ONLINE RESOURCE ALLOCATION WITH REPLENISHABLE BUDGETS

Anonymous authors
Paper under double-blind review

ABSTRACT

Online Resource Allocation (ORA) is a fundamental framework for sequential decision-making problems under budget constraints. Classical ORA models typically assume that resources are monotonic, meaning that selecting actions can only decrease the available budget. In this work, we study a more general setting with *replenishable* budgets, in which actions may either consume or replenish resources over time. This extension is necessary to capture scenarios such as inventory systems or energy markets in which capacity can be actively recovered. We develop a dual-based algorithm that recovers best-of-both-worlds guarantees for standard ORA when the replenishment factor $\beta = 0$, and improves them when $\beta > 0$. In particular, our algorithm attains $\widetilde{\mathcal{O}}(\sqrt{T})$ regret in the stochastic setting and $\widetilde{\mathcal{O}}(\sqrt{T})$ α-regret in the adversarial setting, where α depends on the per-round budget and on the replenishment factor of the void action. Moreover, the algorithm ensures strict satisfaction of the budget constraints.

1 INTRODUCTION

Online Resource Allocation (ORA) (Devanur & Hayes, 2009; Feldman et al., 2010) is a fundamental framework for sequential decision-making under *budget constraints*. At each round, the decision maker observes a request and selects an appropriate action. Crucially, before making this decision, the request fully reveals the reward and resource consumption associated with every possible action (Agrawal et al., 2009; Kesselheim et al., 2013). This full-information structure distinguishes ORA from most online learning paradigms, in which the learner must act without complete knowledge of the current-round outcomes. Consequently, the main challenge in ORA stems from uncertainty about future requests: learning algorithms must balance the immediate rewards obtained from current requests against the need to preserve resources for future ones (Devanur et al., 2019).

ORA problems arise in a wide range of applications, including online auctions (Balseiro & Gur, 2019), online advertising (Mehta et al., 2007) and revenue management (Ball & Queyranne, 2009). Driven by these practical motivations, various works in literature have addressed stochastic ORA problem (Li et al., 2020; Sun et al., 2022). A few efforts have been made beyond standard i.i.d assumption, considering non-stationary setting (Esfandiari et al., 2017) and adversarial setting (Balseiro & Gur, 2019; Stradi et al., 2025), although under strict assumptions. Notably, Balseiro et al. (2023) are the first to consider general ORA problems in both stochastic and adversarial regimes by designing a *best-of-both-world* algorithm.

Despite their contributions, the vast majority of the existing ORA literature relies on a *monotonic* consumption model: every action taken by the agent depletes the available budget. The budget decreases over time until it is eventually exhausted, and the only action left is a *void* action, which has no effect on the resources. While natural in many classical applications, this assumption excludes a broad class of sequential allocation problems in which resources can also be replenished. In many real-world scenarios, actions may either consume or restore capacity, and the agent may resort to a budget-restoring action when it is running low on budget. Consider, for instance, an inventory system in which a retailer can satisfy customer requests depleting the stock, or place orders from suppliers replenishing the stock. Similar dynamics arise in cloud computing, when a server can allocate computational resources to incoming tasks and at the same time terminate tasks to recover memory and CPU capacity.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

In such settings, resource dynamics are inherently *non-monotonic*. The budget oscillates over time, introducing new algorithmic challenges that standard ORA frameworks cannot handle. Classical primal-dual algorithms (Balseiro et al., 2023) rely on monotonic resource depletion to naturally bound the dual variables. When costs can be negative due to replenishments, standard gradient updates risk to artificially decrease the dual multipliers and make the algorithm over-consume resources.

In this work, we address the following research question:

How does the replenishment factor affect the achievable regret bounds and the competitive ratio in online resource allocation?

We answer this question by proposing a novel best-of-both-worlds primal-dual framework capable of handling non-monotonic resource utilization. We prove that the availability of replenishments can be structurally exploited to improve the agent's performance. Specifically, our proposed algorithm achieves simultaneously sublinear expected regret in stochastic environments and sublinear fractional regret in adversarial ones, while deterministically satisfying the budget constraints.

1.1 Original Contribution

In this work, we formally study the online resource allocation problem with replenishable budgets. We propose a dual-based framework capable of handling non-monotonic resource utilization under both stochastic and adversarial assumptions. As summarized in Table 1, our algorithm recovers the optimal guarantees of the state-of-the-art (Balseiro et al., 2023) for standard monotonic environments ($\beta = 0$) and improves state-of-the-art bounds when replenishment is active ($\beta > 0$). Specifically, we establish the following theoretical guarantees:

- $\widetilde{\mathcal{O}}(\sqrt{T})$ cumulative expected regret in the *stochastic* setting, i.e., when requests are sampled i.i.d from an unknown fixed distribution. Notably, the bound scales inversely with the augmented margin $\rho + \beta$, yielding a strictly tighter regret guarantee compared to standard monotonic models;
- $\widetilde{\mathcal{O}}(\sqrt{T})$ cumulative α-regret in the *adversarial* setting, when requests are selected in advance by an adversary, with $\alpha := {}^{\rho+\beta}/_{1+\beta}$ problem-dependent parameter.

The bounds in Table 1 use knowledge of β to tune the learning rate. At the same time, in both settings, the algorithm strictly satisfies the hard budget constraints over the time horizon. Furthermore, the execution of the algorithm does not require prior knowledge of the true replenishment factor β.

1.2 Technical Challenges

In classical ORA, primal-dual algorithms rely on the monotonic depletion of resources to naturally bound and stabilize the dual variables (Balseiro et al., 2023). Since resource consumption is always non-negative, the dual update acts as a straightforward penalty mechanism. Specifically, when per-round consumption exceeds ρ, the corresponding dual multipliers increase, discouraging the primal algorithm from selecting resource-intensive actions.

When introducing active replenishment, this monotonic structure breaks down. Negative "costs" decrease the corresponding dual multipliers. An action might consume one resource while actively restoring another, pushing the respective dual multipliers in opposite directions. This lack of alignment creates a critical vulnerability in the Lagrangian objective: cross-constraints compensation. An action that severely depletes a critical resource may still appear optimal if it heavily replenishes a different one, masking the violation. Thus, the algorithm risks greedily over-consuming the scarce resource. This mechanism leads to premature budget exhaustion, forcing the agent to play the void action until the budget is recovered.

Overcoming this limitation requires a dual-space approach that tolerates non-monotonic consumption and unaligned constraints. In this paper, we tackle this challenge by designing a best-of-both-worlds dual-based framework that prevents the artificial deflation of lagrangian multipliers, keeping them tightly bounded even in the presence of negative costs.

Table 1: Comparison of theoretical guarantees, with known β for our bounds. Exposing the dependence on the budget margin (ρ) and the replenishment margin (β) highlights how our framework not only accommodates non-monotonic ORA ($\beta > 0$), but actively exploits replenishments to tighten the regret bounds and significantly improve the adversarial competitive ratio.

Metric	Margin	Balseiro et al. (2023)	Our Work
Stochastic regret (R_T)	$\boldsymbol{\beta = 0}$	$\widetilde{\mathcal{O}}\left(\frac{1}{\rho}\sqrt{T}\right)$	$\widetilde{\mathcal{O}}\left(\frac{1}{\rho}\sqrt{T}\right)$
	$\boldsymbol{\beta > 0}$	–	$\widetilde{\mathcal{O}}\left(\frac{1}{\rho+\beta}\sqrt{T}\right)$
Adversarial fractional regret (α-R_T)	$\boldsymbol{\beta = 0}$	$\widetilde{\mathcal{O}}\left(\frac{1}{\rho}\sqrt{T}\right)$	$\widetilde{\mathcal{O}}\left(\frac{1}{\rho}\sqrt{T}\right)$
	$\boldsymbol{\beta > 0}$	–	$\widetilde{\mathcal{O}}\left(\frac{1}{\rho+\beta}\sqrt{T}\right)$
Competitive Ratio (α)	$\boldsymbol{\beta = 0}$	ρ	ρ
	$\boldsymbol{\beta > 0}$	–	$\frac{\rho+\beta}{1+\beta}$

1.3 Related works

Online Resource Allocation Classical Online Resource Allocation framework has been studied in many real-world scenarios, such as online auctions (Zhou et al., 2008; Balseiro & Gur, 2019), online advertising (Mehta et al., 2007), and revenue management (Talluri & van Ryzin, 2004; Ball & Queyranne, 2009). Most of the existing literature studied ORA under stochastic feedback. Early two-phases approaches (Devanur & Hayes, 2009; Feldman et al., 2010) estimated static dual prices from an initial data sample, leading to suboptimal regret bounds of order $\mathcal{O}(T^{2/3})$. To overcome this limitation, subsequent works (Agrawal et al., 2009; Kesselheim et al., 2013; Devanur et al., 2019) dynamically updated decisions by periodically re-solving linear programming relaxations, securing the optimal $\widetilde{\mathcal{O}}(\sqrt{T})$. To avoid the computational complexity of repeated LP solvers, recent studies have shifted toward gradient-based primal-dual schemes (Li et al., 2020; Sun et al., 2022). Although these methods attain strong theoretical guarantees, they require specific assumptions or problem structures.

