5):814–833, 2007.

J. Garcia and F. Fernández. Safe exploration of state and action spaces in reinforcement learning. *J ARTIF INTELL RES*, 45:515–564, 2012.

N. Gatti, F. Di Giunta, and S. Marino. Alternating-offers bargaining with one-sided uncertain deadlines: an efficient algorithm. *Artificial Intelligence*, 172(8-9):1119–1157, 2008.

N. Gatti, M. Rocco, S. Ceppi, and E. Gerding. Mechanism design for mobile geo-location advertising. In *AAAI*, pages 691–697, 2014.

N. Gatti, M. Rocco, P. Serafino, and C. Ventre. Towards better models of externalities in sponsored search auctions. *Theoretical Computer Science*, 745:150–162, 2018.

L. Gelauff, A. Goel, K. Munagala, and S. Yandamuri. Advertising for demographically fair outcomes. *arXiv preprint arXiv:2006.03983*, 2020.

I. Giotis and A. Karlin. On the equilibria and efficiency of the gsp mechanismin keyword auctions with externalities. In *WINE*, pages 629–638, 2008.

N. Golrezaei, H. Nazerzadeh, and R. Randhawa. Dynamic pricing for heterogeneous time-sensitive customers. *Manufacturing & Service Operations Management*, 22(3):562–581, 2020.

N. Golrezaei, P. Jaillet, J. C. N. Liang, and V. Mirrokni. Bidding and pricing in budget and roi constrained markets. *arXiv preprint arXiv:2107.07725*, 2021a.

N. Golrezaei, I. Lobel, and R. Paes Leme. Auction design for roi-constrained buyers. In *WWW*, pages 3941–3952, 2021b.

M. Grötschel, L. Lovász, and A. Schrijver. The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica*, 1(2):169–197, 1981.

A. Hans, D. Schneegaß, A. M. Schäfer, and S. Udluft. Safe exploration for reinforcement learning. In *ESANN*, pages 143–148, 2008.

M. Harris and R. M. Townsend. Resource allocation under asymmetric information. *Econometrica: Journal of the Econometric Society*, pages 33–64, 1981.

J. Hartline, D. Hoy, and S. Taggart. Price of anarchy for auction revenue. In *EC*, pages 693–710, 2014.

J. Håstad. Clique is hard to approximate within $n^{1-\epsilon}$. *Acta Mathematica*, 182(1):105–142, 1999.

E. Hazan. *Efficient algorithms for online convex optimization and their applications*. Princeton University, 2006.

E. Hazan. Introduction to online convex optimization, 2019.

E. Hazan, A. Agarwal, and S. Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2):169–192, 2007.

IAB. Interactive Advertising Bureau (IAB) internet advertising revenue report, Full year 2020 results, 2021.

N. Immorlica, K. A. Sankararaman, R. Schapire, and A. Slivkins. Adversarial bandits with knapsacks. In *60th IEEE Annual Symposium on Foundations of Computer Science, FOCS 2019*, pages 202–219. IEEE Computer Society, 2019.

R. Jenatton, J. Huang, and C. Archambeau. Adaptive algorithms for online convex optimization with long-term constraints. In *International Conference on Machine Learning*, pages 402–411. PMLR, 2016.

P. Joulani, A. Gyorgy, and C. Szepesvári. Online learning under delayed feedback. In *ICML*, pages 1453–1461, 2013.

K. Jung, Y. Cho, and S. Lee. Online shoppers' response to price comparison sites. *Journal of Business Research*, 67(10):2079–2087, 2014.

D. Kempe and M. Mahdian. A cascade model for externalities in sponsored search. In *International Workshop on Internet and Network Economics*, pages 585–596. Springer, 2008.

R. Kleinberg and T. Leighton. The value of knowing a demand curve: Bounds on regret for online posted-price auctions. In *44th Annual IEEE Symposium on Foundations of Computer Science, 2003. Proceedings.*, pages 594–605. IEEE, 2003.

D. Kong, X. Fan, K. Shmakov, and J. Yang. A combinational optimization approach for advertising budget allocation. In *WWW*, pages 53–54, 2018.

R. Lavi and N. Nisan. Competitive analysis of incentive compatible on-line auctions. *Theoretical Computer Science*, 310(1-3):159–180, 2004.

B. Li, X. Yang, D. Sun, Z. Ji, Z. Jiang, C. Han, and D. Hao. Incentive mechanism design for roi-constrained auto-bidding. *arXiv preprint arXiv:2012.02652*, 2020.

N. Liakopoulos, A. Destounis, G. Paschos, T. Spyropoulos, and P. Mertikopoulos. Cautious regret minimization: Online optimization with long-term budget constraints. In *International Conference on Machine Learning*, pages 3944–3952. PMLR, 2019.

I. Lobel. Dynamic pricing with heterogeneous patience levels. *Operations Research*, 68(4):1038–1046, 2020.

E. Lorenzon. Collusion With a Rent-Seeking Agency in Sponsored Search Auctions. Technical report, SSRN, 2018.

B. Lucier and R. P. Leme. GSP auctions with correlated types. In *EC*, 2011.

D. G. Luenberger. *Optimization by vector space methods*. John Wiley & Sons, 1997.

M. Mahdavi, R. Jin, and T. Yang. Trading regret for efficiency: online convex optimization with long term constraints. *The Journal of Machine Learning Research*, 13(1):2503–2528, 2012.

T. Mandel, Y.-E. Liu, E. Brunskill, and Z. Popović. The queue method: Handling delay, heuristics, prior data, and evaluation in bandits. In *AAAI*, volume 29, 2015.

A. G. Manegueu, C. Vernade, A. Carpentier, and M. Valko. Stochastic bandits with arm-dependent delays. In *ICML*, pages 3348–3356, 2020.

S. Mannor, J. N. Tsitsiklis, and J. Y. Yu. Online learning with sample path constraints. *J MACH LEARN RES*, 10:569–590, 2009.

W. Mao, Z. Zheng, F. Wu, and G. Chen. Online pricing for revenue maximization with unknown time discounting valuations. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pages 440–446, 2018.

A. Mas-Colell, M. Whinston, and J. Green. *Microeconomic Theory*. Oxford University Press, 1995.

R. Mason and J. Välimäki. Learning about the arrival of sales. *Journal of Economic Theory*, 146(4): 1699–1711, 2011.

D. S. Mitrinovic, J. Pecaric, and A. M. Fink. *Classical and new inequalities in analysis*, volume 61. Springer Science & Business Media, 2013.

M. Mohri and A. M. Medina. Learning theory and algorithms for revenue optimization in second price auctions with reserve. In *International conference on machine learning*, pages 262–270. PMLR, 2014.

M. Mohri and A. Munoz. Optimal regret minimization in posted-price auctions with strategic buyers. In *Advances in Neural Information Processing Systems*, pages 1871–1879, 2014.

A. Moradipari, C. Thrampoulidis, and M. Alizadeh. Stage-wise conservative linear bandits. In *NeurIPS*, pages 11191–11201, 2020.

R. B. Myerson. Optimal auction design. *Mathematics of operations research*, 6(1):58–73, 1981.

J. Nash. Equilibrium points in n-person games. *Proceedings of the national academy of sciences*, 36 (1):48–49, 1950.

J. Nash. Non-cooperative games. *Annals of mathematics*, pages 286–295, 1951.

T. Nedelec, C. Calauzènes, N. El Karoui, V. Perchet, et al. Learning in repeated auctions. *Foundations and Trends® in Machine Learning*, 15(3):176–334, 2022.

G. L. Nemhauser, L. A. Wolsey, and M. L. Fisher. An analysis of approximations for maximizing submodular set functions-i. *Mathematical programming*, 14(1):265–294, 1978.

A. S. Nemirovskij and D. B. Yudin. Problem complexity and method efficiency in optimization. 1983.

G. Neu, A. György, C. Szepesvari, and A. Antos. Online markov decision processes under bandit feedback. *IEEE Transactions on Automatic Control*, 59(3):676–691, 2013.

N. Nisan and A. Ronen. Algorithmic mechanism design. *Games and Economic behavior*, 35(1-2): 166–196, 2001.

N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic Game Theory*. Cambridge University Press, New York, NY, USA, 2007.

A. Nuara, F. Trovò, N. Gatti, and M. Restelli. A combinatorial-bandit algorithm for the online joint bid/budget optimization of pay-per-click advertising campaigns. In *AAAI*, volume 32, 2018.

A. Nuara, N. Sosio, F. Trovò, M. Zaccardi, N. Gatti, and M. Restelli. Dealing with interdependencies and uncertainty in multi-channel advertising campaigns optimization. In *WWW*, pages 1376–1386, 2019.

A. Nuara, F. Trovò, N. Gatti, and M. Restelli. Online joint bid/daily budget optimization of internet advertising campaigns. *Artificial Intelligence*, page 103663, 2022.

OECD. *Algorithms and Collusion: Competition policy in the digital era*. 2017.

R. Paes Leme and E. Tardos. Pure and bayes-nash price of anarchy for generalized second price auction. In *FOCS*, pages 735–744, 2010.

D. C. Parkes. *Online mechanisms*. Cambridge University Press, 2007.

B. Peleg and P. Sudhölter. Introduction to the theory of cooperative games. *American Economic Review*, 2007.

C. Pike-Burke, S. Agrawal, C. Szepesvari, and S. Grunewalder. Bandits with delayed, aggregated anonymous feedback. In *ICML*, pages 4105–4113, 2018.

M. Pinsky and S. Karlin. *An introduction to stochastic modeling*. Academic press, 2010.

M. Pirotta, M. Restelli, A. Pecorino, and D. Calandriello. Safe policy iteration. In *ICML*, pages 307–315, 2013.

C. E. Rasmussen and C. K. Williams. *Gaussian processes for machine learning*, volume 1. MIT Press, 2006.

J. G. Riley and W. F. Samuelson. Optimal auctions. *The American Economic Review*, 71(3):381–392, 1981.

G. Romano, G. Tartaglia, A. Marchesi, and N. Gatti. Online posted pricing with unknown time-discounted valuations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5682–5689, 2021.

G. Romano, A. Agostini, F. Trovò, N. Gatti, and M. Restelli. Multi-armed bandit problem with temporally-partitioned rewards: When partial feedback counts. *Proceedings of IJCAI 2022*, 2022a.

G. Romano, M. Castiglioni, A. Marchesi, and N. Gatti. The power of media agencies in ad auctions: Improving utility through coordinated bidding. *arXiv preprint arXiv:2204.13772*, 2022b.

J. Rong, T. Qin, and B. An. Dynamic pricing for reusable resources in competitive market with stochastic demand. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 4718–4726, 2018.

E. C. Rosenthal. A pricing model for residential homes with Poisson arrivals and a sales deadline. *The Journal of Real Estate Finance and Economics*, 42(2):143–161, 2011.

S. M. Ross, J. J. Kelly, R. J. Sullivan, W. J. Perry, D. Mercer, R. M. Davis, T. D. Washburn, E. V. Sager, J. B. Boyce, and V. L. Bristow. *Stochastic processes*, volume 2. Wiley New York, 1996.

T. Roughgarden, V. Syrgkanis, and E. Tardos. The price of anarchy in auctions. *J. Artif. Intell. Res.*, 59:59–101, 2017.

A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50(1):97–109, 1982.

A. Schrijver. *Combinatorial Optimization: Polyhedra and Efficiency*, volume B. 01 2003.

S. Seifert. *Posted price offers in internet auction markets*, volume 580. Springer Science & Business Media, 2006.

V. Shah, R. Johari, and J. Blanchet. Semi-parametric dynamic contextual pricing. In *Advances in Neural Information Processing Systems*, pages 2363–2373. 2019.

S. Shalev-Shwartz et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.

N. Srinivas, A. Krause, M. Seeger, and S. M. Kakade. Gaussian process optimization in the bandit setting: No regret and experimental design. In *ICML*, pages 1015–1022, 2010.

A. S. Suggala and P. Netrapalli. Online non-convex learning: Following the perturbed leader is optimal. In A. Kontorovich and G. Neu, editors, *Proceedings of the 31st International Conference on Algorithmic Learning Theory*, volume 117 of *Proceedings of Machine Learning Research*, pages 845–861. PMLR, 08 Feb–11 Feb 2020.

Y. Sui, A. Gotovos, J. Burdick, and A. Krause. Safe exploration for optimization with gaussian processes. In *ICML*, pages 997–1005, 2015.

W. Sun, D. Dey, and A. Kapoor. Safety-aware algorithms for adversarial contextual bandit. In *International Conference on Machine Learning*, pages 3280–3288. PMLR, 2017.

B. K. Szymanski and J. Lee. Impact of roi on bidding and revenue in sponsored search advertisement auctions. In *Workshop on Sponsored Search Auctions*, volume 1, pages 1–8, 2006.

C. Taylor. Research on advertising in the metaverse: a call to action. *International Journal of Advertising*, 41:383–384, 2022.

S. Thomaidou, K. Liakopoulos, and M. Vazirgiannis. Toward an integrated framework for automated development and optimization of online advertising campaigns. *INTELL DATA ANAL*, 18(6): 1199–1227, 2014.

W. 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.

T. S. Thune, N. Cesa-Bianchi, and Y. Seldin. Nonstochastic multiarmed bandits with unrestricted delays. In *NeurIPS*, 2019.

F. Trovò, S. Paladino, M. Restelli, and N. Gatti. Budgeted multi-armed bandit in continuous action space. In *ECAI*, pages 560–568, 2016.

F. Trovò, S. Paladino, M. Restelli, and N. Gatti. Improving multi-armed bandit algorithms in online pricing settings. *International Journal of Approximate Reasoning*, 98:196–235, 2018.

D. van der Hoeven and N. Cesa-Bianchi. Nonstochastic bandits and experts with arm-dependent delays. *arXiv preprint arXiv:2111.01589*, 2021.

C. Vernade, O. Cappé, and V. Perchet. Stochastic bandit models for delayed conversions. In *UAI*, 2017.

C. Vernade, A. Carpentier, T. Lattimore, G. Zappella, B. Ermis, and M. Brueckner. Linear bandits with stochastic delayed feedback. In *ICML*, pages 9712–9721, 2020a.

C. Vernade, A. Gyorgy, and T. Mann. Non-stationary delayed bandits with intermediate observations. In *ICML*, pages 9722–9732, 2020b.

Y. Vorobeychik and D. M. Reeves. Equilibrium analysis of dynamic bidding in sponsored search auctions. *International Journal of Electronic Business*, 6:172–193, 2008.

R. Wang. Auctions versus posted-price selling. *The American Economic Review*, pages 838–851, 1993.

X. Wei, H. Yu, and M. J. Neely. Online primal-dual mirror descent under stochastic constraints. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 4(2):1–36, 2020.

X. Yi, X. Li, L. Xie, and K. H. Johansson. Distributed online convex optimization with time-varying coupled inequality constraints. *IEEE Transactions on Signal Processing*, 68:731–746, 2020.

H. Yu and M. J. Neely. A low complexity algorithm with o($\sqrt{T}$) regret and o(1) constraint violations for online convex optimization with long term constraints. *Journal of Machine Learning Research*, 21(1):1–24, 2020.

H. Yu, M. Neely, and X. Wei. Online convex optimization with stochastic constraints. *Advances in Neural Information Processing Systems*, 30, 2017.

J. Yuan and A. Lamperski. Online convex optimization for cumulative constraints. *Advances in Neural Information Processing Systems*, 31, 2018.

W. Zhang, Y. Zhang, B. Gao, Y. Yu, X. Yuan, and T.-Y. Liu. Joint optimization of bid and budget allocation in sponsored search. In *SIGKDD*, pages 1177–1185, 2012.