When relaxing the i.i.d. assumption, the literature branches into non-stationary environments (Ciocan & Farias, 2011; Esfandiari et al., 2017) and fully adversarial models. Early results in the adversarial setting study the specific problem of online matching and AdWords (Buchbinder et al., 2007; Mehta et al., 2007), without a direct application to more general settings. The first *best-of-both-worlds* algorithm for Online Resource Allocation is the one presented in (Balseiro et al., 2023), which is a dual-based approach simultaneously $\widetilde{\mathcal{O}}(\sqrt{T})$ regret in the stochastic setting,and $\widetilde{\mathcal{O}}(\sqrt{T})$ α-regret in the non-stationary and in the adversarial setting. Chiefari et al. (2026) study a generalized ORA framework, finding best-of-both-worlds guarantees incorporating general long-term constraints alongside standard budgets. The algorithm proposed in this work attains $\widetilde{\mathcal{O}}(\sqrt{T})$ regret in the stochastic setting, $\widetilde{\mathcal{O}}(\sqrt{T})$ α-regret in the adversarial setting and $\widetilde{\mathcal{O}}(\sqrt{T})$ cumulative violation of the general constraints in both settings.

Parallel analytical efforts have been dedicated to more specific domains. (Balseiro & Gur, 2019) propose a pacing-based strategy attaining $\widetilde{\mathcal{O}}(\sqrt{T})$ regret in the stochastic setting under suitable assumptions, and provides an impossibility result for the adversarial case: no algorithm can exceed the competitive ratio v/ρ, with v denoting an upper bound on the bidder's per-item value. A later work (Feng et al., 2023) extends the online bidding literature by including the Return-on-Spend constraint, which ensures that the ratio of value to expenditure remains over a prescribed threshold. Although it obtains strong guarantees, this analysis relies on the specific structure of the RoS constraints, thus cannot be generalized. Beyond these bidding-specific results, other contributions have pursued competitive-ratio guarantees directly under budget constraints. For instance, in the single-resource online Knapsack setting, (Zhou et al., 2008) establish competitive ratios depending on the range of the value-to-weight ratio of the items.

Despite their theoretical depth, all the aforementioned frameworks inherently rely on a strictly monotonic budget depletion model. Our work fundamentally diverges from this paradigm. By modeling actions that may yield negative costs, we introduce a non-monotonic budget dynamic through active replenishments, requiring a self-regulating mechanism in the dual space. This concept is detailed in Section 4.1.

Online Learning with Replenishment The idea that resource consumption need not be monotonic has surfaced only recently in the online learning literature, and almost exclusively within the bandit feedback model. The first step in this direction is due to (Kumar & Kleinberg, 2022), who introduced the *Bandits with Replenishable Knapsacks* (BwRK) framework. Their work addresses non-monotonic resource utilization by providing instance-dependent logarithmic bounds specifically tailored for stochastic input models, extending the classic BwK framework (Badanidiyuru et al., 2013; Castiglioni et al., 2022). Building on this formulation, (Bernasconi et al., 2024b) proposed a primal-dual method with best-of-both-worlds guarantees for the replenishable setting. Their algorithm yields a $\widetilde{\mathcal{O}}(\sqrt{T})$ regret bound in the stochastic setting, matching previous results, and ensures a constant competitive ratio in the adversarial setting.

While these works successfully tackle the challenges of non-monotonic resource consumption, they remain confined to the bandit feedback framework, where the agent observes the realized reward and resource consumption *after* the action is selected. Our work addresses the challenge of active replenishment within the full-information ORA setting. We provide further discussion on related works in Appendix A

2 Preliminaries

We study the problem of *Online Resource Allocation* (ORA) with replenishable budgets, in which a decision maker (agent) selects actions over a finite horizon of T rounds. We denote the action space of the agent by $\mathcal{X} \subseteq \mathbb{R}^K$. At each round $t \in [T]$,[1] the agent observes an input tuple $\gamma_t = (f_t, \boldsymbol{b}_t)$, where $f_t : \mathcal{X} \to [0,1]$ is the reward function and $\boldsymbol{b}_t = (b_{t,1}, \dots, b_{t,m}) : \mathcal{X} \to [-1,1]^m$ is the vector of resource consumption functions, with $b_{t,i} : \mathcal{X} \to [-1,1]$ denoting the consumption function for resource $i \in [m]$. The input functions are fully revealed to the agent before any decision is made. We assume that the maxima below are attained and use a fixed measurable rule to select a maximizer. After observing the input, the agent selects an action $x_t \in \mathcal{X}$, which yields a reward $f_t(x_t)$ and a consumption $b_{t,i}(x_t)$ for each resource $i \in [m]$. Unlike traditional ORA models, this setting allows for *non-monotonic* resource utilization: a negative consumption, $b_{t,i}(x_t) < 0$, means that action x_t *replenishes* the budget of resource i rather than consuming it. The input tuples may be either *stochastic*, *i.e.*, generated i.i.d. across rounds according to a fixed probability distribution, or *adversarial*, *i.e.*, chosen arbitrarily by an adversary. Throughout the paper, we denote by Γ the set of all the possible input tuples $\gamma = (f, \boldsymbol{b})$.

The decision maker maintains an m-dimensional vector $\boldsymbol{B}_t \in \mathbb{R}^m_{\geq 0}$ representing the available budget for each resource, which is updated at each round. The initial budget is $T\rho$ for each resource, where $\rho \in (0,1)$ denotes the per-round budget; equivalently, we let $\boldsymbol{\rho} = \rho \mathbf{1} \in \mathbb{R}^m_{\geq 0}$ denote the vector of per-round budgets. Following the standard assumption in the ORA literature (see, *e.g.*, (Balseiro et al., 2023)), we assume the existence of a *void action* $\varnothing \in \mathcal{X}$. To allow recovery from budget depletion when $\beta > 0$, we require the void action to provide a guaranteed replenishment of every resource. Formally, there exists a constant $\beta \in [0,1]$ such that, for every input tuple $(f, \boldsymbol{b}) \in \Gamma$, we have $f(\varnothing) = 0$ and $b_i(\varnothing) \leq -\beta$ for every $i \in [m]$.

Remark 2.1 (On the void action and replenishment)**.** *We assume that the void action $\varnothing$ provides a replenishment of at least β for every input tuple, even in the stochastic setting. This assumption captures realistic ORA applications in which a fallback option, such as pausing operations or acquiring additional resources, guarantees a minimum replenishment regardless of the stochasticity of incoming input requests. The lower bound β need not be known to the algorithm. The case $\beta = 0$ includes standard monotonic ORA.*

[1] Throughout this paper, we denote by $[a] := \{1, \dots, a\}$ the set of the first $a \in \mathbb{N}$ natural numbers.

Given a specific sequence of inputs $\boldsymbol{\gamma} = (\gamma_1, ..., \gamma_T)$, where $\gamma_t = (f_t, \boldsymbol{b}_t)$ for all $t \in [T]$, we define the *cumulative reward* of the agent over the time horizon T as:

$$\texttt{Rew}(T, \boldsymbol{\gamma}) := \sum_{t \in [T]} f_t(x_t).$$

The goal of the agent is to maximize $\texttt{Rew}(T, \boldsymbol{\gamma})$, while satisfying the resource consumption constraint, *i.e.*:

$$\sum_{t \in [T]} b_{t,i}(x_t) \leq \rho T \quad \forall i \in [m].$$

2.1 Lagrangian Formulation

A crucial tool to handle resource constraints across the time horizon is the Lagrangian formulation. This approach allows us to incorporate the budget constraints directly into the objective function. For any round $t \in [T]$, given the instantaneous reward function f_t, the resource consumption vector $\boldsymbol{b}_t$ and the per-iteration budget ρ, we define the instantaneous Lagrangian function as:

$$\mathcal{L}_{f_t, \boldsymbol{b}_t}(x, \boldsymbol{\lambda}) := f_t(x) - \sum_{i=1}^{m} \lambda_i (b_{t,i}(x) - \rho) = f_t(x) - \langle \boldsymbol{\lambda}, \boldsymbol{b}_t(x) - \boldsymbol{\rho} \rangle,$$

where $\boldsymbol{\lambda} \in \mathbb{R}^m_{\geq 0}$ is the vector of dual multipliers.

In this formulation, the dual variables λ_i act as penalty weights for resource utilization. For actions consuming more than ρ units of resource i, the term $b_{t,i}(x) - \rho$ is positive, thus penalizing the Lagrangian objective. However, in our framework with replenishable budgets, playing an action that restores a resource j guarantees a strictly negative term $b_{t,j}(x) - \rho$ to the objective. This structural property naturally encourages the algorithm to select replenishing actions when the dual multipliers become large, preventing the budget to be prematurely exhausted.

2.2 Stochastic Setting

In the stochastic setting, we assume that the input tuples $\gamma_t = (f_t, b_t)$ are sampled i.i.d. from a fixed but unknown probability distribution $\mathcal{P}$ over the input space Γ. Formally, the input sequence is generated as $\boldsymbol{\gamma} = (\gamma_1, ..., \gamma_T) \sim \mathcal{P}^T$. The standard baseline for this setting is defined as the optimal strategy that, in expectation over all input realizations, maximizes rewards while satisfying constraints, *i.e.*, $\text{OPT}_{\text{stoc}} = \mathbb{E}_{\boldsymbol{\gamma} \sim \mathcal{P}^T}[\text{OPT}(\boldsymbol{\gamma})]$, where:

$$\text{OPT}(\boldsymbol{\gamma}) := \begin{cases} \max_{(x_1, ..., x_T) \in \mathcal{X}^T} & \sum_{t \in [T]} f_t(x_t) \\ \text{s.t.} & \sum_{t \in [T]} b_{t,i}(x_t) \leq \rho T \quad \forall i \in [m]. \end{cases}$$

In the stochastic setting, we evaluate the performance of the algorithm by using the *cumulative regret* over T rounds, defined as:

$$R_T := \text{OPT}_{\text{stoc}} - \mathbb{E}_{\boldsymbol{\gamma} \sim \mathcal{P}^T}[\texttt{Rew}(T, \boldsymbol{\gamma})].$$

2.3 Adversarial Setting

In the adversarial setting, we assume that the input tuple γ_t is chosen arbitrarily at each round t by an adversary. Exploiting the replenishable nature of the resources, we can evaluate the agent's performance against a more challenging benchmark, the dynamic *unconstrained* optimum. Given a fixed input sequence $\boldsymbol{\gamma} = (\gamma_1, \ldots, \gamma_T) \in \Gamma^T$, we define OPT_{adv} as:

$$\text{OPT}_{\text{adv}} := \max_{(x_1, ..., x_T) \in \mathcal{X}^T} \sum_{t \in [T]} f_t(x_t).$$

Due to the fundamental impossibility of simultaneously achieving sublinear regret and constraint satisfaction under fully adversarial inputs (see, *e.g.*, (Balseiro et al., 2023)), the standard regret definition is not viable for this setting. To overcome this issue, we measure the performance of our algorithm against a fraction $\alpha \in (0, 1)$ of the optimum, called α-regret:

$$\alpha\text{-}R_T := \alpha \cdot \text{OPT}_{\text{adv}} - \texttt{Rew}(T),$$

Algorithm 1 REPLENISHING DUAL GRADIENT DESCENT

Require: T, η, ρ

1: $\boldsymbol{\lambda}_1 \leftarrow \mathbf{0}$, $\boldsymbol{B}_1 \leftarrow T\boldsymbol{\rho}$
2: **for** $t = 1, 2, \ldots, T$ **do**
3: Observe input tuple $\gamma_t = (f_t, \boldsymbol{b}_t)$
4: **if** $\exists i \in [m] : B_{t,i} < 1$ **then**
5: $x_t \leftarrow \varnothing$
6: $\boldsymbol{B}_{t+1} \leftarrow \boldsymbol{B}_t - \boldsymbol{b}_t(\varnothing)$ ▷ Overall budget update
7: $\boldsymbol{\lambda}_{t+1} \leftarrow \boldsymbol{\lambda}_t$
8: **else**
9: $x_t \leftarrow \arg\max_{x \in \mathcal{X}} \mathcal{L}_{f_t, \boldsymbol{b}_t}(x, \boldsymbol{\lambda}_t)$
10: $\boldsymbol{B}_{t+1} \leftarrow \boldsymbol{B}_t - \boldsymbol{b}_t(x_t)$ ▷ Overall budget update
11: $\boldsymbol{\lambda}_{t+1} \leftarrow \Pi_{\mathbb{R}^m_{\geq 0}}(\boldsymbol{\lambda}_t + \eta(\boldsymbol{b}_t(x_t) - \boldsymbol{\rho}))$ ▷ Dual update
12: **end if**
13: **end for**

where $\alpha := \frac{\rho+\beta}{1+\beta}$ and we let $\texttt{Rew}(T) := \texttt{Rew}(T, \boldsymbol{\gamma})$, as the dependence on the deterministic sequence of input tuples $\boldsymbol{\gamma}$ is clear from the context.

Notice that the replenishable nature of the environment—specifically, the existence of the void action guaranteeing a replenishment $\beta > 0$—implicitly provides a Slater's condition with margin $\rho + \beta$. If $\beta = 0$, no strictly positive replenishment is guaranteed, and the Slater's margin becomes the known per-round budget as in classical budget-constrained online allocation (see, *e.g.*, (Balseiro et al., 2023)).

3 ALGORITHM

In this section, we describe the behavior of REPLENISHING DUAL GRADIENT DESCENT, which is the main algorithmic contribution of this work.

The pseudocode of the algorithm is provided in Algorithm 1. It receives as input the time horizon T, the learning rate η and the per-round budget ρ. As a first step, it initializes the dual variables $\boldsymbol{\lambda}_1$ to zero and instantiates the starting available budget $\boldsymbol{B}_1 := T\boldsymbol{\rho}$ (Line 1).

At each round t, after observing the current input tuple γ_t (Line 3), the algorithm verifies that the available budget is sufficient to proceed safely (Line 4). Since the maximum resource consumption for any action is bounded by 1 (Section 2), a remaining budget $B_{t,i} < 1$ implies that selecting a budget-consuming action could lead to a hard constraint violation for resource i. Therefore, if this critical threshold is reached, the algorithm acts conservatively and plays the void action $\varnothing$ (Line 5). This action does not consume any budget; instead, it provides a guaranteed replenishment ($\beta \geq 0$) effectively increasing the remaining capacity (Line 6). Crucially, during the rounds in which the algorithm is forced to play the void action, the dual variables are not updated (Line 7). Pausing the dual updates is a fundamental design choice: if the update were performed during replenishment, the dual multipliers would be artificially decreased, causing the algorithm to over-consume resources once the budget is recovered.

If the available budget is safe ($B_{t,i} \geq 1$ for all i), the algorithm selects an action x_t by greedily maximizing the instantaneous Lagrangian function (Line 9). The available overall budget is updated based on the consumption given by the selected action (Line 10). As standard in ORA framework, this primal update requires solving a single optimization problem over the action space $\mathcal{X}$, avoiding the computational burden of periodically re-solving large-scale linear programs over the time horizon.

Notice that the budget remains non-negative at every round. Indeed, an active round starts with $B_{t,i} \geq 1$ and consumes at most 1 unit of each resource, while a forced round only replenishes resources. Thus, for every $s \in [T]$ and $i \in [m]$, we have $\sum_{t=1}^{s} b_{t,i}(x_t) \leq \rho T$.

Finally, the algorithm updates the dual variables $\boldsymbol{\lambda}_t$ through a projected *Online Gradient Descent* (OGD) step onto the positive orthant (Line 11). Intuitively, if the selected action consumed more

budget than the per-round budget $\boldsymbol{\rho}$, the gradient step increases the multipliers, effectively penalizing "expensive" actions in subsequent rounds. Conversely, if resources are underutilized, the multipliers decrease, encouraging the algorithm to exploit higher-reward, budget-consuming actions in the future.

4 Theoretical Results

In this section, we prove the theoretical guarantees attained by Algorithm 1. In particular, we first show that the Lagrangian multipliers remain bounded during the learning dynamics. Subsequently, we analyze the cumulative regret in the stochastic setting and the cumulative α-regret in the adversarial setting.

4.1 Bounding the Lagrangian Multipliers

A key result in our analysis is showing how the dual multipliers are automatically bounded by properties of the dual regret minimizer. In standard ORA frameworks (see, *e.g.*, Balseiro et al. (2023)) resource consumption is assumed to be strictly non-negative. This monotonic depletion of resources ensures that the dual variables naturally self-regulate and remain bounded without requiring explicit projection onto a bounded set. Moreover, in these settings, the strict feasibility of the constraints is determined solely by the per-round budget ρ, typically known by the algorithm. In our setting, the non-monotonicity of the budget dynamics could, in principle, destabilize the dual space by making different components of the multiplier vector increase or decrease in a non-aligned way. Thus, we need to formally prove that the primal-dual learning dynamics still naturally self-regulate despite the presence of negative components. To do so, we exploit the existence of the guaranteed replenishment margin $\beta > 0$, which effectively increases the implicit Slater's condition to $\rho + \beta$. This augmented margin allows the algorithm to maintain a tightly bounded dual space.