Z. Zhou, R. Xu, and J. Blanchet. Learning in generalized linear contextual bandits with stochastic delays. In *NeurIPS*, volume 32, 2019.

M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, pages 928–936, 2003.

D. Zuckerman. Linear degree extractors and the inapproximability of max clique and chromatic number. *Theory of Computing*, 3(6):103–128, 2007.

APPENDIX $\mathcal{A}$

Part I

A.1 Chapter 3

A.1.1 An Analytical Expression of $F_{Y_{\lambda T}}$ for the RV Setting with Linear Discount

We study the cumulative distribution function of the random variable $Y_{\lambda T}$ so as to unveil its dependence on F. We perform our analysis for the specific case of a linear discount function; thus:

$$Y_{\lambda T} = \max_{i \in \{1,\dots,N_T\}} V_i \left(1 - \frac{W_i}{T}\right).$$

The results presented in the following crucially rely on some properties of Poisson processes.

First, we introduce some auxiliary definitions and results.

Proposition A.1 (Ross et al. (1996))**.** *The random variable W_i representing the arrival time of agent i has a Gamma distribution $\Gamma(i, \lambda)$, with shape parameter $i > 0$ and rate parameter $\lambda > 0$, whose probability density function is defined as follows:*

$$f_{W_i}(w) \coloneqq \frac{\lambda^i w^{i-1}}{(i-1)!}\, e^{-\lambda w}, \quad \text{for every } w \in [0, T].$$

Theorem A.1 (Pinsky and Karlin (2010))**.** *Let $W_1, W_2, \dots$ be random variables representing the arrival times in a Poisson process with rate parameter $\lambda > 0$. Conditioned on the event $N_T = n$, the variables $W_1, \dots, W_n$ have a joint probability density function defined as follows:*

$$f_{W_1,\dots,W_n | N_T = n}(w_1, \dots, w_n) = n!\, T^{-n}, \quad \text{for } 0 < w_1 < \dots < w_n \leq T.$$

Intuitively, as discussed in (Ross et al., 1996), a consequence of Theorem A.1 is that, conditioned on the event $N_T = n$, the times $W_1, \ldots, W_n$ at which the n arrivals occur, considered as unordered random variables, are distributed uniformly and independently in the interval $[0, T]$. This is the crucial observation that allows to derive the following theorem.

Theorem A.2. *The random variable representing the maximum discounted valuation of agents arriving in the overall time period $[0, T]$ conditioned on the event that $N_T = n$ is defined as follows:*

$$Y_{\lambda T | N_T = n} \coloneqq \max_{i \in \{1,\ldots,n\}} V_i \, U_i, \quad \text{where } U_i \sim \mathcal{U}(0,1). \,^{1}$$

Proof. Given the symmetry of the functional $\max_{i \in \{1,\ldots,N_T\}} V_i \, \left(1 - \frac{W_i}{T}\right)$ and Theorem A.1, we can write the following:

$$\begin{aligned}
\mathbb{P}\left\{Y_{\lambda T} = y \mid N_T = n\right\} &= \mathbb{P}\left\{ \max_{i \in \{1,\ldots,N_\tau\}} V_i \left(1 - \frac{W_i}{T}\right) = y \mid N_\tau = n \right\} \\
&= \mathbb{P}\left\{ \max_{i \in \{1,\ldots,n\}} V_i \left(1 - \frac{\tilde{U}_i}{T}\right) = y \right\}
\end{aligned}$$

where $\tilde{U}_i$ is a random variable distributed according to $\mathcal{U}(0, T)$, which is a continuous uniform distribution with support $[0, T]$. Letting $U_i \coloneqq \left(1 - \frac{\tilde{U}_i}{T}\right)$, it is easy to show that $U_i \sim \mathcal{U}(0, 1)$. Formally, for every $x \in [0, 1]$, the cumulative distribution function F_{U_i} of U_i is defined as follows:

$$\begin{aligned}
F_{U_i}(x) \coloneqq \mathbb{P}\left\{U_i \leq x\right\} &= \mathbb{P}\left\{ \left(1 - \frac{\tilde{U}_i}{T}\right) \leq x \right\} = \mathbb{P}\left\{ T(1-x) \leq \tilde{U}_i \right\} \\
&= 1 - \mathbb{P}\left\{ \tilde{U}_i \leq T(1-x) \right\} = 1 - \frac{T(1-x)}{T} = x
\end{aligned}$$

Moreover, for $x < 0$ it holds $F_{U_i}(x) = 0$, while for $x > 1$ it holds $F_{U_i}(x) = 1$. Thus, F_{U_i} is the cumulative distribution function of a random variable drawn from a uniform with support $[0, 1]$. □

In the following, we denote by Z a product variable $V\,U$, where V and U are random variables distributed according to F and $\mathcal{U}(0, 1)$, respectively. Moreover, we let $Z_i \coloneqq V_i \, U_i$ be the variable Z referred to a specific agent i. Theorem A.2 allows us to express $F_{Y_{\lambda T | N_T = j}}$ as follows:

$$F_{Y_{\lambda T | N_T = j}}(x) = F_{\max_{i \in \{1,\ldots,j\}} Z_i}(x) = \mathbb{P}\left\{ \bigcap_{i=1}^{j} Z_i \leq x \right\} = \prod_{i=1}^{j} \mathbb{P}\left\{Z_i \leq x\right\} = \left[F_Z(x)\right]^j .$$

Hence, we can write $F_{Y_{\lambda T}}$ as:

$$F_{Y_{\lambda T}}(x) = \sum_{j=1}^{\infty} \frac{(\lambda T)^j e^{-\lambda T}}{j!} \left[F_Z(x)\right]^j ,$$

where

$$F_Z(x) = \begin{cases} x \int_1^h \frac{1}{v} f(v) \mathrm{d}v & \text{if } x \in [0, 1) \\ F(x) + x \int_x^h \frac{1}{v} f(v) \mathrm{d}v & \text{if } x \in [1, h] \end{cases} . \tag{A.1}$$

Thus, it is easy to see that $F_{Y_{\lambda T}}$ depends on F and f, which are the cumulative distribution function and the probability density function of agents' initial valuations, respectively.

It remains to show how to derive the expression of F_Z in Equation (A.1). Notice that, since $U \sim \mathcal{U}(0, 1)$, the probability density function of U is defined as $f_U(u) = \mathbf{1}_{[0,1]}(u)$, while its

[1] We denote by $\mathcal{U}(a, b)$ a continuous uniform distribution over the interval $[a, b]$.

cumulative distribution function is $F_U(u) = u\mathbf{1}_{[0,1]}(u)$. The support of Z is $[0, h]$, being V defined on $[1, h]$.

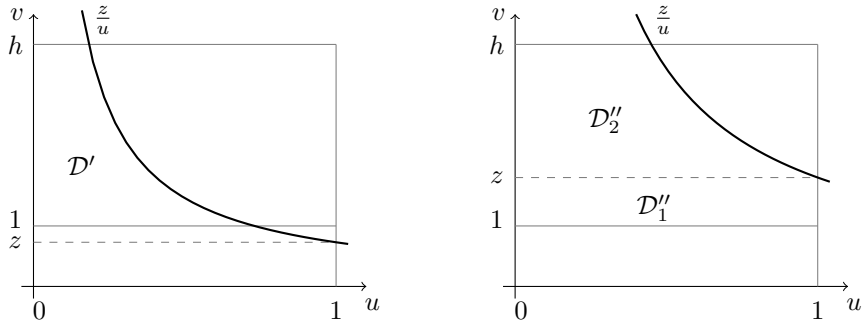
$$\begin{aligned}
F_Z(z) =& \mathbb{P}\{V\,U \leq z\} = \mathbb{P}\left\{U \leq \frac{z}{V}\right\} = \\
=& \mathbf{1}_{[0,1)}(z) \iint_{\mathcal{D}'} f(v) f_U(u) \mathrm{d}v\mathrm{d}u + \mathbf{1}_{[1,h]}(z) \iint_{\mathcal{D}''} f(v) f_U(u) \mathrm{d}v\mathrm{d}u = \\
=& \mathbf{1}_{[0,1)}(z) \int_1^h f(v) \int_0^{z/v} f_U(u) \mathrm{d}u\mathrm{d}v + \\
&+ \mathbf{1}_{[1,h]}(z) \left(\int_1^z f(v)\mathrm{d}v \int_0^1 f_U(u)\mathrm{d}u + \int_z^h f(v) \int_0^{z/v} f_U(u)\mathrm{d}u\mathrm{d}v \right) \\
=& \mathbf{1}_{[0,1)}(z) \left(z \int_1^h \frac{1}{v} f(v)\mathrm{d}v \right) + \mathbf{1}_{[1,h]}(z) \left(F(z) + z \int_z^h \frac{1}{v} f(v)\mathrm{d}v \right),
\end{aligned}$$

where the domains of integration $\mathcal{D}'$ and $\mathcal{D}''$ are defined as:

$$\mathcal{D}' \coloneqq \left\{(u, v) : 0 \leq u \leq \frac{z}{v}, 1 \leq v \leq h\right\}$$

$$\mathcal{D}'' \coloneqq \mathcal{D}''_1 \cup \mathcal{D}''_2 \coloneqq \{(u, v) : 0 \leq u \leq 1, 1 \leq v \leq z\} \cup \left\{(u, v) : 0 \leq u \leq \frac{z}{v}, z < v \leq h\right\}$$

See also Figure A.1 for a graphical representation of the domains.



(a) $\mathcal{D}'$ *for* $z \in [0, 1)$. **(b)** $\mathcal{D}''$ *for* $z \in [1, h]$.

Figure A.1: *Graphical representation of the domain of integration* $\mathcal{D}'$ *and* $\mathcal{D}''$.

A.2 Chapter 4

A.2.1 `Delayed-UCB1` Baseline

We show how to apply the `Delayed-UCB1` algorithm, provided by Joulani et al. [2013] and originally designed for the Delayed-MAB setting, to the TP-MAB setting. In the TP-MAB problem, the realization of the cumulative reward r_t^i is observed after $\tau_{\max} - 1$ rounds from the pull of the arm. As a consequence, one always waits for $\tau_{\max} - 1$ rounds before collecting the reward from a pull. This approach, corresponds to a delayed-feedback MAB setting in which the delay is known and deterministic. After such a delay, the learner updates the policy $\mathfrak{U}_{\texttt{D-UCB1}}$ of `Delayed-UCB1` with the value of the cumulative reward.

Algorithm A.1 `Delayed-UCB1`

1: **for** $t \in \{1, \dots, \tau_{\max}\}$ **do** ▷ init phase
2: Pull arm $i_t = ((t-1) \mod K) + 1$
3: **end for**
4: **for** $t \in \{\tau_{\max}+1, \dots, T\}$ **do** ▷ loop phase
5: **for** $i \in \{1, \dots, K\}$ **do**
6: $\hat{R}^i_{t-1} \leftarrow \frac{1}{s^i_{t-1}} \sum_{h=1}^{t-\tau_{\max}} r^i_h \mathbf{1}_{\{i_h = i\}}$
7: $c^i_{t-1} \leftarrow \overline{R}^i \sqrt{\frac{2\ln(t-1)}{s^i_{t-1}}}$
8: $u^i_{t-1} \leftarrow \hat{R}^i_{t-1} + c^i_{t-1}$
9: **end for**
10: Pull arm $i_t = \text{argmax}_{i \in [K]}\, u^i_{t-1}$
11: Observe reward $r^{i_{t-\tau_{\max}+1}}_{t-\tau_{\max}+1}$ of the arms pulled at round $t - \tau_{\max} + 1$
12: **end for**

The pseudo-code of the `Delayed-UCB1` algorithm applied to a generic TP-MAB setting is reported in Algorithm A.1. During the initialization phase, all arms are pulled in a round robin fashion until at least one reward is collected (Line 2). Subsequently, at each round t, the learner computes the empirical mean $\hat{R}^i_{t-1}$ of the cumulative rewards collected up to round $t-1$ (Line 6), where $s^i_{t-1} := \sum_{h=1}^{t-\tau_{\max}} \mathbf{1}_{\{i_h = i\}}$ is the number of complete reward observed so far for arm i, and the confidence interval c^i_{t-1} (Line 7). Finally, the learner pulls the arm with the largest upper confidence bound u^i_{t-1} (Line 10), and observes the reward corresponding to the pull occurred at round $t - \tau_{\max} + 1$ (Line 11). When no sample is available for an arm i its upper bound is set to $+\infty$. We provide the following upper bound on the regret of the `Delayed-UCB1` algorithm (see Joulani et al. [2013]).

Theorem A.3. *The pseudo-regret of* `Delayed-UCB1` *after $T \in \mathbb{N}^*$ rounds in the TP-MAB setting is:*

$$R^T(\mathfrak{U}_{\texttt{D-UCB1}}) \leq \sum_{i:\mu_i<\mu^*} \frac{8(\overline{R}^i)^2 \ln T}{\Delta_i} + \left(1 + \frac{\pi^2}{3} + \tau_{\max}\right) \sum_{i:\mu_i<\mu^*} \Delta_i. \tag{A.2}$$

Proof. The theorem follows from Theorem 7 by Joulani et al. (2013), where the expected value of the maximum number of missing feedback of arm i during the first t time steps is $\mathbb{E}[G^*_{i,t}] < \tau_{\max}$, where $G^*_{i,t}$ is the maximum number of missing feedbacks during the first t rounds for arm i. □

A.2.2 Additional Experiments

In what follows, we provide a detailed description of those setting which have been presented in Section 4.4 and further experiments confirming what has been showed in the chapter.

Setting #2 (main scenario). In this setting, each arm is described by a maximum reward $\overline{R}^i$ and two vectors $\boldsymbol{a}^i := \left(a^i_1, \dots, a^i_\alpha\right)$ and $\boldsymbol{b}^i := \left(b^i_1, \dots, b^i_\alpha\right)$ of length α. The aggregated reward $Z^i_{t,k}$ are distributed as $\mathcal{D}^i_k = \frac{\overline{R}^i}{\alpha} Beta(a^i_k, b^i_k)$, $\forall k \in [\alpha]$. The results presented in the chapter are those corresponding to $\boldsymbol{a}^i := \mathbf{1}_\alpha$ and $\boldsymbol{b}^i := \mathbf{1}_\alpha$, where $\mathbf{1}_\alpha$ is a vector of length α whose elements are all 1. This setting corresponds to a uniform distribution over $\frac{\overline{R}^i}{\alpha}$ for each variable $Z^i_{t,k}$. The corresponding results are presented in Section 4.4. The regret over the entire time horizon is presented in Figure A.2.

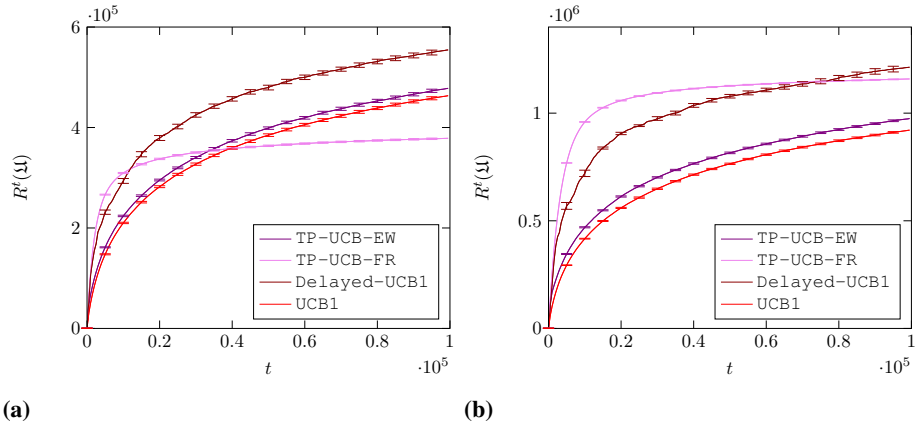


Figure A.2: *Experiments for Setting #2 and uniform reward distribution: (a)* $\tau_{\max} = 100$*,* $\alpha = 10$*, (b)* $\tau_{\max} = 200$*,* $\alpha = 20$*.*

Spotify Setting. The original Spotify dataset Brost et al. (2019) consists of listening sessions with levels of appreciation for each song associated to a user on the Spotify service. Each listening session is truncated to 20 tracks (songs). Each row corresponds to the playback of one track pertaining to a specific listening session. The dataset describes how users sequentially interact with the streamed content they are presented with. More precisely, it contains information about when a user skips the playback of a track.

We preprocessed the available data as follows. At first, for computational reasons we analysed only a fraction of the Spotify dataset. Since we are interested in the listening sessions linked to a playlist, from that initial dataset we drop all the data associated with a `context_type` field context which is different from *editorial playlist*. Moreover, we discarded all the listening sessions with less than 20 songs and/or the user changed playlist during a single listening session (`context_switch = true`). This way, each listening sessions is composed of 20 song coming from a single playlist. We selected the 6 most listened playlists having no overlapping songs, and extracted from the dataset the listening sessions corresponding to them.

The process of recommending the playlists is modeled as follows.

Example A.1 (Playlist Recommendation Problem - Reprise)**.** *When a new user accesses the system, a playlist is proposed. This action corresponds to the selection of an arm* i *by the recommendation algorithm. The user will start the reproduction of the playlist, composed of exactly* $N = 20$ *songs. For each song, at any time, the agent could decide to skip to the next song until the end of the playlist. We aim at finding the playlist that maximizes the overall listening time. Each song has a reward equal to* `skip_1`*,* `skip_2`*,* `skip_3`*, and* `not_skipped`*, representing increasing level of interest from the user. These levels corresponds to the the realization of instantaneous reward* $X^i_{t,j}$ *of Bernoulli r.v. that takes the value of* 1 *if the user has reached at least the corresponding level and* 0 *otherwise; The vector* $\boldsymbol{X}^i_t$ *has size equal to the number of songs of a playlist (*i.e.*, aggregated rewards) times the number instant rewards returned by a song (*i.e.*,* ϕ*), and in this case* $\tau_{\max} = 20 \times 4 = 80$*. A summary of the expected rewards of the different playlists is provided in Table A.1. Figure A.3 shows an example of the reproduction of part of* 5 *songs of a playlist. Songs* 1 *and* 3 *were listened completely, while Song* 2 *was listened up to level the* `skip_2`*. Song* 4 *and Song* 5 *were entirely skipped.*

In what follows we provide additional experiments.

1 1 1 1	1 1 0 0	1 1 1 1	0 0 0 0	0 0 0 0
Song 1	Song 2	Song 3	Song 4	Song 5

Figure A.3: *Example of a realization of an a subset of a playlist in the Spotify Setting.*

	$i=1$	$i=2$	$i=3$	$i=4$	$i=5$	$i=6$
μ^i	38.59	52.35	38.44	43.89	23.48	36.20
σ^i	21.83	20.11	23.09	23.14	23.48	23.8

Table A.1: *Description of the arms in the Spotify Setting.*

Setting #2.1 In this experiment, the setting is the same as the one in Setting #2, except that we designed the rewards s.t. the first aggregated rewards after the pull are smaller than the last ones. Specifically, the distribution are defined by the following vectors:

- $\tau_{\max} = 100$, $\alpha = 10$:

$$\boldsymbol{a}^i = (2, 4, 6, 8, 10, 10, 10, 10, 10, 10);$$
$$\boldsymbol{b}^i = (10, 10, 10, 10, 10, 10, 8, 6, 4, 2);$$

- $\tau_{\max} = 200$, $\alpha = 20$:

$$\boldsymbol{a}^i = (2, 4, \ldots, 18, 20, \ldots, 20);$$
$$\boldsymbol{b}^i = (20, \ldots, 20, 18, \ldots, 4, 2);$$

- $\tau_{\max} = 100$, $\alpha = 50$:

$$\boldsymbol{a}^i = (2, 4, \ldots, 48, 50, \ldots, 50);$$
$$\boldsymbol{b}^i = (50, \ldots, 50, 48, \ldots, 4, 2);$$

- $\tau_{\max} = 200$, $\alpha = 100$:

$$\boldsymbol{a}^i = (2, 4, \ldots, 98, 100, \ldots, 100);$$
$$\boldsymbol{b}^i = (100, \ldots, 100, 98, \ldots, 4, 2).$$

The corresponding results are provided in Figure A.4. They are in line with the ones of Setting #2.

Setting #2.2 In this experiment, the setting is the same as the one in Setting #2, except that we designed the rewards s.t. the first aggregated rewards after the pull are larger than the last ones.

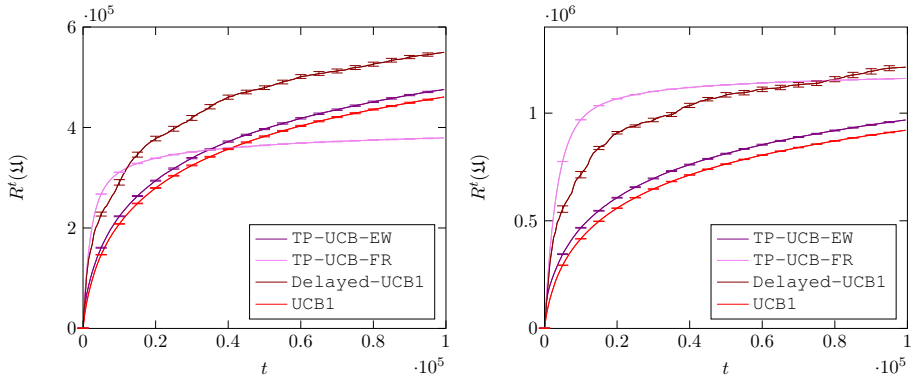
Specifically, the distribution are defined by the following vectors:

- $\tau_{\max} = 100$, $\alpha = 10$:

$$\boldsymbol{a}^i = (10, 10, 10, 10, 10, 10, 8, 6, 4, 2);$$
$$\boldsymbol{b}^i = (2, 4, 6, 8, 10, 10, 10, 10, 10, 10);$$

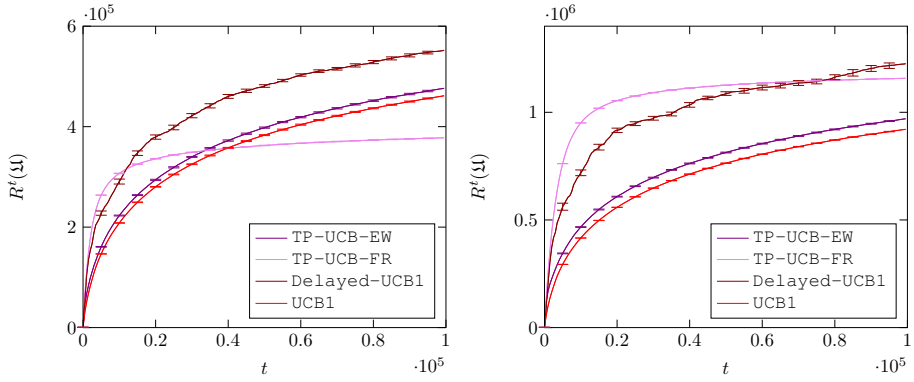
- $\tau_{\max} = 200$, $\alpha = 20$:

$$\boldsymbol{a}^i = (20, \ldots, 20, 18, \ldots, 4, 2);$$
$$\boldsymbol{b}^i = (2, 4, \ldots, 18, 20, \ldots, 20);$$



(a) **(b)**

Figure A.4: *Experiments for Setting #2.1: (a)* $\tau_{\max} = 100$, $\alpha = 10$, *(b)* $\tau_{\max} = 200$, $\alpha = 20$.



(a) **(b)**

Figure A.5: *Experiments for Setting #2.2: (a)* $\tau_{\max} = 100$, $\alpha = 10$, *(b)* $\tau_{\max} = 200$, $\alpha = 20$.

- $\tau_{\max} = 100$, $\alpha = 50$:

$$\boldsymbol{a}^i = (50, \ldots, 50, 48, \ldots, 4, 2);$$
$$\boldsymbol{b}^i = (2, 4, \ldots, 48, 50, \ldots, 50);$$

- $\tau_{\max} = 200$, $\alpha = 100$:

$$\boldsymbol{b}^i = (100, \ldots, 100, 98, \ldots, 4, 2);$$
$$\boldsymbol{a}^i = (2, 4, \ldots, 98, 100, \ldots, 100).$$

The corresponding results are provided in Figure A.5. They are in line with the ones of Setting #2.

Setting #2.3 Finally, in this experiment, the setting is the same as the one in Setting #2, except that the reward distributions are randomly chosen.

Specifically, the distribution sampled used in the experiments are:

- $\tau_{\max} = 100$, $\alpha = 10$:

$$\boldsymbol{a}^i = (7, 7, 1, 5, 9, 8, 7, 5, 8, 6);$$
$$\boldsymbol{b}^i = (10, 4, 9, 3, 5, 3, 2, 10, 5, 9);$$

- $\tau_{\max} = 200$, $\alpha = 20$:

$$\boldsymbol{a}^i = (10, 3, 5, 2, 2, 6, 8, 9, 2, 6, 7, 6, 10, 4, 9, 8, 8, 9, 5, 1);$$
$$\boldsymbol{b}^i = (9, 1, 2, 7, 1, 10, 8, 6, 4, 6, 2, 4, 10, 4, 4, 3, 9, 8, 2, 2);$$

- $\tau_{\max} = 100$, $\alpha = 50$:

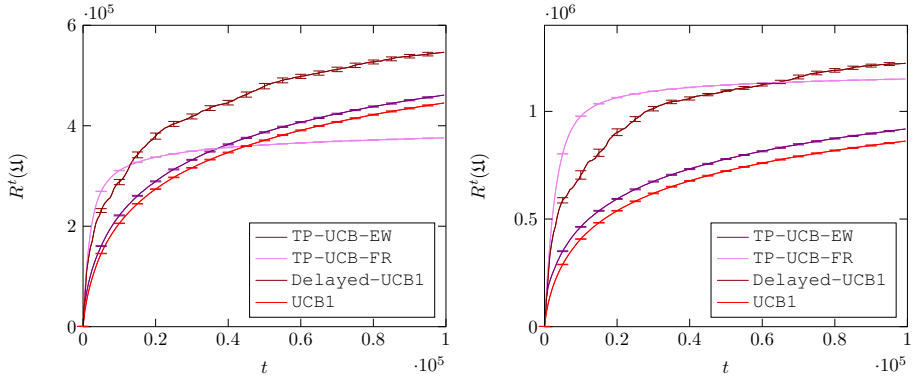
$$\boldsymbol{a}^i = (6, 9, 8, 2, 5, 9, 5, 2, 9, 6, 9, 4, 10, 9, 10, 5, 8, 2, 10, 7, 6, 10, 4, 5, 3, 4, 3, 1, 10, 5, 8, 2, 2, 3, 3, 1, 2, 9, 7, 9, 5, 9, 4, 4, 10, 7, 10, 5, 8, 8);$$
$$\boldsymbol{b}^i = (6, 2, 6, 10, 2, 8, 10, 6, 4, 4, 1, 5, 2, 4, 6, 3, 6, 7, 1, 2, 3, 4, 1, 10, 9, 10, 2, 1, 2, 4, 10, 10, 2, 7, 2, 6, 2, 1, 10, 1, 4, 3, 2, 8, 4, 1, 1, 9, 7, 10);$$

- $\tau_{\max} = 200$, $\alpha = 100$:

$$\boldsymbol{a}^i = (2, 5, 2, 4, 2, 5, 6, 7, 3, 1, 9, 8, 1, 10, 2, 7, 4, 5, 6, 8, 10, 3, 4, 1, 3, 3, 6, 9, 5, 2, 10, 8, 3, 1, 8, 7, 10, 9, 5, 6, 7, 5, 3, 9, 1, 8, 2, 6, 1, 9, 5, 3, 4, 8, 6, 10, 5, 6, 10, 10, 3, 5, 7, 7, 2, 1, 10, 4, 6, 3, 4, 4, 8, 7, 10, 7, 1, 7, 10, 7, 1, 3, 8, 2, 5, 3, 8, 9, 8, 9, 10, 1, 1, 8, 6, 5, 8, 1, 7, 4);$$
$$\boldsymbol{b}^i = (9, 2, 3, 1, 7, 7, 6, 1, 4, 1, 1, 9, 10, 2, 4, 2, 10, 4, 5, 5, 3, 2, 8, 7, 2, 1, 5, 8, 2, 5, 3, 9, 6, 2, 3, 5, 1, 1, 1, 4, 5, 9, 6, 6, 10, 1, 10, 8, 8, 7, 6, 9, 3, 4, 7, 10, 5, 1, 3, 3, 5, 6, 6, 6, 2, 6, 10, 1, 1, 5, 3, 3, 10, 5, 6, 7, 9, 3, 5, 2, 8, 4, 1, 5, 3, 9, 2, 5, 7, 6, 5, 7, 2, 2, 9, 8, 8, 6, 6, 2).$$

The corresponding results are provided in Figure A.6. They are in line with the ones of Setting #2.

Summary for Setting #2 The overall results for the previous setting #2, #2.1, #2.2, and #2.3 are reported in Table A.2, A.3, A.4, A.5.



(a) **(b)**

Figure A.6: *Experiments for Setting #2.3: (a)* $\tau_{\max} = 100$, $\alpha = 10$, *(b)* $\tau_{\max} = 200$, $\alpha = 20$.

$\tau_{\max}$	α	Scenario	Learner	Regret	Confidence Interval
100	10	1	TP-UCB-FR	379407.7536	641.3890868
100	10	1	TP-UCB-EW	476211.7734	1379.593546
100	10	1	Delayed-UCB1	550020.3093	3383.218936
100	10	1	UCB1	461295.3133	1198.377002
100	10	2	TP-UCB-FR	378590.4996	1444.810301
100	10	2	TP-UCB-EW	478543.3454	3282.169025
100	10	2	Delayed-UCB1	556264.2577	4563.491842
100	10	2	UCB1	464045.2915	3127.506071
100	10	3	TP-UCB-FR	377928.2537	550.2470147
100	10	3	TP-UCB-EW	477050.7314	1370.65113
100	10	3	Delayed-UCB1	552254.3013	2871.253395
100	10	3	UCB1	462051.9847	1022.873814
100	10	4	TP-UCB-FR	376004.9497	713.1333679
100	10	4	TP-UCB-EW	461523.0728	1159.826331
100	10	4	Delayed-UCB1	546401.0207	3116.186928
100	10	4	UCB1	445761.5334	1160.681727

Table A.2: *Summary of result for setting #2,* $\tau_{\max} = 100$, $\alpha = 10$.