Lemma 4.1. *In both the stochastic and the adversarial setting, Algorithm 1 with $0 < \eta \leq 1/2m$ guarantees that the ℓ_1-norm of the dual variables is bounded for all $t \in [T+1]$ as:*

$$\|\boldsymbol{\lambda}_t\|_1 \leq \frac{4\sqrt{m}}{\rho + \beta}.$$

This result formally captures the structural power of our algorithm. Indeed, the maximum magnitude of the multipliers is inversely proportional to the effective margin $\rho + \beta$. We prove Lemma 4.1 by controlling the squared Euclidean norm of the dual variables. During active rounds, the greedy primal step dominates the void action. Thus, we obtain:

$$\mathcal{L}_{f_t,\boldsymbol{b}_t}(x_t, \boldsymbol{\lambda}_t) \geq \mathcal{L}_{f_t,\boldsymbol{b}_t}(\varnothing, \boldsymbol{\lambda}_t) \geq (\rho + \beta)\|\boldsymbol{\lambda}_t\|_1.$$

Since rewards are bounded by 1, this implies:

$$\langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho} \rangle \leq 1 - (\rho + \beta)\|\boldsymbol{\lambda}_t\|_1.$$

Moreover, the projected OGD update satisfies:

$$\|\boldsymbol{\lambda}_{t+1}\|_2^2 \leq \|\boldsymbol{\lambda}_t\|_2^2 + 2\eta\big(1 - (\rho + \beta)\|\boldsymbol{\lambda}_t\|_1\big) + 4\eta^2 m.$$

Then, for $\eta \leq 1/2m$, the squared norm cannot increase whenever $\|\boldsymbol{\lambda}_t\|_1 \geq 2/\rho+\beta$. Below this threshold, a single update has bounded size. During forced rounds, the dual variables remain unchanged. Combining these properties yields the stated bound. The full details of the proof can be found in Appendix B.

Notice that the learning rates in Theorems 4.2 and 4.4 may exceed $1/2m$ for small horizons. In this case, the stated regret bounds already exceed T and follow from bounded rewards. Thus, we only apply Lemma 4.1 when its learning-rate condition holds.

4.2 Stochastic Setting

In this section, we analyze the performance of Algorithm 1 in the stochastic setting, where inputs are drawn from a fixed but unknown distribution.

Theorem 4.2. *In the stochastic setting, assuming the replenishment parameter β is known, Algorithm 1 with learning rate $\eta := \frac{3}{2(\rho+\beta)}\sqrt{\frac{1}{T}}$ guarantees:*

$$R_T \leq \frac{2 + 12m\sqrt{T}}{\rho + \beta}.$$

If β is unknown, Algorithm 1 with learning rate $\eta := \frac{3}{2\rho}\sqrt{\frac{1}{T}}$ guarantees:

$$R_T \leq \frac{2}{\rho + \beta} + 6m\left(\frac{\rho}{(\rho+\beta)^2} + \frac{1}{\rho}\right)\sqrt{T}.$$

Theorem 4.2 establishes that Algorithm 1 attains $\mathcal{O}(\sqrt{T})$ cumulative regret bound, which is the optimal guarantee for stochastic ORA. The proof includes two phases: we first derive an upper bound on the stochastic benchmark OPT_{stoc} by invoking weak duality and defining the expected Lagrangian over the input distribution:

$$\bar{\mathcal{L}}_{f,\boldsymbol{b}}(\boldsymbol{\lambda}|\mathcal{P}) := \mathbb{E}_{(f,\boldsymbol{b})\sim\mathcal{P}}[\max_{x\in\mathcal{X}} \mathcal{L}_{f,\boldsymbol{b}}(x,\boldsymbol{\lambda})].$$

Next, we evaluate algorithmic performance across the active decision rounds. By taking the conditional expectation of the instantaneous reward $f_t(x_t)$ with respect to the filtration of past events $\Gamma_{t-1} = \{\gamma_1, ..., \gamma_{t-1}\}$, we can relate it to the instantaneous Lagrangian via a martingale argument:

$$\mathbb{E}[f_t(x_t)\cdot\mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}|\Gamma_{t-1}] = \mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}\bar{\mathcal{L}}_{f,\boldsymbol{b}}(\boldsymbol{\lambda}_t|\mathcal{P}) + \mathbb{E}[\mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}\langle\boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle|\Gamma_{t-1}].$$

By summing over the time horizon and applying the Tower Rule, this step allows us to upper bound the overall regret by the expected number of forced replenishments and the cumulative dual penalty during the active rounds $\mathcal{T}_{\text{active}}$:

$$R_T \leq \mathbb{E}_{\boldsymbol{\gamma}}[|\mathcal{T}_{\text{forced}}|] - \mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}}\langle\boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle\right].$$

Finally, we bound the cumulative dual penalty by exploiting the no-regret guarantees of the OGD update. Crucially, the physical limits of the budget ensure that whenever the safety threshold is reached—thus triggering a forced replenishment—the overall resource consumption provides a strict lower bound on the dual penalty. The complete proof can be found in Appendix B.

Remark 4.3. *As the theorem shows, the algorithm does not need to know the true replenishment parameter β to guarantee the optimal $\widetilde{\mathcal{O}}(\sqrt{T})$ rate. When β is unknown, the learning rate is tuned solely on the known per-round budget ρ, since $\rho \leq \rho + \beta$. While the constant factors inevitably worsen, suggesting that an accurate estimate of β is useful to tighten the bound, the sublinearity is preserved also in this case.*

4.3 Adversarial Setting

Finally, in this section, we analyze the performance of the algorithm in the adversarial regime, where inputs may arbitrarily change at each round.

Theorem 4.4. *In the adversarial setting, assuming the replenishment parameter β is known, Algorithm 1 with learning rate $\eta := \frac{3}{2(\rho+\beta)}\sqrt{\frac{1}{T}}$ guarantees:*

$$\alpha\text{-}R_T \leq \frac{2 + 12m\sqrt{T}}{\rho + \beta}.$$

If β is unknown, Algorithm 1 with learning rate $\eta := \frac{3}{2\rho}\sqrt{\frac{1}{T}}$ guarantees:

$$\alpha\text{-}R_T \leq \frac{2}{\rho + \beta} + 6m\left(\frac{\rho}{(\rho+\beta)^2} + \frac{1}{\rho}\right)\sqrt{T}.$$

Theorem 4.4 proves that REPLENISHING DUAL GRADIENT DESCENT ensures an $\mathcal{O}(\sqrt{T})$ bound for the α-regret under fully adversarial inputs. We establish this result by leveraging the regret guarantees of the unconstrained OGD updates. The analysis naturally partitions the time horizon into the set of active decision rounds $\mathcal{T}_{\text{active}}$ and the set of forced replenishment rounds $\mathcal{T}_{\text{forced}}$. During the forced replenishments, the algorithm conservatively plays the void action $\varnothing$, thus the regret incurred in this phase is trivially bounded by $|\mathcal{T}_{\text{forced}}|$.

To analyze the active rounds, we rely on the fact that the primal update greedily maximizes the instantaneous Lagrangian. We benchmark the algorithm's performance against a tailored, randomized reference strategy $\xi_t^\diamond$. Specifically, at each step t, this strategy selects the hindsight optimal action x_t^* with probability $\alpha := \frac{\rho+\beta}{1+\beta}$ and defaults to the void action with probability $1-\alpha$. The parameter α is determined by the requirement to offset the worst-case resource consumption of the offline optimal sequence with the guaranteed margin β provided by the void action.

Since $f_t(\varnothing) = 0$, the expected reward of this strategy scales exactly by α:

$$\mathbb{E}_{x\sim\xi_t^\diamond}[f_t(x)] = \alpha \cdot f_t(x_t^*) + (1-\alpha) f_t(\varnothing) \geq \alpha \cdot f_t(x_t^*).$$

Simultaneously, thanks to the guaranteed replenishment margin β provided by the void action, this randomized policy remains strictly feasible in expectation:

$$\mathbb{E}_{x\sim\xi_t^\diamond}[b_{t,i}(x)] \leq \alpha \cdot 1 + (1-\alpha)(-\beta) = \rho.$$

This allows us to reduce the α-regret analysis to bounding the cumulative dual penalty during the active rounds $\mathcal{T}_{\text{active}}$, plus the trivial regret incurred during forced replenishments $\mathcal{T}_{\text{forced}}$:

$$\alpha\text{-}R_T \leq -\sum_{t\in\mathcal{T}_{\text{active}}} \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle + |\mathcal{T}_{\text{forced}}|.$$

Similarly to the stochastic analysis, we control the penalty via the no-regret properties of the OGD updates. The complete proof is provided in Appendix B.

5 DISCUSSION AND OPEN PROBLEMS

In this work, we introduce a generalized model for online resource allocation that includes replenishable budgets, departing from the standard monotonic consumption assumption. We present REPLENISHING DUAL GRADIENT DESCENT, a dual-based algorithm that achieves optimal $\widetilde{\mathcal{O}}(\sqrt{T})$ expected regret in stochastic environments and $\widetilde{\mathcal{O}}(\sqrt{T})$ α-regret in adversarial settings, while strictly satisfying the budget constraints. Our analysis shows that the presence of an active replenishment margin ($\beta > 0$) tightens the regret bounds and improves the adversarial competitive ratio α, without requiring prior knowledge of the parameter β.

This problem formulation opens several directions for future theoretical research. For instance:

- Replenishment in expectation: we currently assume that the void action $\varnothing$ provides a replenishment of at least β for every input tuple. A natural relaxation is to require this lower bound only in conditional expectation. In this context, it would be interesting to study whether the algorithm can estimate β, rather than relying on the worst case bound when β is unknown.
- Resource-specific replenishing actions: our current model assumes a single void action that uniformly replenishes all m resources by at least β. In many real-world applications (e.g., inventory systems), however, replenishment is inherently resource-specific. It may be available through multiple independent replenishing actions with different margins, instead of a single global one. Extending our dual-based framework to this more general setting would broaden its practical applicability.