$\tau_{\max}$	α	Scenario	Learner	Regret	Confidence Interval
200	20	1	TP-UCB-FR	1161392.507	653.9898656
200	20	1	TP-UCB-EW	969119.3579	2376.133933
200	20	1	Delayed-UCB1	1215396.1	11238.84718
200	20	1	UCB1	921857.7185	1262.074342
200	20	2	TP-UCB-FR	1159038.888	1855.393219
200	20	2	TP-UCB-EW	976387.8607	4103.793005
200	20	2	Delayed-UCB1	1214717.526	12958.26024
200	20	2	UCB1	922123.0453	3911.196296
200	20	3	TP-UCB-FR	1158406.886	719.1511692
200	20	3	TP-UCB-EW	971023.1429	2128.831649
200	20	3	Delayed-UCB1	1225998.654	12586.53841
200	20	3	UCB1	922097.5566	1084.342302
200	20	4	TP-UCB-FR	1150596.776	1373.38433
200	20	4	TP-UCB-EW	919231.1795	2971.38115
200	20	4	Delayed-UCB1	1224143.761	6816.6797
200	20	4	UCB1	863043.4276	2568.233259

Table A.3: *Summary of result for setting #2,* $\tau_{\max} = 200$*,* $\alpha = 20$*.*

$\tau_{\max}$	α	Scenario	Learner	Regret	Confidence Interval
100	50	1	TP-UCB-FR	280850.7628	200.0363298
100	50	1	TP-UCB-EW	470206.8356	610.8394845
100	50	1	Delayed-UCB1	555004.3727	3611.482174
100	50	1	UCB1	461125.7678	433.1909748
100	50	2	TP-UCB-FR	280469.8885	600.1158378
100	50	2	TP-UCB-EW	470948.6985	1810.491059
100	50	2	Delayed-UCB1	551713.5918	3167.855141
100	50	2	UCB1	460454.4842	1535.465475
100	50	3	TP-UCB-FR	280432.6875	194.6246275
100	50	3	TP-UCB-EW	470851.5341	678.1378134
100	50	3	Delayed-UCB1	552354.8852	2784.797814
100	50	3	UCB1	461262.8902	406.9041603
100	50	4	TP-UCB-FR	277350.6683	357.2049513
100	50	4	TP-UCB-EW	431428.2109	845.9105653
100	50	4	Delayed-UCB1	533550.167	6134.964191
100	50	4	UCB1	419308.3464	840.25097

Table A.4: *Summary of result for setting #2,* $\tau_{\max} = 100$*,* $\alpha = 50$*.*

$\tau_{\max}$	α	Scenario	Learner	Regret	Confidence Interval
200	100	1	TP-UCB-FR	998723.9102	348.3923308
200	100	1	TP-UCB-EW	962166.9976	1574.53646
200	100	1	Delayed-UCB1	1217054.205	13791.12121
200	100	1	UCB1	922801.461	681.1463488
200	100	2	TP-UCB-FR	997866.0232	1163.306506
200	100	2	TP-UCB-EW	962888.2947	2886.588981
200	100	2	Delayed-UCB1	1223555.271	13076.51935
200	100	2	UCB1	924666.3352	1936.282782
200	100	3	TP-UCB-FR	995734.719	386.1528975
200	100	3	TP-UCB-EW	962419.0355	1671.591765
200	100	3	Delayed-UCB1	1224181.588	14560.25523
200	100	3	UCB1	923018.9128	593.7216922
200	100	4	TP-UCB-FR	996058.5901	681.2301995
200	100	4	TP-UCB-EW	937032.8774	1815.90584
200	100	4	Delayed-UCB1	1214671.825	12459.63383
200	100	4	UCB1	893569.8466	1098.403796

Table A.5: *Summary of result for setting #2,* $\tau_{\max} = 200$*,* $\alpha = 100$*.*

	$\boldsymbol{a}^i$	$\boldsymbol{b}^i$
Scenario 1	[8, 2, 8, 7, 1, 5, 6, 3, 3, 10]	[7, 2, 2, 2, 4, 4, 1, 7, 1, 2]
Scenario 2	[7, 9, 9, 5, 8, 8, 10, 4, 7, 2]	[6, 4, 5, 10, 3, 7, 4, 6, 2, 2]
Scenario 3	[1, 9, 8, 4, 2, 8, 7, 5, 4, 1]	[4, 10, 3, 2, 4, 8, 7, 6, 9, 3]
Scenario 4	[2, 10, 8, 3, 10, 7, 7, 9, 8, 6]	[8, 8, 4, 9, 10, 4, 1, 6, 6, 6]
Scenario 5	[1, 9, 3, 5, 10, 3, 7, 10, 5, 8]	[2, 2, 9, 1, 2, 4, 3, 1, 5, 1]
Scenario 6	[8, 6, 3, 3, 8, 6, 9, 7, 9, 9]	[1, 10, 2, 9, 10, 2, 7, 4, 5, 9]
Scenario 7	[10, 7, 8, 7, 10, 10, 4, 1, 1, 3]	[5, 9, 10, 5, 6, 2, 8, 5, 5, 7]
Scenario 8	[7, 7, 1, 3, 3, 4, 5, 6, 1, 1]	[8, 7, 3, 8, 10, 2, 3, 6, 7, 1]
Scenario 9	[10, 8, 7, 8, 1, 2, 8, 3, 1, 1]	[10, 10, 3, 6, 2, 9, 6, 4, 7, 8]
Scenario 10	[2, 1, 10, 8, 10, 6, 2, 10, 5, 3]	[7, 5, 2, 9, 4, 1, 7, 8, 6, 4]

Table A.6: *Parameters used in Setting #4.*

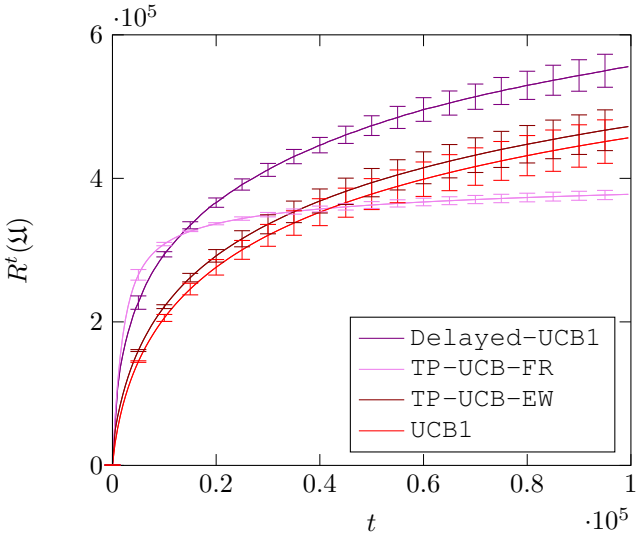


Figure A.7: *Experiments for Setting # 4:* $\tau_{\max} = 100$, $\alpha = 10$

Setting #4 In this setting, each arm is described by a maximum reward $\overline{R}^i = \tau_{\max} \cdot i$, and two vectors $\boldsymbol{a}^i = \left(a^i_1, \ldots, a^i_\alpha\right)$ and $\boldsymbol{b}^i = \left(b^i_1, \ldots, b^i_\alpha\right)$ of length α. The aggregated rewards $Z^i_{t,k}$ are distributed as $\mathcal{D}^i_k = \frac{\overline{R}^i}{\alpha} Beta(a^i_k, b^i_k)$, $\forall k \in [\alpha]$. In this experiment, we fix $\tau_{\max} = 100$, $\alpha = 10$, $T = 10^5$, and we design ten scenarios differing in the vectors $\boldsymbol{a}^i$ and $\boldsymbol{b}^i$. The parameters characterizing such randomly generated scenarios are reported in Table A.6. The results for each scenario are averaged over 50 independent runs. In Figure A.7, we provide the average result over the 10 scenarios, with whiskers corresponding to 95% confidence intervals.

Figure A.7 shows an aggregated result on the pseudo-regret $R^t(\mathfrak{U})$ for the analysed algorithms. Even over randomly generated scenarios we see that the proposed method are able to provide a significant improvement over the `Delayed-UCB1` algorithm. Moreover, consistently the `TP-UCB-FR` algorithm result to be the best one at the end of the analysed time horizon $T = 10^5$. Conversely, for shorter time horizon ($T \leq 0.35 \cdot 10^5$) the algorithm performing the best among the ones for the

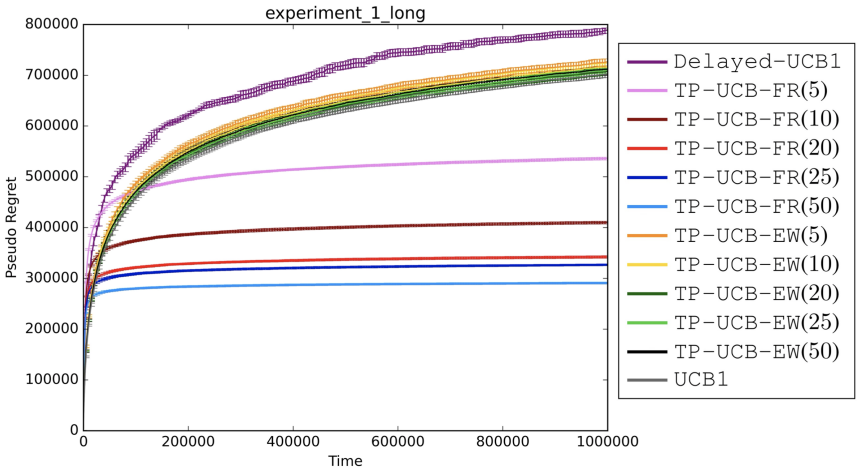


Figure A.8: *Experiments for Setting # 5:* $\tau_{\max} = 100$, $\alpha = 20$

TP-MAB setting is the TP-UCB-EW, which strengthen the idea that this algorithm is better suited for shorter time horizons.

Setting #5 Finally, we provide an experiment over a longer time horizon of $T = 10^6$ in the same configuration depicted by Setting #1. The pseudo-regret over time for this experiment is provided in Figure A.8. Let us focus on the regret of TP-UCB-FR(20), *i.e.*, the TP-UCB-FR algorithm where parameter α corresponds to the one of the environment, and compare it with the regret of Delayed-UCB1. The regret of TP-UCB-FR(20) (red line) has a slower growth w.r.t. Delayed-UCB1 (purple line), and, consequently, the difference in terms of regret increases (logarithmically) over time. The parameter influencing the regret of TP-UCB-FR is α, which characterizes the specific setting we are tackling. More specifically, if we fix the other parameters (e.g., $\tau_{\max}$) and increase the value of alpha, we have a proportional improvement in the upper bound of the regret of TP-UCB-FR. Therefore, we expect to have an even larger improvement of our algorithm when the value of α is large.

A.2.3 Additional Real-World Examples

In this section, we report some additional real-world examples which can be modeled through the TP-MAB setting. The following scenarios are characterized by the α-smoothness property with different values of the α parameters.

Example A.2 (E-commerce)**.** *An agent periodically receives a batch of identical items to sell on an e-commerce platform. Every time a slot of N items arrives, the agent decides a price p_i to post on a website, which corresponds to the arm i_t chosen for the round t. The selected time horizon to sell the items, which are perishable, is one month. Each day, the seller checks how many items have been ordered and collects the payments* (i.e.*, rewards). In this example, the maximum delay is $\tau_{\max} = 30$ days, and one round is equal to 1 day. The upper bound on the cumulative reward is $\overline{R}^i = p_i N$. Notice that the partial reward of each round is also upper bounded by $p_i N$. This implies that the reward has no structure, and consequently the α-smoothness in this setting holds with $\alpha = 1$.*

Example A.3 (Lottery Ticket)**.** *There are K different lotteries to choose from. Lottery $i \in [K]$ has N winning scratch cards, each with a prize of M. The probability to extract a winning ticket in lottery i is p_i. The player has to choose a lottery at each time step. At each round, the player buys n tickets and sequentially scratches them and observes the reward. If $N = 1$ the total amount the player can win is M and the reward is 1-smooth. Indeed, suppose that the first $n - 1$ tickets are not winning. This does not precludes the possibility of still gaining the maximum cumulative reward with the last ticket. Conversely, if $N = n$ the total amount the player can win is $\overline{R}^i = NM$, and the reward is n-smooth. More specifically, by scratching the first ticket, the player can get useful information on the cumulative reward if the reward is either zero or M. If the player observed a zero reward so far, the maximum achievable cumulative reward becomes $(N - 1)M$. Conversely if the player observed a positive reward, the overall reward is in the interval $[M, NM]$.*

APPENDIX B

Part II

B.1 Chapter 7

Theorem 7.3. *META-SD-NE is APX-Hard.*

Proof. let $\eta = \max\{c, (1 - \frac{1}{5}^5)\}$, where c is the constant factor approximation in Theorem 7.2. Notice that Theorem 7.2 holds even if we replace the approximation factor c with the weaker constant $\eta \geq c$.

Given an instance of 3-SAT-5 with clauses C and variables V, we build an instance of the META-SD-NE problem with the following structure. The tree of scene is composed by a line with a scene s_v for each $v \in V$ in an arbitrary order. Then, it follows a line that includes a scene s_c for each clause $c \in C$ in an arbitrary order. All the transition probabilities $\pi_{s,s'}$ are set to 1. The set of ads A includes two ads a_v and $a_{\sim v}$ for each variable $v \in V$. Let $\epsilon = 1 - \eta^{1/5}$, and let l denote a literal, *i.e.*, l is a variable or its negation. The qualities of the ads are defined as follows: $q_{a_v,s_v} = q_{a_{\sim v},s_v} = 1$ for each $v \in V$, and for each clause $c \in C$ the quality is $q_{a_l,s_c} = \epsilon$ if the literal l belongs to the clause. Every other quality is 0. Finally, let $\theta_a = 1$ for each $a \in A$.

In the following, we show that if there exists an assignment that satisfies all the clauses the utility is at least $|V| + |C|\eta^{4/5}(1 - \eta^{1/5})$, while if no assignment satisfies a η fraction of the clauses the utility is at most $|V| + (1 - \eta^{1/5})\eta|C|$. To conclude the proof notice that $|C| = \frac{3}{5}|V|$. Hence,

$$\frac{|V| + (1 - \eta^{1/5})\eta|C|}{|V| + \eta^{4/5}(1 - \eta^{1/5})|C|} = \frac{|V| + \frac{3}{5}(1 - \eta^{1/5})\eta|V|}{|V| + \frac{3}{5}\eta^{4/5}(1 - \eta^{1/5})|V|} = \frac{1 + \frac{3}{5}(1 - \eta^{1/5})\eta}{1 + \frac{3}{5}\eta^{4/5}(1 - \eta^{1/5})},$$

which is a constant strictly smaller than 1.

soundness. Consider an assignment L, *i.e.*, a set of literals including v or $\sim v$ for each variable $v \in V$, that satisfies all the clauses. We build an assignment x of ads to scenes as follows. For each variable v, let x assigns the ad a_l to the scene s_v, where $l \in \{v, \sim v\}$ is the literal *not* in the assignment L, *i.e.*, such that $l \in \{v, \sim v\} \setminus L$. Finally, let assign to each scene s_c, $c \in C$, an ad a_l such that the literal $l \in L$ satisfies the clause and belongs to L. Notice that this clause exists since the assignment satisfies all the clauses. Then, for each scene s_v, $v \in V$, we have that the expected value from the scene is 1. Moreover, for each scene s_c, $c \in C$, we have that the quality $q_{s_c,x(s_c)} = \epsilon$, while $\Xi(x, s_c)$ is at least $(1-\epsilon)^4$ since each literal appears in at most five clauses. Hence, the expected value of the allocation is at least

$$|V| + |C|(1-\epsilon)^4\epsilon = |V| + |C|\eta^{4/5}(1-\eta^{1/5}).$$

completeness. Consider an assignment of ads to nodes x. Let $V^* \subseteq V$ be the set of variables $v \in V$ such that $\tilde{q}(x, s_v) = 1$. Then, notice that the expected value of each scene s_v, $v \in V \setminus V^*$ is 0. Let $C^* \subseteq C$ be the set of clauses c such that an ad is assigned to s_c and $q_{x(s_c),s_c} = \epsilon$. Then, notice that the expected value of each scene s_c, $c \in C \setminus C^*$ is 0. We can split C^* in two subsets. The set $C_2 = \{c \in C^* : x(s_c) \in \{a_v, a_{\sim v}\}_{v \in V^*}\}$, while the set $C_1 = C^* \setminus C_2$. Then, we show that there exists a feasible assignment L that satisfies at least C_2 clauses, implying that $|C_2| \leq \eta|V|$. To see that, consider the assignment $L = \{l : a_l \in \{x(s_c)\}_{c \in C_2}\}$. As a first step, we show that the partial assignment is feasible. Suppose by contradiction that there exist two literals $v, \sim v$ belonging to L. Since $a_v \in L$, then there exists a clause $c \in C_2$ such that $x(s_c) = a_v$. Moreover, since $c \in C^*$, the scene s_c has positive quality and $x(s_v) \neq a_v$. Then, since $v \in V^*$, we have that $x(s_v) = a_{\sim v}$. By the definition of C^*, C_2 does not include any clause c such that $x(s_c) = a_{\sim v}$ since they have 0 utility (the ad has been converted in scene s_v and $\Xi(x, s_c) = 0$). Moreover, it is easy to see that the assignment satisfies all the clauses in C_2 by the definition of C^* and the qualities of the scenes.

Now, we bound the cardinality of C_1. Notice that since each variable $v \in V$ appears in 5 clauses (considering v and its negation $\sim v$), for each variable $v \notin V^*$ there exist at most 5 clauses $c \in C$ such that $q_{x(s_c),s_c} = \epsilon$ and $x(s_c) \in \{a_v, a_{\sim v}\}$. Then, for each $c \in C_1$ there exists a literal v such $x(s_c) = a_v$ or $x(s_c) = a_{\sim v}$, $x(s_v) \neq a_v$, and $x(s_v) \neq a_{\sim v}$. Recall that $V \setminus V^*$ is the set of variable v such that $x(s_v) \neq a_v$ and $x(s_v) \neq a_{\sim v}$. Since each variable appears in at most 5 clauses, we have that $|C_1| \leq 5(|V| - |V^*|)$. Moreover, by the definition of η it holds $5\epsilon = 5(1-\eta^{1/5}) = 1$. Hence, the total utility is at most

$$\begin{aligned}
|V^*| + \epsilon\big[|C_2| + |C_1|\big] &\leq |V^*| + \epsilon[|C_2| + 5(|V| - |V^*|)] \\
&= |V|^* + (|V| - |V^*|) + \epsilon|C_2| \\
&= |V| + \epsilon|C_2| \\
&\leq |V| + \epsilon\eta|C| \\
&= |V| + (1-\eta^{1/5})\eta|C|
\end{aligned}$$

This concludes the proof. □

Lemma 7.2. *Function $f(\cdot)$ is monotone submodular.*

Proof. It is easy to see that the function is monotone since adding an ad $a \in A$ to a scene $s \in S$ we increase the expected probability that ad a is converted. Moreover, since this ad a has no externalities on other ads, the convertion probability of the other ads does not decreases.

In the following, we show that the function f is submodular. Consider two sets D, D' such that $D \subseteq D'$. Then, for each $(a^*, s^*) \in G \setminus D'$ we need to show show that

$$f(D \cup \{(a^*, s^*)\}) - f(D) \geq f(D' \cup \{(a^*, s^*)\}) - f(D').$$

To do so, it is sufficient to prove this for all the functions $f_{\bar{s},\bar{a}}(\cdot)$, $a \in A$, $\bar{s} \in S : \rho(\bar{s}) = \emptyset$. If $\bar{a} \neq a^*$ or $s^* \notin \sigma^{\bar{s}}$ it trivially holds since the value of the function $f_{\bar{s},\bar{a}}(\cdot)$ does *not* depend on the element (a^*, s^*). Otherwise, notice that $a^* = \bar{a}$ and

$$f_{\bar{s},a^*}(D) = \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} : (a^*,s) \in D} q_{a^*,s} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a^*,s') \in D} (1 - q_{a^*,s'}).$$

Then,

$$\begin{aligned}
&f_{\bar{s},a^*}(D \cup (a^*, s^*)) - f_{\bar{s},a^*}(D) \\
&= \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} : (a^*,s) \in D \cup \{(a^*,s^*)\}} q_{s,a^*} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (s',a^*) \in D \cup \{(a^*,s^*)\}} (1 - q_{a^*,s'}) - \\
&- \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} : (a^*,s) \in D} q_{a^*,s} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a^*,s') \in D} (1 - q_{a^*,s'}) = \\
&= \sum_{s \in \boldsymbol{\sigma}^{s^*} \setminus \{s^*\} : (a^*,s) \in D \cup \{(a^*,s^*)\}} q_{a^*,s} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a^*,s') \in D} (1 - q_{a^*,s'}) + \\
&+ q_{a^*,s^*} \prod_{s \in \boldsymbol{\sigma}^{s^*} \setminus \{s^*\} : (a^*,s) \in D} (1 - q_{a^*,s}) + \\
&+ \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} \setminus \boldsymbol{\sigma}^{s^*} : (a^*,s) \in D \cup \{(a^*,s^*)\}} q_{a^*,s}(1 - q_{a^*,s^*}) \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a^*,s') \in D} (1 - q_{a^*,s'}) - \\
&- \sum_{s \in \boldsymbol{\sigma}^{s^*} \setminus \{s^*\} : (a^*,s) \in D} q_{a^*,s} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a^*,s') \in D} (1 - q_{a^*,s'}) - \\
&- \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} \setminus \boldsymbol{\sigma}^{s^*} : (a^*,s) \in D} q_{a^*,s} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a^*,s') \in D} (1 - q_{a^*,s'}) \\
&= q_{a^*,s^*} \prod_{s \in \boldsymbol{\sigma}^{s^*} \setminus \{s^*\} : (a^*,s) \in D} (1 - q_{a^*,s}) + \\
&+ \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} \setminus \boldsymbol{\sigma}^{s^*} : (a^*,s) \in D \cup \{(a^*,s^*)\}} q_{a^*,s}(1 - q_{a^*,s^*}) \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a^*,s') \in D} (1 - q_{a^*,s'}) - \\
&- \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} \setminus \boldsymbol{\sigma}^{s^*} : (a^*,s) \in D} q_{a^*,s} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a^*,s') \in D} (1 - q_{a^*,s'}) = \\
&= q_{a^*,s^*} \prod_{s \in \boldsymbol{\sigma}^{s^*} \setminus \{s^*\} : (a^*,s) \in D} (1 - q_{a^*,s}) - \\
&- q_{a^*,s^*} \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} \setminus \boldsymbol{\sigma}^{s^*} : (a^*,s) \in D} q_{a^*,s} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \{s\} : (a^*,s') \in D} (1 - q_{a^*,s'}) = \\
&= q_{a^*,s^*} \left[\prod_{s \in \boldsymbol{\sigma}^{s^*} \setminus \{s^*\} : (a^*,s) \in D} (1 - q_{a^*,s}) \right] \cdot \\
&\cdot \left[1 - \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} \setminus \boldsymbol{\sigma}^{s^*} : (a^*,s) \in D} q_{a^*,s} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \boldsymbol{\sigma}^{s^*} : (a^*,s') \in D} (1 - q_{a^*,s'}) \right] \geq \\
&\geq q_{a^*,s^*} \left[\prod_{s \in \boldsymbol{\sigma}^{s^*} \setminus \{s^*\} : (a^*,s) \in D'} (1 - q_{a^*,s}) \right] \cdot \\
&\cdot \left[1 - \sum_{s \in \boldsymbol{\sigma}^{\bar{s}} \setminus \boldsymbol{\sigma}^{s^*} : (a^*,s) \in D} q_{a^*,s} \prod_{s' \in \boldsymbol{\sigma}^s \setminus \boldsymbol{\sigma}^{s^*} : (a^*,s') \in D} (1 - q_{a^*,s'}) \right] \geq
\end{aligned}$$

$$\geq q_{a^*,s^*}\left[\prod_{s\in\boldsymbol{\sigma}^{s^*}\setminus\{s^*\}:(a^*,s)\in D'}(1-q_{a^*,s})\right]\cdot$$

$$\cdot\left[1-\sum_{s\in\boldsymbol{\sigma}^{\bar{s}}\setminus\boldsymbol{\sigma}^{s^*}:(a^*,s)\in D'}q_{a^*,s}\prod_{s'\in\boldsymbol{\sigma}^{s}\setminus\boldsymbol{\sigma}^{s^*}:(a^*,s')\in D'}(1-q_{a^*,s'})\right]=$$

$$=\sum_{s\in\boldsymbol{\sigma}^{\bar{s}}:(a^*,s)\in D'\cup\{(a^*,s^*)\}}q_{s,a^*}\prod_{s'\in\boldsymbol{\sigma}^{s}\setminus\{s\}:(s',a^*)\in D'\cup\{(a^*,s^*)\}}(1-q_{a^*,s'})-$$

$$-\sum_{s\in\boldsymbol{\sigma}^{\bar{s}}:(a^*,s)\in D'}q_{a^*,s}\prod_{s'\in\boldsymbol{\sigma}^{s}\setminus\{s\}:(a^*,s')\in D'}(1-q_{a^*,s'})=$$

$$=f_{\bar{s},a^*}(D'\cup(a^*,s^*))-f_{\bar{s},a^*}(D')$$

where the first inequality holds since we are adding elements smaller than 1 to the product. The second inequality comes from the following probabilistic analysis:

$$\sum_{s\in\boldsymbol{\sigma}^{\bar{s}}\setminus\boldsymbol{\sigma}^{s^*}:(a^*,s)\in D}q_{a^*,s}\prod_{s'\in\boldsymbol{\sigma}^{s}\setminus\boldsymbol{\sigma}^{s^*}:(a^*,s')\in D}(1-q_{a^*,s'})$$

is the probability that at least one element of the set $\{s\in\boldsymbol{\sigma}^{\bar{s}}\setminus\boldsymbol{\sigma}^{s^*}:(a^*,s)\in D\}$ is converted when each scene s converts with probability $q_{a^*,s}$. Hence, adding more scenes in which a^* can be converted, *i.e.* replacing D with D', increases the probability of conversion. Finally, the second-to-last equality comes from steps similar to the one in the second, third, fourth, and fifth equalities in reverse order. This concludes the proof. □

Theorem 7.6. *For any $\epsilon>0$, it is NP-Hard to approximate* `META-SI-E` *to within a factor $|k+1|^{1-\epsilon}$, where k is the memory length.*

Proof. Given a graph $G=(V,E)$, we build an instance of META-SI-E such that there exists an ad a_v for each $v\in V$. The tree of scenes is composed by a line with a scene s_v for each $v\in V$ in an arbitrary order. All the transition probabilities $\pi_{s,s'}$ are set to 1, all the qualities $q_{a,s}$ are set to 1, and all the values per conversion are θ_a are set to 1. Finally, the externalities $\gamma_{a_v,a_{v'}}=1$ if $(v,v')\in E$ and 0 otherwise, while the memory length $k=|V|-1$. We show that if there exists a clique of size $|V|^{1-\epsilon}$, then there exists an ad allocation with value at least $|k+1|^{1-\epsilon}$, while if all the cliques have size at most $|V|^{\epsilon}$, then all the allocations have value at most $|k+1|^{\epsilon}$. Since ϵ can be arbitrary small, this concludes the proof.

soundness. Suppose that there exists a clique V^* of size $|V|^{1-\epsilon}$. Consider the allocation x^* in which each ad in V^* is allocated to one of the scene (in an arbitrary order), while the other ads are not allocated. It is easy to see that since there are $|V|^{1-\epsilon}$ different ads allocated and there are no there are no externalities among the allocated ads, *i.e.*, $\gamma_{a_v,a_{v'}}=1$ for all $v,v'\in V^*$, then the allocation expected value is $|V^*|\geq|V|^{1-\epsilon}=|k+1|^{1-\epsilon}$.

completeness. Suppose by contradiction that all the cliques have size at most $|V|^{\epsilon}$ and that there exists an allocation x with value strictly larger than $|k+1|^{\epsilon}$. First, notice that the expected value provided by an ad $a\in A$ is at most 1. This holds since $\theta_a=1$ and an ad can be converted ad most one time. Let $\bar{A}$ be the set of allocated ads that do not suffer negative externalities and hence provide a positive utility, *i.e.*, the set of ads a assigned to a scene s such that $\gamma_{x(s'),a}=1$ for each $s'\in\boldsymbol{\sigma}^s$. Recall that externalities strictly smaller than 1 set the probability of conversion to 0 by construction. Hence all the ads not in $\bar{A}$ provide 0 utility. Then, the set $\bar{A}$ does not include two ads a_v, $a_{v'}$ such that $(v,v')\notin E$. Otherwise, we have that the first visualized ad has negative externalities on the second, contradiction the definition of $\bar{A}$. Let $\bar{V}$ be the node of the graph relative to the ad in $\bar{A}$, *i.e.*, $v\in\bar{V}$ if and only if $a_v\in\bar{A}$. $\bar{V}$ is such that $(v,v')\in E$ for each $v,v'\in\bar{V}$ and hence $\bar{V}$ is a clique. Since the expected value of the allocation is $|\bar{A}|=|\bar{V}|\leq|V|^{\epsilon}=|k+1|^{\epsilon}$ we reach a contradiction, □

APPENDIX C

Part III

C.1 Chapter 8

C.1.1 Omitted Proofs

Theorem 8.1 (Inapproximability)**.** *For any $\rho \in (0, 1]$, there is no polynomial-time algorithm returning a ρ-approximation to the problem in Equations* (8.1a)–(8.1c)*, unless* $\mathsf{P} = \mathsf{NP}$.

Proof. We restrict to the instances of SUBSET-SUM such that $z \leq \sum_{i \in S} u_i$. Solving these instances is trivially NP-hard, as any instance with $z > \sum_{i \in S} u_i$ is not satisfiable, and we can decide it in polynomial time. Given an instance of SUBSET-SUM, let $\ell = \frac{\sum_{i \in S} u_i + 1}{\rho}$. Let us notice that, the lower the degree of approximation we aim, the larger the value of ℓ. For instance, when study the problem of computing an exact solution, we set $\rho = 1$ and therefore $\ell = \sum_{i \in S} u_i + 1$, whereas, when we require a $1/2$-approximation, we set $\rho = 1/2$ and therefore $\ell = 2(\sum_{i \in S} u_i + 1)$. We have $|S| + 1$ subcampaigns, each denoted with C_j. The available bid values belong to $\{0, 1\}$ for every subcampaign C_j. The parameters of the subcampaigns are set as follows.