AI USE STATEMENT

Generative AI tools were used during the drafting of this paper to improve the readability and refine the English exposition of both the text and the proofs. The authors bear full responsibility for the scientific content and the final manuscript.

REFERENCES

Shipra Agrawal and Nikhil R. Devanur. Bandits with global convex constraints and objective. *Oper. Res.*, 67(5):1486–1502, September 2019. ISSN 0030-364X. doi: 10.1287/opre.2019.1840. URL https://doi.org/10.1287/opre.2019.1840.

Shipra Agrawal, Zizhuo Wang, and Yinyu Ye. A dynamic near-optimal algorithm for online linear programming. *CoRR*, abs/0911.2974, 2009. URL http://arxiv.org/abs/0911.2974.

Ashwinkumar Badanidiyuru, Robert Kleinberg, and Aleksandrs Slivkins. Bandits with knapsacks. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pp. 207–216. IEEE, October 2013. doi: 10.1109/focs.2013.30. URL http://dx.doi.org/10.1109/FOCS.2013.30.

Michael O. Ball and Maurice Queyranne. Toward robust revenue management: Competitive analysis of online booking. *Operations Research*, 57(4):950–963, 2009. doi: 10.1287/opre.1080.0654. URL https://doi.org/10.1287/opre.1080.0654.

Santiago R Balseiro and Yonatan Gur. Learning in repeated auctions with budgets: Regret minimization and equilibrium. *Management Science*, 65(9):3952–3968, 2019.

Santiago R Balseiro, Haihao Lu, and Vahab Mirrokni. The best of many worlds: Dual mirror descent for online allocation problems. *Operations Research*, 71(1):101–119, 2023.

Martino Bernasconi, Matteo Castiglioni, and Andrea Celli. No-regret is not enough! Bandits with general constraints through adaptive regret minimization. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*. PMLR, 2024a.

Martino Bernasconi, Matteo Castiglioni, Andrea Celli, and Federico Fusco. Bandits with replenishable knapsacks: the best of both worlds. In *The Twelfth International Conference on Learning Representations*, 2024b. URL https://openreview.net/forum?id=54a1495b06c4ee2f07184afb9a37abda.

Niv Buchbinder, Kamal Jain, and Joseph Seffi Naor. Online primal-dual algorithms for maximizing ad-auctions revenue. In *Proceedings of the 15th Annual European Conference on Algorithms*, ESA'07, pp. 253–264, Berlin, Heidelberg, 2007. Springer-Verlag. ISBN 3540755195.

Matteo Castiglioni, Andrea Celli, and Christian Kroer. Online learning with knapsacks: the best of both worlds. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 2767–2783. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/castiglioni22a.html.

Matteo Castiglioni, Andrea Celli, and Christian Kroer. Online learning under budget and roi constraints via weak adaptivity. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pp. 5792–5816. PMLR, 2024.

Eleonora Fidelia Chiefari, Francesco Emanuele Stradi, Matteo Castiglioni, and Alberto Marchesi. Online resource allocation with general constraints, 2026. URL https://arxiv.org/abs/2605.10519.

Dragos Florin Ciocan and Vivek F. Farias. Dynamic allocation problems with volatile demand. 2011. URL https://api.semanticscholar.org/CorpusID:17524307.

Nikhil Devanur and Thomas P. Hayes. The adwords problem: Online keyword matching with budgeted bidders under random permutations. In *ACM Conference on Electronic Commerce*. Association for Computing Machinery, Inc., January 2009.

Nikhil R. Devanur, Kamal Jain, Balasubramanian Sivan, and Christopher A. Wilkens. Near optimal online algorithms and fast approximation algorithms for resource allocation problems. *CoRR*, abs/1903.03944, 2019. URL http://arxiv.org/abs/1903.03944.

Hossein Esfandiari, Nitish Korula, and Vahab S. Mirrokni. Online allocation with traffic spikes: Mixing adversarial and stochastic models. *CoRR*, abs/1711.05764, 2017. URL http://arxiv.org/abs/1711.05764.

Jon Feldman, Monika Henzinger, Nitish Korula, Vahab S. Mirrokni, and Clifford Stein. Online stochastic ad allocation: Efficiency and fairness. *CoRR*, abs/1001.5076, 2010. URL http://arxiv.org/abs/1001.5076.

Zhe Feng, Swati Padmanabhan, and Di Wang. Online bidding algorithms for return-on-spend constrained advertisers, 2023. URL https://arxiv.org/abs/2208.13713.

Nicole Immorlica, Karthik Abinav Sankararaman, Robert Schapire, and Aleksandrs Slivkins. Adversarial bandits with knapsacks, 2023. URL https://arxiv.org/abs/1811.11881.

Rodolphe Jenatton, Jim Huang, and Cédric Archambeau. Adaptive algorithms for online convex optimization with long-term constraints, 2015. URL https://arxiv.org/abs/1512.07422.

Thomas Kesselheim and Sahil Singla. Online learning with vector costs and bandits with knapsacks. *CoRR*, abs/2010.07346, 2020. URL https://arxiv.org/abs/2010.07346.

Thomas Kesselheim, Klaus Radke, Andreas Tönnis, and Berthold Vöcking. Primal beats dual on online packing lps in the random-order model. *CoRR*, abs/1311.2578, 2013. URL http://arxiv.org/abs/1311.2578.

Raunak Kumar and Robert Kleinberg. Non-monotonic resource utilization in the bandits with knapsacks problem, 2022. URL https://arxiv.org/abs/2209.12013.

Xiaocheng Li, Chunlin Sun, and Yinyu Ye. Simple and fast algorithm for binary integer and online linear programming. *CoRR*, abs/2003.02513, 2020. URL https://arxiv.org/abs/2003.02513.

Mehrdad Mahdavi, Rong Jin, and Tianbao Yang. Trading regret for efficiency: Online convex optimization with long term constraints, 2012. URL https://arxiv.org/abs/1111.6082.

Shie Mannor, John N. Tsitsiklis, and Jia Yuan Yu. Online learning with sample path constraints. *Journal of Machine Learning Research*, 10(20):569–590, 2009. URL http://jmlr.org/papers/v10/mannor09a.html.

Aranyak Mehta, Amin Saberi, Umesh Vazirani, and Vijay Vazirani. Adwords and generalized online matching. *J. ACM*, 54(5):22–es, October 2007. ISSN 0004-5411. doi: 10.1145/1284320.1284321. URL https://doi.org/10.1145/1284320.1284321.

Francesco Emanuele Stradi, Matteo Castiglioni, Alberto Marchesi, Nicola Gatti, and Christian Kroer. No-regret learning under adversarial resource constraints: A spending plan is all you need! In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Rui Sun, Xinshang Wang, and Zijie Zhou. Near-optimal primal-dual algorithms for quantity-based network revenue management, 2022. URL https://arxiv.org/abs/2011.06327.

Kalyan T. Talluri and Garrett J. van Ryzin. *The Theory and Practice of Revenue Management*. International Series in Operations Research & Management Science. Springer New York, NY, 2004. ISBN 978-1-4020-7701-2. doi: 10.1007/b139000.

Hao Yu, Michael J. Neely, and Xiaohan Wei. Online convex optimization with stochastic constraints, 2017. URL https://arxiv.org/abs/1708.03741.

Yunhong Zhou, Deeparnab Chakrabarty, and Rajan Lukose. Budget constrained bidding in keyword auctions and online knapsack problems. In Christos Papadimitriou and Shuzhong Zhang (eds.), *Internet and Network Economics*, pp. 566–576, Berlin, Heidelberg, 2008. Springer Berlin Heidelberg. ISBN 978-3-540-92185-1.

A Additional Related Works

A line of work related to our setting studies decision making under more general long-term constraints, which unlike budget constraints are not hard, i.e., they can suffer temporary violations over the time horizon. In this framework, the quantity to control during the learning process is the cumulative constraint violation, which can receive negative per-round contribution: an action may satisfy a constraint with slack, compensating for violations incurred in earlier rounds. Despite showing this "compensating" feature, this setting fundamentally differs from the replenishable ORA setting. For general constraints, no single action is guaranteed a priori to offset violations across all constraints simultaneously. In our setting, a single, known action $\varnothing$ provides a contribution of at most $-\beta$ for every input tuple. (Mahdavi et al., 2012) and (Jenatton et al., 2015) were among the first to show that sublinear regret and constraint violation can be attained simultaneously under fixed constraints. (Mannor et al., 2009), (Yu et al., 2017) and (Sun et al., 2022) later extended this guarantee for time-varying constraints. A closely related model, *Bandits with Knapsacks* (BwK) was introduced by (Badanidiyuru et al., 2013) for budget constraints and was sharpened for the stochastic setting by several works, such as (Agrawal & Devanur, 2019). The results were extended to adversarial and non-stationary environments by (Immorlica et al., 2023), (Kesselheim & Singla, 2020) and (Castiglioni et al., 2022). Most relevant to our discussion, (Bernasconi et al., 2024a) and (Castiglioni et al., 2024) generalize BwK beyond budgets accommodating arbitrary long-term constraints.