- Subcampaign C_0: we set $v_0 = 1$, and

$$c_0(x) = \begin{cases} 2\ell + z & \text{if } x = 1 \\ 0 & \text{otherwise} \end{cases}, \qquad n_0(x) = \begin{cases} \ell & \text{if } x = 1 \\ 0 & \text{otherwise} \end{cases}.$$

- Subcampaign C_j for every $j \in S$: we set $v_j = 1$, and

$$c_j(x) = \begin{cases} u_i & \text{if } x = 1 \\ 0 & \text{otherwise} \end{cases}, \qquad n_j(x) = \begin{cases} u_i & \text{if } x = 1 \\ 0 & \text{otherwise} \end{cases}.$$

We set the daily budget $\beta = 2(z + \ell)$ and the ROI limit $\Lambda = \frac{1}{2}$.[1]

We show that, if a SUBSET-SUM instance is satisfiable, then the corresponding instance of our problem admits a solution with a revenue larger than ℓ, while, if a SUBSET-SUM instance is not satisfiable, the maximum revenue in the corresponding instance of our problem is at most $\rho\,\ell - 1$. Thus, the application of any polynomial-time ρ-approximation algorithm to instances of our problem generated from instances of SUBSET-SUM as described above would return a solution whose value is not smaller than $\rho\,\ell$ when the SUBSET-SUM instance is satisfiable, and it is not larger than $\rho\,\ell - 1$ when the SUBSET-SUM instance is not satisfiable. As a result, whenever such an algorithm returns a solution with a value that is not smaller than $\rho\,\ell$, we can decide that the corresponding SUBSET-SUM instance is satisfiable. Analogously, whenever such an algorithm returns a solution with a value that is in the range $[\rho(\rho\,\ell - 1), \rho\,\ell - 1]$, we can decide that the corresponding SUBSET-SUM instance is not satisfiable. Let us notice that the range $[\rho(\rho\,\ell - 1), \rho\,\ell - 1]$ is well defined for every $\rho \in (0, 1]$, as, by construction, $\rho\,\ell = \sum_{i \in S} u_i + 1 \geq 1$ and therefore $\rho\,\ell - 1 \geq \rho(\rho\,\ell - 1)$. Hence, such an algorithm would decide in polynomial time whether or not a SUBSET-SUM instance is satisfiable, but this is not possible unless $\mathsf{P} = \mathsf{NP}$. Since this holds for every $\rho \in (0, 1]$, then no ρ-approximation to our problem is allowed in polynomial time unless $\mathsf{P} = \mathsf{NP}$.

If. Suppose that SUBSET-SUM is satisfied by the set $S^* \subseteq S$ and that itss solution assigns $x_i = 1$ if $i \in S^*$ and $x_i = 0$ otherwise, and it assigns $x_0 = 1$. The total revenue is $\ell + z \geq \ell$ and the constraints are satisfied. In particular, the sum of the costs is $2\ell + z + z = 2(\ell + z)$, while ROI $= \frac{\ell + z}{2\ell + 2z} = \frac{1}{2}$.

Only if. Assume by contradiction that the instance of our problem admits a solution with a revenue strictly larger than $\rho\,\ell - 1$ and that SUBSET-SUM is not satisfiable. Then, it is easy to see that we need $x_0 = 1$ for campaign C_0 as the maximum achievable revenue is $\sum_{i \in S} u_i = \rho\,\ell - 1$ when $x_0 = 0$. Thus, since $x_0 = 1$, the budget constraint forces $\sum_{i \in S : x_i = 1} c_i(x_i) \leq z$, thus implying $\sum_{i \in S : x_i = 1} u_i \leq z$. By the satisfaction of the ROI constraint, *i.e.*, $\frac{\sum_{i \in S : x_i = 1} u_i + l}{\sum_{i \in S : x_i = 1} u_i + 2l + z} \geq \frac{1}{2}$, it must hold $\sum_{i \in S : x_i = 1} u_i \geq z$. Therefore, the set $S^* = \{i \in S : x_i = 1\}$ is a solution to SUBSET-SUM, thus reaching a contradiction. This concludes the proof. □

Theorem 8.4 (GCB pesudo-regret)**.** *Given $\delta \in (0,\ 1)$, GCB applied to the problem in Equations* (8.1a)–(8.1c)*, with probability at least $1 - \delta$, suffers from a pseudo-regret of:*

$$R^T(\mathit{GCB}) \leq \sqrt{\frac{8 T v_{\max}^2 N^3 b_T}{\ln(1 + \sigma^2)} \sum_{j=1}^{N} \gamma_{j,T}}\ ,$$

where $b_t := 2 \ln\left(\frac{\pi^2 N Q T t^2}{3\delta}\right)$ is an uncertainty term used to guarantee the confidence level required by GCB, $v_{\max} := \max_{j \in \{1,\dots,N\}} v_j$ is the maximum value per click over all subcampaigns, and $Q := \max_{j \in \{1,\dots,N\}} |X_j|$ is the maximum number of bids in a subcampaign.

[1] For the sake of clarity, the proof uses simple instances. The adoption of these instances is crucial to identify the most basic settings in which the problem is hard, and it is customarily done in the literature. Let us notice that it is possible to prove the theorem using more realistic instances. For example, we can build a reduction in which the costs are smaller than the values, *i.e.*, $c_i(x) < n_i(x) v_i$. In particular, the reduction holds even if we set $c_0(1) = \epsilon(2l + z)$, $c_j(1) = \epsilon u_i$, $\beta = 2\epsilon(z + l)$, and $\Lambda = 1/(2\epsilon)$ for an arbitrary small ϵ.

Proof. This proof extends the proof provided by Accabi et al. (2018) to the case in which multiple independent GPs are present in the optimization problem.

Let us define $r_{\boldsymbol{\mu}}(\mathbf{x})$ as the expected reward provided by a specific allocation $\mathbf{x} = (x_1, \dots, x_N)$ under the assumption that the parameter vector of the optimization problem is $\boldsymbol{\mu}$. Moreover, let

$$\boldsymbol{\eta} := \big[w_1(x_1), \dots, w_N(x_{|X_N|}), w_1(x_1), \dots, w_N(x_{|X_N|}), -c_1(x_1), \dots, -c_N(x_{|X_N|})\big],$$

be the vector characterizing the optimization problem in Equations (8.1a)-(8.1c), $\mathbf{x}_t$ be the allocation chosen by the GCB algorithm at round t, $\mathbf{x}^*_{\boldsymbol{\eta}}$ the optimal allocation—*i.e.*, the one solving the discrete version of the optimization problem in Equations (8.1a)-(8.1c) with parameter $\boldsymbol{\eta}$—, and $r^*_{\boldsymbol{\eta}}$ the corresponding expected reward.

To guarantee that GCB provides a sublinear pseudo-regret, we need that a few assumptions are satisfied. More specifically, we need a *monotonicity property*, stating that the value of the objective function increases as the values of the elements in $\boldsymbol{\mu}$ increase and a *Lipschitz continuity* assumption between the parameter vector $\boldsymbol{\mu}$ and the value returned by the objective function in Equation (8.1a). Formally:

Assumption C.1 (Monotonicity)**.** *The expected reward $r_{\boldsymbol{\mu}}(S) := \sum_{j=1}^{N} v_j\, n_j(x_{j,t})$, where S is the bid allocation, is monotonically non decreasing in $\boldsymbol{\mu}$, i.e., given $\boldsymbol{\mu}$, $\boldsymbol{\eta}$ s.t. $\mu_i \leq \eta_i$ for each i, we have $r_{\boldsymbol{\mu}}(S) \leq r_{\boldsymbol{\eta}}(S)$ for each S.*

Assumption C.2 (Lipschitz continuity)**.** *The expected reward $r_{\boldsymbol{\mu}}(S)$ is Lipschitz continuous in the infinite norm w.r.t. the expected payoff vector $\boldsymbol{\mu}$, with Lipschitz constant $L > 0$. Formally, for each $\boldsymbol{\mu}, \boldsymbol{\eta}$ we have $|r_{\boldsymbol{\mu}}(S) - r_{\boldsymbol{\eta}}(S)| \leq L||\boldsymbol{\mu} - \boldsymbol{\eta}||_\infty$, where the infinite norm of a payoff vector is $||\boldsymbol{\mu}||_\infty := \max_i |\mu_i|$.*

Trivially, we have that the Lipschitz continuity holds with constant $L = N$ (number of subcampaigns). Instead, the monotonicity property holds by definition of $\boldsymbol{\mu}$, as the increase of a value of $\overline{w}_j(x)$ would increase the value of the objective function, and the increase of the values of $\underline{w}_j(x)$ or $\overline{c}_j(x)$ would enlarge the feasibility region of the problem, thus not excluding optimal solutions.

Let us now focus on the per-step expected regret, defined as:

$$reg_t := r^*_{\boldsymbol{\eta}} - r_{\boldsymbol{\eta}}(\mathbf{x}_t).$$

Let us recall a property of the Gaussian distribution which will be useful in what follows. Be $r \sim \mathcal{N}(0,1)$ and $c \in \mathbb{R}^+$, we have:

$$\begin{aligned}\mathbb{P}[r > c] &= \frac{1}{\sqrt{2\pi}} e^{-\frac{c^2}{2}} \int_c^\infty e^{-\frac{(r-c)^2}{2} - c(r-c)}\, dr \\ &\leq e^{-\frac{c^2}{2}}\, \mathbb{P}[r > 0] = \frac{1}{2} e^{-\frac{c^2}{2}},\end{aligned}$$

since $e^{-c(r-c)} \leq 1$ for $r \geq c$. For the symmetry of the Gaussian distribution, we have:

$$\mathbb{P}[|r| > c] \leq e^{-\frac{c^2}{2}}. \tag{C.1}$$

Let us focus on the GP modeling the number of clicks. Given a generic sequence of elements $(x_{j,1}, \dots, x_{j,1})$, with $x_{j,h} \in X_j$, and the corresponding sequence of number of clicks $(\tilde{n}_{j,1}(x_{j,1}), \dots, \tilde{n}_{j,t}(x_{j,t}))$, we have that:

$$n_{j,t}(x) \sim \mathcal{N}(\hat{n}_{j,t}(x), (\hat{\sigma}^n_{j,t}(x))^2),$$

for all $x \in X_j$. Thus, substituting $r = \frac{\hat{n}_{j,t}(x) - n_{j,t}(x)}{\hat{\sigma}^n_{j,t}(x)}$ and $c = \sqrt{b_t}$ in Equation (C.1), we obtain:

$$\mathbb{P}\left[|\hat{n}_{j,t}(x) - n_{j,t}(x)| > \sqrt{b_t}\hat{\sigma}^n_{j,t}(x)\right] \leq e^{-\frac{b_t}{2}}. \tag{C.2}$$

Appendix C. Part III

Recall that, after n rounds, each arm can be chosen a number of times from 1 to n. Applying the union bound over the rounds ($h \in \{1, \dots, T\}$), the sub-campaigns C_j (C_j with $j \in \{1, \dots, N\}$), the number of times the arms in C_j are chosen ($t \in \{1, \dots, n\}$), and the available arms in C_j ($x \in X_j$), and exploiting Equation (C.2), we obtain:

$$\mathbb{P}\left[\bigcup_{h,j,t,x}\Big(|\hat{n}_{j,t}(x) - n_{j,t}(x)| > \sqrt{b_t}\hat{\sigma}^n_{j,t}(x)\Big)\right] \tag{C.3}$$

$$\leq \sum_{h=1}^{T}\sum_{j=1}^{N}\sum_{t=1}^{n}|X_j|e^{-\frac{b_t}{2}}. \tag{C.4}$$

Thus, choosing $b_t = 2\ln\left(\frac{\pi^2 NQTt^2}{3\delta}\right)$, we obtain:

$$\sum_{h=1}^{T}\sum_{j=1}^{N}\sum_{t=1}^{n}|X_j|e^{-\frac{b_t}{2}} \leq \sum_{h=1}^{T}\sum_{j=1}^{N}\sum_{t=1}^{n}Q\frac{3\delta}{\pi^2 NQTt^2}$$

$$\sum_{n=1}^{\infty}\frac{2\delta}{\pi^2 t^2} = \frac{\delta}{2},$$

where we used the fact that $Q \geq |X_j|$ for each $j \in \{1, \dots N\}$.

Using the same proof on the GP defined over the costs leads to:

$$\mathbb{P}\left[\bigcup_{h,j,t,x}\Big(|\hat{c}_{j,t}(x) - \hat{c}_{j,t}(x)| > \sqrt{b_t}\hat{\sigma}^c_{j,t}(x)\Big)\right] \leq \frac{\delta}{2}.$$

The above proof implies that the union of the event that all the bounds used in the GCB algorithm holds with probability at least $1-\delta$. Formally, for each $t \geq 1$, we know that with probability at least $1-\delta$ the following holds for all $x_j \in X_j$, $j \in \{1, \dots N\}$, and number of times the the arm x_j has been pulled over t rounds:

$$|\hat{n}_j(x_j) - n_j(x_j)| \leq \sqrt{b_t}\hat{\sigma}^n_{j,t}(x_j), \tag{C.5}$$

$$|\hat{c}_j(x_j) - c_j(x_j)| \leq \sqrt{b_t}\hat{\sigma}^c_{j,t}(x_j). \tag{C.6}$$

From now on, let us assume we are in the *clean event* that the previous bounds hold.

Let us focus on the term $r_{\boldsymbol{\mu}}(\mathbf{x}_t)$. The following holds:

$$r_{\boldsymbol{\mu}}(\mathbf{x}_t) \geq r^*_{\boldsymbol{\mu}} \geq r_{\boldsymbol{\mu}}(\mathbf{x}^*_{\boldsymbol{\mu}}) \geq r_{\boldsymbol{\eta}}(\mathbf{x}^*_{\boldsymbol{\mu}}) = r^*_{\boldsymbol{\eta}}, \tag{C.7}$$

where we use the definition of $r^*_{\boldsymbol{\mu}}$, and the monotonicity property of the expected reward (Assumption C.1), being $(\boldsymbol{\mu})_i \geq (\boldsymbol{\eta})_i, \forall i$. Using Equation (C.7), the instantaneous expected pseudo-regret reg_t at round t satisfies the following inequality:

$$reg_t = r^*_{\boldsymbol{\eta}} - r_{\boldsymbol{\eta}}(\mathbf{x}_t) \leq r_{\boldsymbol{\mu}}(\mathbf{x}_t) - r_{\boldsymbol{\eta}}(\mathbf{x}_t) = \tag{C.8}$$

$$\leq \underbrace{r_{\boldsymbol{\mu}}(\mathbf{x}_t) - r_{\hat{\boldsymbol{\mu}}}(\mathbf{x}_t)}_{r_a} + \underbrace{r_{\hat{\boldsymbol{\mu}}}(\mathbf{x}_t) - r_{\boldsymbol{\eta}}(\mathbf{x}_t)}_{r_b}, \tag{C.9}$$

where

$$\hat{\boldsymbol{\mu}} := \big[\hat{w}_{1,t-1}(x_1), \dots, \hat{w}_{N,t-1}(x_{|X_N|}), \hat{w}_{1,t-1}(x_1), \dots, \hat{w}_{N,t-1}(x_{|X_N|}), \tag{C.10}$$

$$- \hat{c}_{1,t-1}(x_1), \dots, -\hat{c}_{N,t-1}(x_{|X_N|})\big],$$

is the vector composed of the estimated average payoffs for each arm $x \in X_j$ and each campaign C_j, where $\hat{w}_{j,t-1}(x) := v_j \hat{n}_{j,t-1}(x)$.