B Omitted Proofs and Lemmas

B.1 Omitted Proofs and Lemmas of Section 4

Lemma B.1. *Let $\mathcal{T}_{forced}$ be the set of rounds in which the budget condition forces Algorithm 1 to play $\varnothing$, and let $\mathcal{T}_{active} = [T] \setminus \mathcal{T}_{forced}$. Let the dual algorithm be OGD on $\mathbb{R}^m_{\geq 0}$ with learning rate $\eta > 0$, with no updates during forced rounds. Then, for any $\boldsymbol{\mu} \in \mathbb{R}^m_{\geq 0}$ and $[t_1, t_2] \subseteq [T]$, it holds:*

$$\sum_{t \in [t_1,t_2] \cap \mathcal{T}_{active}} \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t - \boldsymbol{\rho} \rangle \geq \sum_{t \in [t_1,t_2] \cap \mathcal{T}_{active}} \langle \boldsymbol{\mu}, \boldsymbol{b}_t - \boldsymbol{\rho} \rangle - \frac{\|\boldsymbol{\lambda}_{t_1} - \boldsymbol{\mu}\|_2^2}{2\eta} - 2\eta T m,$$

where $\boldsymbol{b}_t = \boldsymbol{b}_t(x_t)$.

Proof. For each active round, by the properties of Euclidean projection onto a convex set,

$$\begin{aligned}
\|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\mu}\|_2^2 &= \|\Pi_{\mathbb{R}^m_{\geq 0}}(\boldsymbol{\lambda}_t + \eta(\boldsymbol{b}_t - \boldsymbol{\rho})) - \boldsymbol{\mu}\|_2^2 \\
&\leq \|\boldsymbol{\lambda}_t + \eta(\boldsymbol{b}_t - \boldsymbol{\rho}) - \boldsymbol{\mu}\|_2^2 \\
&= \|\boldsymbol{\lambda}_t - \boldsymbol{\mu}\|_2^2 + \eta^2 \|\boldsymbol{b}_t - \boldsymbol{\rho}\|_2^2 + 2\eta \langle \boldsymbol{\lambda}_t - \boldsymbol{\mu}, \boldsymbol{b}_t - \boldsymbol{\rho} \rangle.
\end{aligned}$$

Rearranging the terms:

$$\begin{aligned}
2\eta \langle \boldsymbol{\mu} - \boldsymbol{\lambda}_t, \boldsymbol{b}_t - \boldsymbol{\rho} \rangle &\leq \|\boldsymbol{\lambda}_t - \boldsymbol{\mu}\|_2^2 - \|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\mu}\|_2^2 + \eta^2 \|\boldsymbol{b}_t - \boldsymbol{\rho}\|_2^2 \\
\langle \boldsymbol{\mu} - \boldsymbol{\lambda}_t, \boldsymbol{b}_t - \boldsymbol{\rho} \rangle &\leq \frac{1}{2\eta} \|\boldsymbol{\lambda}_t - \boldsymbol{\mu}\|_2^2 - \frac{1}{2\eta} \|\boldsymbol{\lambda}_{t+1} - \boldsymbol{\mu}\|_2^2 + \frac{\eta}{2} \|\boldsymbol{b}_t - \boldsymbol{\rho}\|_2^2.
\end{aligned}$$

Summing over active rounds, the squared-distance terms telescope since the dual variables remain unchanged during forced rounds. Thus, we obtain:

$$\begin{aligned}
\sum_{t \in [t_1,t_2] \cap \mathcal{T}_{\text{active}}} \langle \boldsymbol{\mu} - \boldsymbol{\lambda}_t, \boldsymbol{b}_t - \boldsymbol{\rho} \rangle &\leq \frac{1}{2\eta} \|\boldsymbol{\lambda}_{t_1} - \boldsymbol{\mu}\|_2^2 - \frac{1}{2\eta} \|\boldsymbol{\lambda}_{t_2+1} - \boldsymbol{\mu}\|_2^2 + \frac{\eta}{2} \sum_{t \in [t_1,t_2] \cap \mathcal{T}_{\text{active}}} \|\boldsymbol{b}_t - \boldsymbol{\rho}\|_2^2 \\
&\leq \frac{1}{2\eta} \|\boldsymbol{\lambda}_{t_1} - \boldsymbol{\mu}\|_2^2 - \frac{1}{2\eta} \|\boldsymbol{\lambda}_{t_2+1} - \boldsymbol{\mu}\|_2^2 + \frac{\eta}{2} T m (1+\rho)^2 \\
&\leq \frac{1}{2\eta} \|\boldsymbol{\lambda}_{t_1} - \boldsymbol{\mu}\|_2^2 + 2\eta T m,
\end{aligned}$$

where in the last step we used the fact that $\rho < 1$, yielding $(1+\rho)^2 \leq 4$. Rearranging the terms gives the stated bound. □

Lemma B.2. *Let $\mathcal{T}_{forced} \subseteq [T]$ be the set of rounds in which the algorithm is forced to play the void action—due to the triggering of the budget condition—and $\mathcal{T}_{active} = [T] \setminus \mathcal{T}_{forced}$ be the set of active rounds, in which the algorithm maximizes the Lagrangian. Assume $\mathcal{T}_{forced} \neq \varnothing$. Let $t_\varnothing \in \mathcal{T}_{forced}$ be the last forced round, and $j \in [m]$ the scarcest resource in round $t_\varnothing$. Formally:*

$$t_\varnothing \coloneqq \max \mathcal{T}_{forced}, \qquad j \coloneqq \arg\min_{i \in [m]} B_{t_\varnothing, i}.$$

Then the cumulative consumption of resource j during the active rounds preceding $t_\varnothing$ can be bounded as:

$$\sum_{t \in \mathcal{T}_{active} : t < t_\varnothing} b_{t,j}(x_t) \geq \rho T - 2 + \beta |\mathcal{T}_{forced}|.$$

Proof. From the budget update rule, for any $t \in [T]$ we have $B_{t+1,j} = B_{t,j} - b_{t,j}(x_t)$. Rearranging and summing up to round $t_\varnothing - 1$, we get:

$$\begin{aligned} \sum_{t \in [t_\varnothing - 1]} b_{t,j}(x_t) &= \sum_{t \in [t_\varnothing - 1]} (B_{t,j} - B_{t+1,j}) \\ &= B_{1,j} - B_{t_\varnothing, j} \\ &> \rho T - 1, \end{aligned} \tag{1}$$

where Inequality (1) follows from the initial budget $B_{1,j} = \rho T$ and the fact that $B_{t_\varnothing, j} < 1$ since round $t_\varnothing$ triggered a replenishment.

Exploiting this result, we can write:

$$\begin{aligned} \sum_{t \in \mathcal{T}_{\text{active}} : t < t_\varnothing} b_{t,j}(x_t) &= \sum_{t \in [t_\varnothing - 1]} b_{t,j}(x_t) - \sum_{t \in \mathcal{T}_{\text{forced}} \setminus \{t_\varnothing\}} b_{t,j}(x_t) \\ &\geq \rho T - 1 - \sum_{t \in \mathcal{T}_{\text{forced}} \setminus \{t_\varnothing\}} b_{t,j}(\varnothing) \qquad (2) \\ &\geq \rho T - 1 + \beta(|\mathcal{T}_{\text{forced}}| - 1) \qquad (3) \\ &\geq \rho T - 2 + \beta |\mathcal{T}_{\text{forced}}|, \end{aligned}$$

where Inequality (2) follows from Inequality (1) and Inequality (3) holds by definition of the void action $\varnothing$. The last step is due to the assumption $\beta \leq 1$. This concludes the proof. □

Lemma B.3. *For each $t \in \mathcal{T}_{active}$, Algorithm 1 with learning rate $\eta > 0$ guarantees:*

$$\|\boldsymbol{\lambda}_{t+1}\|_2^2 \leq \|\boldsymbol{\lambda}_t\|_2^2 + 2\eta\big(1 - (\rho + \beta)\|\boldsymbol{\lambda}_t\|_1\big) + 4\eta^2 m.$$

Proof. Since x_t maximizes the Lagrangian, we have,

$$\begin{aligned} f_t(x_t) - \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho} \rangle &\geq -\langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(\varnothing) - \boldsymbol{\rho} \rangle \\ &\geq (\rho + \beta)\|\boldsymbol{\lambda}_t\|_1. \end{aligned}$$

Thus, since $f_t(x_t) \leq 1$, we obtain:

$$\langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho} \rangle \leq 1 - (\rho + \beta)\|\boldsymbol{\lambda}_t\|_1.$$

By non-expansiveness of the projection and $\|\boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\|_2^2 \leq 4m$, we have:

$$\begin{aligned} \|\boldsymbol{\lambda}_{t+1}\|_2^2 &\leq \|\boldsymbol{\lambda}_t\|_2^2 + 2\eta\langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho} \rangle + \eta^2 \|\boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\|_2^2 \\ &\leq \|\boldsymbol{\lambda}_t\|_2^2 + 2\eta\big(1 - (\rho + \beta)\|\boldsymbol{\lambda}_t\|_1\big) + 4\eta^2 m. \end{aligned}$$

This concludes the proof. □

Lemma B.4. *Let the dual variables be generated by Algorithm 1 with $0 < \eta \leq 1/2m$. It holds*

$$\sum_{t \in \mathcal{T}_{active}} \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho} \rangle \geq |\mathcal{T}_{forced}| - \frac{2}{\rho + \beta} - \frac{9m}{\eta(\rho + \beta)^2} - 4\eta T m.$$

Proof. If $\mathcal{T}_{\text{forced}} = \varnothing$, Lemma B.1 with $\boldsymbol{\mu} = \mathbf{0}$ and $\boldsymbol{\lambda}_1 = \mathbf{0}$ gives a lower bound of $-2\eta T m$, which implies the claim. Thus, assume $\mathcal{T}_{\text{forced}} \neq \varnothing$. We provide a lower bound on the term $\sum_{t\in\mathcal{T}_{\text{active}}} \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle$ employing Lemma B.1 with different comparators for the rounds before and after $t_\varnothing$. Since the dual multipliers are not updated during forced rounds, Lemma B.1 applies to the active rounds on either side of $t_\varnothing$. If either set is empty, the corresponding OGD bound holds trivially.

- $(t < t_\varnothing)$ Let j be the resource identified in Lemma B.2. We choose the comparator $\boldsymbol{\mu}$ such that $\mu_j = \frac{1}{\rho+\beta}$ and $\mu_i = 0$ for each $i \neq j$. We obtain:

$$\begin{aligned}
\sum_{t\in\mathcal{T}_{\text{active}}:t<t_\varnothing} \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle &\geq \sum_{t\in\mathcal{T}_{\text{active}}:t<t_\varnothing} \mu_j(b_{t,j}(x_t) - \rho) \\
&\quad - \frac{1}{2\eta}\|\boldsymbol{\lambda}_1 - \boldsymbol{\mu}\|_2^2 - 2\eta T m \\
&\geq \frac{1}{\rho+\beta}\left(\sum_{t\in\mathcal{T}_{\text{active}}:t<t_\varnothing} b_{t,j}(x_t) - \sum_{t\in\mathcal{T}_{\text{active}}:t<t_\varnothing} \rho\right) + \\
&\quad - \frac{1}{2\eta(\rho+\beta)^2} - 2\eta T m \\
&\geq \frac{1}{\rho+\beta}\left(\rho T - 2 + \beta|\mathcal{T}_{\text{forced}}| - \sum_{t\in\mathcal{T}_{\text{active}}:t<t_\varnothing} \rho\right) + \\
&\quad - \frac{1}{2\eta(\rho+\beta)^2} - 2\eta T m, \quad (4)
\end{aligned}$$

  where Inequality (4) follows from Lemma B.2. Notice that the total horizon budget ρT can be decomposed across the disjoint sets as $\rho T = \rho|\mathcal{T}_{\text{forced}}| + \sum_{t\in\mathcal{T}_{\text{active}}:t<t_\varnothing} \rho + \sum_{t\in\mathcal{T}_{\text{active}}:t>t_\varnothing} \rho$. Substituting this expansion we obtain:

$$\begin{aligned}
&= \frac{1}{\rho+\beta}\left(\rho|\mathcal{T}_{\text{forced}}| - 2 + \beta|\mathcal{T}_{\text{forced}}| + \sum_{t\in\mathcal{T}_{\text{active}}:t>t_\varnothing} \rho\right) + \\
&\quad - \frac{1}{2\eta(\rho+\beta)^2} - 2\eta T m \\
&= \frac{\sum_{t\in\mathcal{T}_{\text{active}}:t>t_\varnothing} \rho}{\rho+\beta} - \frac{2}{\rho+\beta} + |\mathcal{T}_{\text{forced}}| \\
&\quad - \frac{1}{2\eta(\rho+\beta)^2} - 2\eta T m \\
&\geq -\frac{2}{\rho+\beta} + |\mathcal{T}_{\text{forced}}| - \frac{1}{2\eta(\rho+\beta)^2} - 2\eta T m, \quad (5)
\end{aligned}$$

  where Inequality (5) holds because $\sum \rho \geq 0$.

- $(t > t_\varnothing)$ Since $t_\varnothing$ is the last round in which a replenishment is required, in the phase following $t_\varnothing$ the budget condition never forces the void action again. We choose the comparator $\boldsymbol{\mu} = \mathbf{0}$ to obtain:

$$\begin{aligned}
\sum_{t\in\mathcal{T}_{\text{active}}:t>t_\varnothing} \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle &\geq 0 - \frac{1}{2\eta}\|\boldsymbol{\lambda}_{t_\varnothing+1}\|_2^2 - 2\eta T m \\
&\geq -\frac{1}{2\eta}\left(\frac{4\sqrt{m}}{\rho+\beta}\right)^2 - 2\eta T m \quad (6) \\
&\geq -\frac{16m}{2\eta(\rho+\beta)^2} - 2\eta T m, \quad (7)
\end{aligned}$$

  where Inequality (6) holds by Lemma 4.1.

Putting everything together:

$$\begin{aligned}\sum_{t\in\mathcal{T}_{\text{active}}}\langle\boldsymbol{\lambda}_t,\boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\rangle &= \sum_{t\in\mathcal{T}_{\text{active}}:t<t_\varnothing}\langle\boldsymbol{\lambda}_t,\boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\rangle \\ &\quad+\sum_{t\in\mathcal{T}_{\text{active}}:t>t_\varnothing}\langle\boldsymbol{\lambda}_t,\boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\rangle \\ &\geq -\left(\frac{2}{\rho+\beta}-|\mathcal{T}_{\text{forced}}|+\frac{1}{2\eta(\rho+\beta)^2}+2\eta Tm\right) \\ &\quad-\left(\frac{16m}{2\eta(\rho+\beta)^2}+2\eta Tm\right) \\ &\geq |\mathcal{T}_{\text{forced}}|-\frac{2}{\rho+\beta}-\frac{9m}{\eta(\rho+\beta)^2}-4\eta Tm,\end{aligned}$$

which concludes the proof. □

Lemma 4.1. *In both the stochastic and the adversarial setting, Algorithm 1 with $0<\eta\leq 1/2m$ guarantees that the ℓ_1-norm of the dual variables is bounded for all $t\in[T+1]$ as:*

$$\|\boldsymbol{\lambda}_t\|_1\leq\frac{4\sqrt{m}}{\rho+\beta}.$$

Proof. We prove by induction that $\|\boldsymbol{\lambda}_t\|_2\leq 4/\rho+\beta$ for all $t\in[T+1]$. The claim holds at $t=1$ since $\boldsymbol{\lambda}_1=\mathbf{0}$. During forced rounds, the dual variables remain unchanged.

Consider an active round t. If $\|\boldsymbol{\lambda}_t\|_1\geq 2/\rho+\beta$, Lemma B.3 gives:

$$\|\boldsymbol{\lambda}_{t+1}\|_2^2\leq\|\boldsymbol{\lambda}_t\|_2^2-2\eta+4\eta^2m\leq\|\boldsymbol{\lambda}_t\|_2^2,$$

where the last inequality follows from $\eta\leq 1/2m$. Otherwise, by non-expansiveness of the projection, we obtain:

$$\begin{aligned}\|\boldsymbol{\lambda}_{t+1}\|_2 &\leq \|\boldsymbol{\lambda}_t\|_2+\eta\|\boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\|_2 \\ &\leq \|\boldsymbol{\lambda}_t\|_1+2\eta\sqrt{m} \\ &\leq \frac{2}{\rho+\beta}+\frac{1}{\sqrt{m}} \\ &\leq \frac{4}{\rho+\beta},\end{aligned}$$

since $\rho+\beta\leq 2$ and $m\geq 1$. Thus, the induction is complete. Finally,

$$\|\boldsymbol{\lambda}_t\|_1\leq\sqrt{m}\|\boldsymbol{\lambda}_t\|_2\leq\frac{4\sqrt{m}}{\rho+\beta},$$

which concludes the proof. □

Theorem 4.2. *In the stochastic setting, assuming the replenishment parameter β is known, Algorithm 1 with learning rate $\eta:=\frac{3}{2(\rho+\beta)}\sqrt{\frac{1}{T}}$ guarantees:*

$$R_T\leq\frac{2+12m\sqrt{T}}{\rho+\beta}.$$

If β is unknown, Algorithm 1 with learning rate $\eta:=\frac{3}{2\rho}\sqrt{\frac{1}{T}}$ guarantees:

$$R_T\leq\frac{2}{\rho+\beta}+6m\left(\frac{\rho}{(\rho+\beta)^2}+\frac{1}{\rho}\right)\sqrt{T}.$$

Proof. First, consider the case $\eta>1/2m$. If β is known, the prescribed learning rate implies $\sqrt{T}<3m/\rho+\beta$. Thus,

$$R_T\leq T<\frac{3m\sqrt{T}}{\rho+\beta}\leq\frac{2+12m\sqrt{T}}{\rho+\beta}.$$

If β is unknown, we similarly have $\sqrt{T} < {}^{3m}/_{\rho}$ and

$$R_T \leq T < \frac{3m\sqrt{T}}{\rho} \leq \frac{2}{\rho+\beta} + 6m\left(\frac{\rho}{(\rho+\beta)^2} + \frac{1}{\rho}\right)\sqrt{T}.$$

Therefore, both claims hold in this case. For the remainder of the proof, assume $0 < \eta \leq {}^{1}/_{2m}$.