We use the Lipschitz property of the expected reward function (see Assumption C.2) to bound the terms in Equation (C.9) as follows:

$$r_a \leq L||\boldsymbol{\mu} - \hat{\boldsymbol{\mu}}||_\infty = L \max_{j \in \{1,\dots,N\}} \left(v_{\max} \sqrt{b_t} \max_{x \in X_j} \hat{\sigma}^n_{j,t}(x) \right) \tag{C.11}$$

$$\leq N v_{\max} \sqrt{b_t} \max_{j \in \{1,\dots,N\}} \left(\max_{x \in X_j} \hat{\sigma}^n_{j,t}(x) \right) \tag{C.12}$$

$$\leq N v_{\max} \sqrt{b_t} \sum_{j=1}^{N} \left(\max_{x \in X_j} \hat{\sigma}^n_{j,t}(x) \right), \tag{C.13}$$

$$r_b \leq L||\hat{\boldsymbol{\mu}} - \boldsymbol{\eta}||_\infty$$

$$\leq N v_{\max} \sqrt{b_t} \sum_{j=1}^{N} \left(\max_{x \in X_j} \hat{\sigma}^n_{j,t}(x) \right), \tag{C.14}$$

where Equation (C.11) holds by the definition of $\boldsymbol{\mu}$, Equation (C.13) holds since the maximum over a set is not greater than the sum of the elements of the set, if they are all non-negative, and Equation (C.14) directly follows from Equation (C.5). Plugging Equations (C.13) and (C.14) into Equation (C.9), we obtain:

$$reg_t \leq 2N v_{\max} \sqrt{b_t} \sum_{j=1}^{N} \left(\max_{x \in X_j} \hat{\sigma}^n_{j,t}(x) \right). \tag{C.15}$$

We need now to upper bound $\hat{\sigma}^n_{j,t}(x)$. Consider a realization $n_j(\cdot)$ of a GP over X_j and recall that, thanks to Lemma 5.3 in (Srinivas et al., 2010), under the Gaussian assumption we can express the information gain $IG_{j,t}$ provided by $(\tilde{n}_j(\hat{x}_{j,1}), \dots, \tilde{n}_j(\hat{x}_{j,|X_j|}))$ corresponding to the sequence of arms $(\hat{x}_{j,1}, \dots, \hat{x}_{j,|X_j|})$ as:

$$IG_{j,t} = \frac{1}{2} \sum_{h=1}^{t} \log\left(1 + \sigma^{-2} \left(\hat{\sigma}^n_{j,t}(\hat{x}_{j,h})\right)^2\right). \tag{C.16}$$

We have that:

$$\left(\hat{\sigma}^n_{j,t}(\hat{x}_{j,h})\right)^2 = \sigma^2 \left[\sigma^{-2} \left(\hat{\sigma}^n_{j,t}(\hat{x}_{j,h})\right)^2\right] \leq \frac{\log\left[1 + \sigma^{-2} \left(\hat{\sigma}^n_{j,t}(\hat{x}_{j,h})\right)^2\right]}{\log\left(1 + \sigma^{-2}\right)}, \tag{C.17}$$

since $s^2 \leq \frac{\sigma^{-2} \log(1+s^2)}{\log\left(1+\sigma^{-2}\right)}$ for all $s \in [0, \sigma^{-1}]$, and $\sigma^{-2} \left(\hat{\sigma}^n_{j,t}(\hat{x}_{j,h})\right)^2 \leq \sigma^{-2}\, k(\hat{x}_{j,h}, \hat{x}_{j,h}) \leq \sigma^{-2}$, where $k(\cdot,\cdot)$ is the kernel of the GP. Since Equation (C.17) holds for any $x \in X_j$ and for any $j \in \{1, \dots N\}$, then it also holds for the arm $\hat{x}_{\max}$ maximizing the variance $\left(\hat{\sigma}^n_{j,t}(\hat{x}_{j,h})\right)^2$ over X_j. Thus, setting $\bar{c} = \frac{8\, N^2}{\log\left(1+\sigma^{-2}\right)}$ and exploiting the Cauchy-Schwarz inequality, we obtain:

$$\left(R^T\right)^2 (GCB) \leq T \sum_{t=1}^{T} reg_t^2$$

$$\leq T \sum_{t=1}^{T} 4N^2 v_{\max}^2 b_t \left[\sum_{j=1}^{N} \left(\max_{x \in X_j} \hat{\sigma}^n_{j,t}(x) \right)\right]^2$$

$$\leq 4N^2 v_{\max}^2 T b_T \sum_{t=1}^{T} \left[N \sum_{j=1}^{N} \max_{x \in X_j} \left(\hat{\sigma}^n_{j,t}(x)\right)^2 \right]$$

$$\leq \bar{c} N v_{\max}^2 T b_T \sum_{j=1}^{N} \frac{1}{2} \sum_{t=1}^{T} \max_{x \in X_j} \log\left(1 + \sigma^{-2} \left(\hat{\sigma}_{j,t}^n(\hat{x}_{j,h})\right)^2\right)$$

$$\leq \bar{c} N v_{\max}^2 T b_T \sum_{j=1}^{N} \gamma_{j,T}.$$

We conclude the proof by taking the square root on both the r.h.s. and the l.h.s. of the last inequality. □

Theorem 8.8 ($\mathsf{GCB}_{\mathtt{safe}}(\psi, 0)$ pseudo-regret and safety with tolerance)**.** *When:*

$$\psi \geq 2 \frac{\beta_{opt} + n_{\max}}{\beta_{opt}^2} \sum_{j=1}^{N} v_j \sqrt{2 \ln\left(\frac{\pi^2 N Q T^3}{3\delta'}\right)} \sigma$$

and

$$\beta_{opt} < \beta \frac{\sum_{j=1}^{N} v_j}{\frac{N\ \beta_{opt} \psi}{\beta_{opt} + n_{\max}} + \sum_{j=1}^{N} v_j},$$

where $\delta' \leq \delta$, β_{opt} is the spend at the optimal solution of the original problem, and $n_{\max} := \max_{j,x} n_j(x)$ is the maximum over the sub-campaigns and the admissible bids of the expected number of clicks, $\mathsf{GCB}_{\mathtt{safe}}(\psi, 0)$ provides a pseudo-regret w.r.t. the optimal solution to the original problem of $\mathcal{O}\left(\sqrt{T \sum_{j=1}^{N} \gamma_{j,T}}\right)$ with probability at least $1 - \delta - \frac{\delta'}{QT^2}$, while being δ-safe w.r.t. the constraints of the auxiliary problem.

Proof. In what follows, we show that, at a specific day t, since the optimal solution of the original problem $\left\{x_j^*\right\}_{j=1}^{N}$ is included in the set of feasible ones, we are in a setting analogous to the one of GCB, in which the regret is sublinear. Let us assume that the upper bounds on all the quantities (number of clicks and costs) holds. This has been shown before to occur with overall probability δ over the whole time horizon T. Moreover, notice that combining the properties of the budget of the optimal solution of the original problem β_{opt} and using $\psi = 2 \frac{\beta_{opt} + n_{\max}}{\beta_{opt}^2} \sum_{j=1}^{N} v_j \sqrt{2 \ln\left(\frac{\pi^2 N Q T^3}{3\delta'}\right)} \sigma$, we have:

$$\beta_{opt} < \beta \frac{\sum_{j=1}^{N} v_j}{\frac{N\ \beta_{opt} \psi}{\beta_{opt} + n_{\max}} + \sum_{j=1}^{N} v_j} \tag{C.18}$$

$$\left(\frac{N\ \beta_{opt} \psi}{\beta_{opt} + n_{\max}} + \sum_{j=1}^{N} v_j\right) \beta_{opt} < \beta \sum_{j=1}^{N} v_j \tag{C.19}$$

$$2N \sum_{j=1}^{N} v_j \sqrt{2 \ln\left(\frac{\pi^2 N Q T^3}{3\delta'}\right)} \sigma + \sum_{j=1}^{N} v_j \beta_{opt} < \beta \sum_{j=1}^{N} v_j \tag{C.20}$$

$$\beta > \beta_{opt} + 2N \sqrt{2 \ln\left(\frac{\pi^2 N Q T^3}{3\delta'}\right)} \sigma. \tag{C.21}$$

First, let us evaluate the probability that the optimal solution is not feasible. This occurs if its bounds are either violating the ROI or budget constraints. First, we show that analysing the budget constraint, the optimal solution of the original problem is feasible with high probability. Formally, it is not feasible with probability:

$$\mathbb{P}\left(\sum_{j=1}^{N} \overline{c}_j(x_j^*) > \beta\right) \leq \mathbb{P}\left(\sum_{j=1}^{N} \overline{c}_j(x_j^*) > \beta_{opt} + 2N \sqrt{2 \ln\left(\frac{\pi^2 N Q T^3}{3\delta'}\right)} \sigma\right) \tag{C.22}$$

$$= \mathbb{P}\left(\sum_{j=1}^{N} \overline{c}_j(x_j^*) > \sum_{j=1}^{N} c_j(x_j^*) + 2N\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\right) \tag{C.23}$$

$$\leq \sum_{j=1}^{N} \mathbb{P}\left(\overline{c}_j(x_j^*) > c_j(x_j^*) + 2\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\right) \tag{C.24}$$

$$= \sum_{j=1}^{N} \mathbb{P}\left(\hat{c}_{j,t-1}(x_j^*) - c_j(x_j^*) > -\sqrt{b_t}\hat{\sigma}^c_{j,t-1}(x_j^*) + 2\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\right) \tag{C.25}$$

$$\leq \sum_{j=1}^{N} \mathbb{P}\left(\hat{c}_{j,t-1}(x_j^*) - c_j(x_j^*) > \sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\hat{\sigma}^c_{j,t-1}(x_j^*)\right) \tag{C.26}$$

$$\leq \sum_{j=1}^{N} \mathbb{P}\left(\frac{\hat{c}_{j,t-1}(x_j^*) - c_j(x_j^*)}{\hat{\sigma}^c_{j,t-1}(x_j^*)} > \sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\right) \tag{C.27}$$

$$\leq \sum_{j=1}^{N} \frac{3\delta'}{\pi^2 NQT^3} = \frac{3\delta'}{\pi^2 QT^3}, \tag{C.28}$$

where, in the inequality in Equation (C.22) we used Equation (C.21), in Equation (C.27) we used the fact that $\frac{\pi^2 NQt^2 T}{3\delta} \leq \frac{\pi^2 NQT^3}{3\delta'}$ for each $t \in \{1, \ldots, T\}$, $\hat{\sigma}^c_{j,t-1}(x_j^*) \leq \sigma$ for each j and t, and the inequality in Equation (C.28) is from Srinivas et al. (2010). Summing over the time horizon T, we get that the optimal solution of the original problem $\left\{x_j^*\right\}_{j=1}^N$ is excluded from the set of the feasible ones with probability at most $\frac{3\delta'}{\pi^2 QT^2}$.

Second, we derive a bound over the probability that the optimal solution of the original problem is feasible due to the newly defined ROI constraint. Let us notice that since the ROI constraint is active we have $\Lambda = \Lambda_{opt}$. The probability that $\left\{x_j^*\right\}_{j=1}^N$ is not feasible due to the ROI constraint is:

$$\mathbb{P}\left(\frac{\sum_{j=1}^N v_j\ \underline{n}_j(x_j^*)}{\sum_{j=1}^N\ \overline{c}_j(x_j^*)} < \Lambda - \psi\right) \tag{C.29}$$

$$\leq \mathbb{P}\left(\frac{\sum_{j=1}^N v_j\ \underline{n}_j(x_j^*)}{\sum_{j=1}^N\ \overline{c}_j(x_j^*)} < \Lambda_{opt} - 2\frac{\beta_{opt} + n_{\max}}{\beta_{opt}^2}\sum_{j=1}^{N} v_j\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\right) \tag{C.30}$$

$$= \mathbb{P}\left(\frac{\sum_{j=1}^N v_j\ \underline{n}_j(x_j^*)}{\sum_{j=1}^N\ \overline{c}_j(x_j^*)} < \frac{\sum_{j=1}^N v_j\ n_j(x_j^*)}{\sum_{j=1}^N\ c_j(x_j^*)} - 2\frac{\beta_{opt} + n_{\max}}{\beta_{opt}^2}\sum_{j=1}^{N} v_j\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\right) \tag{C.31}$$

$$\begin{aligned} = \mathbb{P}\Bigg(&\sum_{j=1}^{N} c_j(x_j^*)\sum_{j=1}^{N} v_j\ \underline{n}_j(x_j^*) < \sum_{j=1}^{N}\ \overline{c}_j(x_j^*)\sum_{j=1}^{N} v_j\ n_j(x_j^*) \\ &-2\frac{\beta_{opt} + n_{\max}}{\beta_{opt}^2}\sum_{j=1}^{N} c_j(x_j^*)\sum_{j=1}^{N}\ \overline{c}_j(x_j^*)\sum_{j=1}^{N} v_j\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma\Bigg) \end{aligned} \tag{C.32}$$

$$\begin{aligned} = \mathbb{P}\Bigg(&\sum_{j=1}^{N} c_j(x_j^*)\sum_{j=1}^{N} v_j\ \underline{n}_j(x_j^*) - \sum_{j=1}^{N} c_j(x_j^*)\sum_{j=1}^{N} v_j\ n_j(x_j^*) + \\ &\frac{2}{\beta_{opt}}\sum_{j=1}^{N} c_j(x_j^*)\sum_{j=1}^{N}\ \overline{c}_j(x_j^*)\sum_{j=1}^{N} v_j\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma \end{aligned}$$

$$+\sum_{j=1}^{N} c_j(x_j^*)\sum_{j=1}^{N} v_j\ n_j(x_j^*) - \sum_{j=1}^{N} \overline{c}_j(x_j^*)\sum_{j=1}^{N} v_j\ n_j(x_j^*) +$$
$$\frac{2n_{\max}}{\beta_{opt}^2}\sum_{j=1}^{N} c_j(x_j^*)\sum_{j=1}^{N} \overline{c}_j(x_j^*)\sum_{j=1}^{N} v_j\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma < 0\Bigg) \tag{C.33}$$

$$\leq \mathbb{P}\left(\sum_{j=1}^{N} v_j\ \underline{n}_j(x_j^*) - \sum_{j=1}^{N} v_j\ n_j(x_j^*) + 2\underbrace{\frac{\sum_{j=1}^{N}\overline{c}_j(x_j^*)}{\beta_{opt}}}_{\geq 1}\sum_{j=1}^{N} v_j\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma < 0\right)$$
$$+\mathbb{P}\Bigg(\sum_{j=1}^{N} c_j(x_j^*)\sum_{j=1}^{N} v_j\ n_j(x_j^*) - \sum_{j=1}^{N}\overline{c}_j(x_j^*)\sum_{j=1}^{N} v_j\ n_j(x_j^*)$$
$$+2\underbrace{\frac{\sum_{j=1}^{N} c_j(x_j^*)\sum_{j=1}^{N}\overline{c}_j(x_j^*)}{\beta_{opt}^2}}_{\geq 1}\sum_{j=1}^{N} v_j\ \underbrace{n_{\max}}_{\geq n_j(x_j^*)}\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma < 0\Bigg) \tag{C.34}$$

$$\leq \sum_{j=1}^{N}\mathbb{P}\left(\underline{n}_j(x_j^*) - n_j(x_j^*) + 2\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma \leq 0\right)$$
$$+\sum_{j=1}^{N}\mathbb{P}\left(c_j(x_j^*) - \overline{c}_j(x_j^*) + 2\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma < 0\right) \tag{C.35}$$

$$\leq \sum_{j=1}^{N}\mathbb{P}\left(\hat{n}_{j,t-1}(x_j^*) - \sqrt{b_t}\hat{\sigma}_{j,t-1}^n(x_j^*) - n_j(x_j^*) + 2\underbrace{\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma}_{\geq\sqrt{b_t}\hat{\sigma}_{j,t-1}^n(x_j^*)} < 0\right)$$
$$+\sum_{j=1}^{N}\mathbb{P}\left(c_j(x_j^*) - \hat{c}_{j,t-1}(x_j^*) - \sqrt{b_t}\hat{\sigma}_{j,t-1}^c(x_j^*) + 2\underbrace{\sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\sigma}_{\geq\sqrt{b_t}\hat{\sigma}_{j,t-1}^c(x_j^*)} < 0\right) \tag{C.36}$$

$$\leq \sum_{j=1}^{N}\mathbb{P}\left(n_j(x_j^*) < \hat{n}_{j,t-1}(x_j^*) + \sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\hat{\sigma}_{j,t-1}^n(x_j^*)\right)$$
$$+\sum_{j=1}^{N}\mathbb{P}\left(c_j(x_j^*) < \hat{c}_{j,t-1}(x_j^*) - \sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\hat{\sigma}_{j,t-1}^c(x_j^*)\right) \tag{C.37}$$

$$= \sum_{j=1}^{N}\mathbb{P}\left(\frac{n_j(x_j^*) - \hat{n}_{j,t-1}(x_j^*)}{\hat{\sigma}_{j,t-1}^n(x_j^*)} > \sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\right)$$
$$+\sum_{j=1}^{N}\mathbb{P}\left(\frac{\hat{c}_{j,t-1}(x_j^*) - c_j(x_j^*)}{\hat{\sigma}_{j,t-1}^c(x_j^*)} > \sqrt{2\ln\frac{\pi^2 NQT^3}{3\delta'}}\right) \tag{C.38}$$

$$\leq 2\sum_{j=1}^{N}\frac{3\delta'}{\pi^2 NQT^3} = \frac{6\delta'}{\pi^2 QT^3}, \tag{C.39}$$

where in Equation (C.37) we used the fact that $\frac{\pi^2 NQt^2 T}{3\delta} \leq \frac{\pi^2 NQT^3}{3\delta'}$ for each $t \in \{1,\ldots,T\}$,

	C_1	C_2	C_3	C_4	C_5
θ_j	60	77	75	65	70
δ_j	0.41	0.48	0.43	0.47	0.40
α_j	497	565	573	503	536
γ_j	0.65	0.62	0.67	0.68	0.69
σ_f GP revenue	0.669	0.499	0.761	0.619	0.582
l GP revenue	0.425	0.469	0.471	0.483	0.386
σ_f GP cost	0.311	0.443	0.316	0.349	0.418
l GP cost	0.76	0.719	0.562	0.722	0.727

Figure C.1: *Parameters of the synthetic settings used in Experiment #1.*

$\hat{\sigma}^n_{j,t-1}(x^*_j) \leq \sigma$ for each j and t, and the inequality in Equation (C.39) is from Srinivas et al. (2010). Summing over the time horizon T ensures that the optimal solution of the original problem $\left\{x^*_j\right\}_{j=1}^N$ is excluded from the feasible solutions at most with probability $\frac{6\delta'}{\pi^2 QT^2}$. Finally, using a union bound, we have that the optimal solution can be chosen over the time horizon with probability at least $1 - \frac{3\delta'}{\pi^2 QT^2} - \frac{6\delta'}{\pi^2 QT^2} \leq 1 - \frac{\delta'}{QT^2}$.

Notice that here we want to compute the regret of the $\mathsf{GCB}_{\texttt{safe}}$ algorithm w.r.t. $\left\{x^*_j\right\}_{j=1}^N$ which is not optimal for the analysed relaxed problem. Nonetheless, the proof on the pseudo-regret provided in Theorem 8.4 is valid also for suboptimal solutions in the case it is feasible with high probability. This can be trivially shown using the fact that the regret w.r.t. a generic solution cannot be larger than the one computed w.r.t. the optimal one. Thanks to that, using a union bound over the probability that the bounds hold and that $\left\{x^*_j\right\}_{j=1}^N$ is feasible, we conclude that with probability at least $1 - \delta - \frac{\delta'}{QT^2}$ the regret $\mathsf{GCB}_{\texttt{safe}}$ is of the order of $\mathcal{O}\left(\sqrt{T\sum_{j=1}^N \gamma_{j,T}}\right)$. Finally, thanks to the property of the $\mathsf{GCB}_{\texttt{safe}}$ algorithm shown in Theorem 8.6, the learning policy is δ-safe for the relaxed problem. □

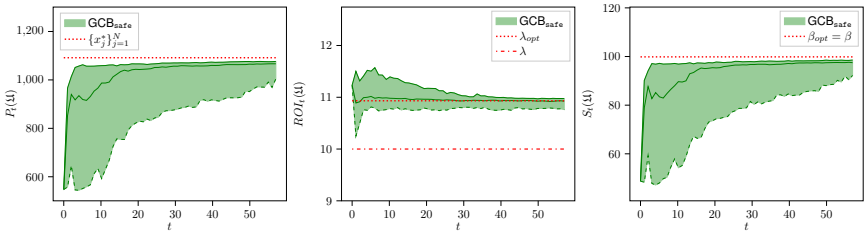
C.1.2 Additional Details on the Experiments

Parameters and Setting of Experiment #1

Table C.1 specifies the values of the parameters of cost and number-of-click functions of the subcampaigns used in Experiment #1.

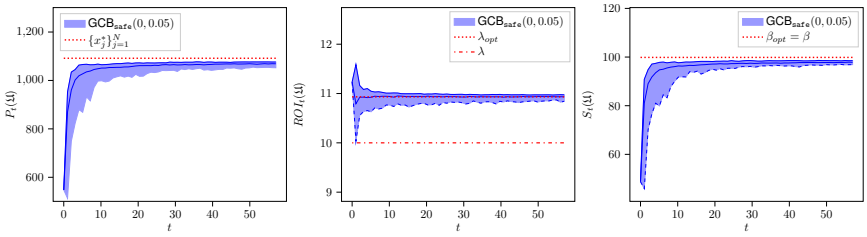
Additional Results of Experiment #2

In Figures C.2, C.3, C.4, and C.5 we report the 90% and 10% of the quantities related to Experiment #2 provided by the $\mathsf{GCB}_{\mathtt{safe}}$, $\mathsf{GCB}_{\mathtt{safe}}(0, 0.05)$, $\mathsf{GCB}_{\mathtt{safe}}(0, 0.10)$, and $\mathsf{GCB}_{\mathtt{safe}}(0, 0.15)$, respectively.



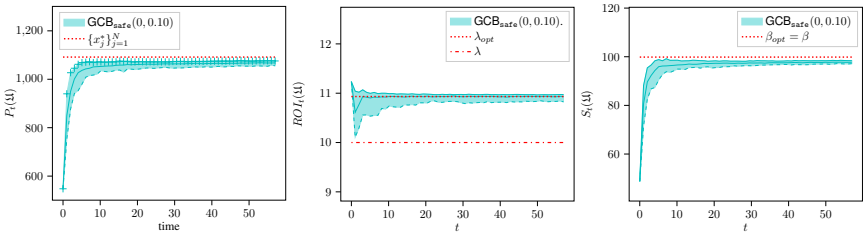
(a) **(b)** **(c)**

Figure C.2: *Results of Experiment #2: daily revenue (a), ROI (b), and spend (c) obtained by* $\mathsf{GCB}_{\mathtt{safe}}$. *The dash-dotted lines correspond to the optimum values for the revenue and ROI, while the dashed lines correspond to the values of the ROI and budget constraints.*



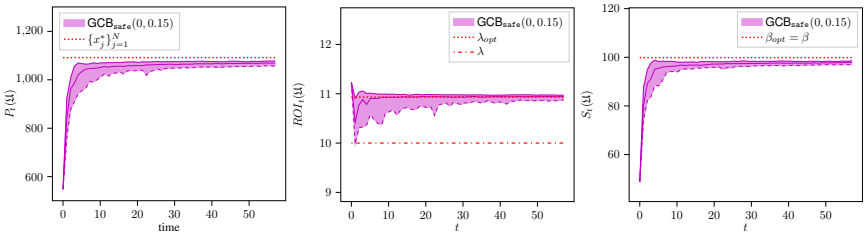
(a) **(b)** **(c)**

Figure C.3: *Results of Experiment #2: daily revenue (a), ROI (b), and spend (c) obtained by and* $\mathsf{GCB}_{\mathtt{safe}}(0, 0.05)$. *The dash-dotted lines correspond to the optimum values for the revenue and ROI, while the dashed lines correspond to the values of the ROI and budget constraints.*



(a) **(b)** **(c)**

Figure C.4: *Results of Experiment #2: daily revenue (a), ROI (b), and spend (c) obtained by and* $GCB_{\texttt{safe}}(0, 0.10)$. *The dash-dotted lines correspond to the optimum values for the revenue and ROI, while the dashed lines correspond to the values of the ROI and budget constraints.*

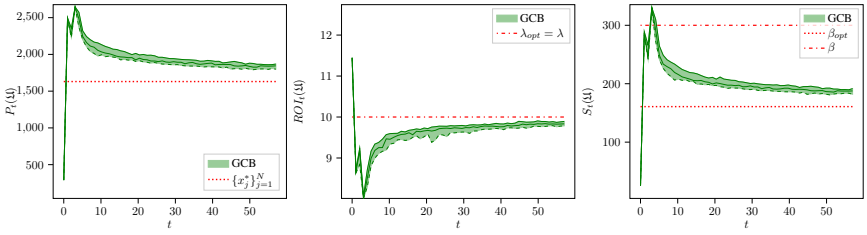


(a) **(b)** **(c)**

Figure C.5: *Results of Experiment #2: daily revenue (a), ROI (b), and spend (c) obtained by and* $GCB_{\texttt{safe}}(0, 0.15)$. *The dash-dotted lines correspond to the optimum values for the revenue and ROI, while the dashed lines correspond to the values of the ROI and budget constraints.*

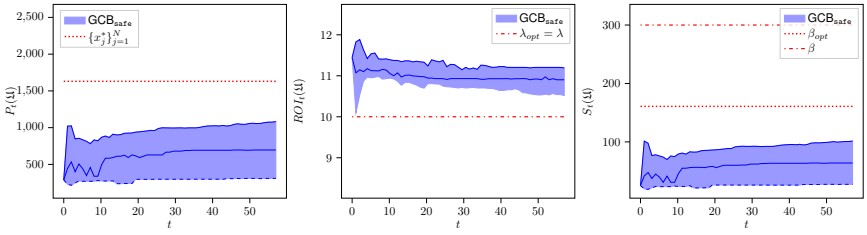
Additional Results of Experiment #3

In Figures C.6, C.7, and C.8 we report the 90% and 10% of the quantities analysed in the experimental section for Experiment #3 provided by the GCB, $\text{GCB}_{\texttt{safe}}$, and $\text{GCB}_{\texttt{safe}}(0.05, 0)$, respectively. These results show that the constraints are satisfied by $\text{GCB}_{\texttt{safe}}$, and $\text{GCB}_{\texttt{safe}}(0.05, 0)$ also with high probability. While for $\text{GCB}_{\texttt{safe}}$ this is expected due to the theoretical results we provided, the fact that also $\text{GCB}_{\texttt{safe}}(0.05, 0)$ guarantees safety w.r.t. the original optimization problem suggests that in some specific setting $\text{GCB}_{\texttt{safe}}$ is too conservative. This is reflected in a lower cumulative revenue, which might be negative from a business point of view.



(a) (b) (c)

Figure C.6: *Results of Experiment #3: daily revenue (a), ROI (b), and spend (c) obtained by GCB. The dash-dotted lines correspond to the optimum values for the revenue and ROI, while the dashed lines correspond to the values of the ROI and budget constraints.*



(a) (b) (c)

Figure C.7: *Results of Experiment #3: daily revenue (a), ROI (b), and spend (c) obtained by $\text{GCB}_{\texttt{safe}}$. The dash-dotted lines correspond to the optimum values for the revenue and ROI, while the dashed lines correspond to the values of the ROI and budget constraints.*

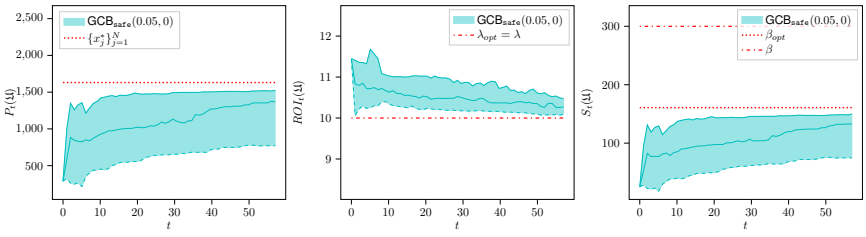


Figure C.8: *Results of Experiment #3: daily revenue (a), ROI (b), and spend (c) obtained by* $GCB_{\texttt{safe}}(0.05, 0)$. *The dash-dotted lines correspond to the optimum values for the revenue and ROI, while the dashed lines correspond to the values of the ROI and budget constraints.*

		C_1	C_2	C_3	C_4	C_5	Λ
Setting 1	θ_j	530	417	548	571	550	10.0
	δ_j	0.356	0.689	0.299	0.570	0.245	
	α_j	83	97	72	100	96	
	γ_j	0.939	0.856	0.484	0.661	0.246	
Setting 2	θ_j	597	682	698	456	444	14.0
	δ_j	0.202	0.520	0.367	0.393	0.689	
	α_j	83	98	56	60	51	
	γ_j	0.224	0.849	0.726	0.559	0.783	
Setting 3	θ_j	570	514	426	469	548	10.5
	δ_j	0.217	0.638	0.694	0.391	0.345	
	α_j	97	78	53	80	82	
	γ_j	0.225	0.680	1.051	0.412	0.918	
Setting 4	θ_j	487	494	467	684	494	12.0
	δ_j	0.348	0.424	0.326	0.722	0.265	
	α_j	62	79	76	69	99	
	γ_j	0.460	1.021	0.515	0.894	1.056	
Setting 5	θ_j	525	643	455	440	600	14.0
	δ_j	0.258	0.607	0.390	0.740	0.388	
	α_j	52	87	68	99	94	
	γ_j	0.723	0.834	1.054	1.071	0.943	
Setting 6	θ_j	617	518	547	567	576	11.0
	δ_j	0.844	0.677	0.866	0.252	0.247	
	α_j	71	53	87	98	59	
	γ_j	0.875	0.841	1.070	0.631	0.288	
Setting 7	θ_j	409	592	628	613	513	11.5
	δ_j	0.507	0.230	0.571	0.359	0.307	
	α_j	77	78	91	50	71	
	γ_j	0.810	0.246	0.774	0.516	0.379	
Setting 8	θ_j	602	605	618	505	588	13.0
	δ_j	0.326	0.265	0.201	0.219	0.291	
	α_j	67	80	99	77	99	
	γ_j	0.671	0.775	0.440	0.310	0.405	
Setting 9	θ_j	486	684	547	419	453	13.0
	δ_j	0.418	0.330	0.529	0.729	0.679	
	α_j	53	82	58	96	100	
	γ_j	0.618	0.863	0.669	0.866	0.831	
Setting 10	θ_j	617	520	422	559	457	14.0
	δ_j	0.205	0.539	0.217	0.490	0.224	
	α_j	51	86	93	61	84	
	γ_j	1.0493	0.779	0.233	0.578	0.562	

Table C.1: *Values of the parameters used in the* 10 *different settings of Experiment #4.*

Parameters of Settings of Experiment #4

We report in Table C.1 the values of the parameters of cost and number-of-click functions of the subcampaigns used in Experiment #4.