First, we provide an upper bound on the hindsight stochastic optimum. We define the expected Lagrangian evaluated with respect to the input distribution as:

$$\bar{\mathcal{L}}_{f,\boldsymbol{b}}(\boldsymbol{\lambda}|\mathcal{P}) := \mathbb{E}_{(f,\boldsymbol{b})\sim\mathcal{P}}\left[\max_{x\in\mathcal{X}} \mathcal{L}_{f,\boldsymbol{b}}(x,\boldsymbol{\lambda})\right].$$

By weak duality, for every fixed $\boldsymbol{\lambda} \in \mathbb{R}^m_{\geq 0}$ and every input sequence $\boldsymbol{\gamma}$, we have:

$$\text{OPT}(\boldsymbol{\gamma}) \leq \sum_{t\in[T]} \max_{x\in\mathcal{X}} \mathcal{L}_{f_t,\boldsymbol{b}_t}(x,\boldsymbol{\lambda}).$$

Taking expectations gives $\text{OPT}_{\text{stoc}} \leq T\bar{\mathcal{L}}_{f,\boldsymbol{b}}(\boldsymbol{\lambda} \mid \mathcal{P})$ for every $\boldsymbol{\lambda} \geq \mathbf{0}$. Notice that this is an inequality between deterministic functions of $\boldsymbol{\lambda}$. Thus, it also holds pointwise when evaluated at a random dual vector. Let $\bar{\boldsymbol{\lambda}}_{\mathcal{T}_{\text{active}}} := \frac{1}{|\mathcal{T}_{\text{active}}|}\sum_{t\in\mathcal{T}_{\text{active}}} \boldsymbol{\lambda}_t$ when $\mathcal{T}_{\text{active}} \neq \varnothing$, and let $\bar{\boldsymbol{\lambda}}_{\mathcal{T}_{\text{active}}} = \mathbf{0}$ otherwise. Then,

$$\begin{aligned}\text{OPT}_{\text{stoc}} &= \mathbb{E}_{\boldsymbol{\gamma}}\left[\frac{|\mathcal{T}_{\text{active}}|}{T}\text{OPT}_{\text{stoc}}\right] + \mathbb{E}_{\boldsymbol{\gamma}}\left[\frac{|\mathcal{T}_{\text{forced}}|}{T}\text{OPT}_{\text{stoc}}\right] \\ &\leq \mathbb{E}_{\boldsymbol{\gamma}}\left[|\mathcal{T}_{\text{active}}|\bar{\mathcal{L}}_{f,\boldsymbol{b}}(\bar{\boldsymbol{\lambda}}_{\mathcal{T}_{\text{active}}} \mid \mathcal{P})\right] + \mathbb{E}_{\boldsymbol{\gamma}}[|\mathcal{T}_{\text{forced}}|], \end{aligned} \tag{8}$$

where we used that rewards are in $[0,1]$.

During active rounds, Algorithm 1 maximizes the instantaneous Lagrangian. Therefore, we can write the instantaneous reward as:

$$f_t(x_t) = \mathcal{L}_{f_t,\boldsymbol{b}_t}(x_t,\boldsymbol{\lambda}_t) + \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle.$$

Let $\Gamma_t = (\gamma_1, ..., \gamma_t)$ denote the history of observed inputs up to time t. Since $\boldsymbol{\lambda}_t$ and $\mathbb{I}\{t \in \mathcal{T}_{\text{active}}\}$ are Γ_{t-1}-measurable, taking the conditional expectation during active rounds we obtain:

$$\begin{aligned}\mathbb{E}[f_t(x_t)\cdot\mathbb{I}\{t\in\mathcal{T}_{\text{active}}\} \mid \Gamma_{t-1}] &= \mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}\mathbb{E}[\mathcal{L}_{f_t,\boldsymbol{b}_t}(x_t,\boldsymbol{\lambda}_t) \mid \Gamma_{t-1}] + \\ &\quad + \mathbb{E}[\mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}\langle\boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\rangle \mid \Gamma_{t-1}] \\ &= \mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}\bar{\mathcal{L}}_{f,\boldsymbol{b}}(\boldsymbol{\lambda}_t \mid \mathcal{P}) + \\ &\quad + \mathbb{E}[\mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}\langle\boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\rangle \mid \Gamma_{t-1}]. \end{aligned} \tag{9}$$

Thus, summing Equation (9) over t and taking the expectation on both sides, we have:

$$\begin{aligned}\sum_{t\in[T]} \mathbb{E}\left[\mathbb{E}[f_t(x_t)\cdot\mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}|\Gamma_{t-1}]\right] &= \sum_{t\in[T]} \mathbb{E}[\mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}\bar{\mathcal{L}}_{f,\boldsymbol{b}}(\boldsymbol{\lambda}_t|\mathcal{P}) \\ &\quad + \mathbb{E}[\mathbb{I}\{t\in\mathcal{T}_{\text{active}}\}\langle\boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\rangle|\Gamma_{t-1}]]\end{aligned}$$

By Tower Rule ($\mathbb{E}[\mathbb{E}[\cdot|\Gamma_{t-1}]] = \mathbb{E}[\cdot]$) and definition of $\mathbb{I}\{t \in \mathcal{T}_{\text{active}}\}$ we obtain:

$$\mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}} f_t(x_t)\right] = \mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}} \bar{\mathcal{L}}_{f,\boldsymbol{b}}(\boldsymbol{\lambda}_t|\mathcal{P})\right] + \mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}} \langle\boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\rangle\right].$$

By convexity of the expected Lagrangian, with both sides of Jensen's inequality equal to zero when there are no active rounds, we can write:

$$\begin{aligned}\mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}} f_t(x_t)\right] &= \mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}} \bar{\mathcal{L}}_{f,\boldsymbol{b}}(\boldsymbol{\lambda}_t|\mathcal{P})\right] + \mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}} \langle\boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\rangle\right] \\ &\geq \mathbb{E}\left[|\mathcal{T}_{\text{active}}|\bar{\mathcal{L}}_{f,\boldsymbol{b}}(\bar{\boldsymbol{\lambda}}_{\mathcal{T}_{\text{active}}}|\mathcal{P})\right] + \mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}} \langle\boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t)-\boldsymbol{\rho}\rangle\right].\end{aligned}$$

Combining this result with Inequality (8), we can bound the regret as:

$$\begin{aligned}
R_T &= \mathrm{OPT}_{stoc} - \mathbb{E}_{\boldsymbol{\gamma}}\left[\sum_{t\in\mathcal{T}_{\text{active}}} f_t(x_t)\right] \\
&\le \mathbb{E}_{\boldsymbol{\gamma}}\left[|\mathcal{T}_{\text{active}}|\cdot \bar{\mathcal{L}}_{f,\boldsymbol{b}}(\bar{\boldsymbol{\lambda}}_{\mathcal{T}_{\text{active}}}|\mathcal{P})\right] + \mathbb{E}_{\boldsymbol{\gamma}}[|\mathcal{T}_{\text{forced}}|] \\
&\quad - \mathbb{E}\left[|\mathcal{T}_{\text{active}}|\bar{\mathcal{L}}_{f,\boldsymbol{b}}(\bar{\boldsymbol{\lambda}}_{\mathcal{T}_{\text{active}}}|\mathcal{P})\right] \\
&\quad - \mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}} \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle\right] \\
&\le \mathbb{E}_{\boldsymbol{\gamma}}[|\mathcal{T}_{\text{forced}}|] - \mathbb{E}\left[\sum_{t\in\mathcal{T}_{\text{active}}} \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle\right] \\
&\le \mathbb{E}_{\boldsymbol{\gamma}}[|\mathcal{T}_{\text{forced}}|] + \mathbb{E}\left[-|\mathcal{T}_{\text{forced}}| + \frac{2}{\rho+\beta} + \frac{9m}{\eta(\rho+\beta)^2} + 4\eta T m\right] \qquad (10) \\
&\le \frac{2}{\rho+\beta} + \frac{9m}{\eta(\rho+\beta)^2} + 4\eta T m.
\end{aligned}$$

where Inequality (10) holds by Lemma B.4. To obtain the stated bounds, we tune the learning rate η depending on whether the replenishment parameter β is known to the algorithm:

- Setting 1 (β is known): $\eta := \frac{3}{2(\rho+\beta)}\sqrt{\frac{1}{T}}$. The regret is bounded by:

$$\begin{aligned}
R_T &\le \frac{2}{\rho+\beta} + \frac{9m}{\eta(\rho+\beta)^2} + 4\eta T m \\
&\le \frac{2+12m\sqrt{T}}{\rho+\beta}.
\end{aligned}$$

- Setting 2 (β is unknown): If the algorithm does not know the value of β, the Slater's margin can be lower bounded by ρ, since $\beta \ge 0$. Consequently, we can set $\eta := \frac{3}{2(\rho)}\frac{1}{\sqrt{T}}$. The regret is bounded by:

$$\begin{aligned}
R_T &\le \frac{2}{\rho+\beta} + \frac{9m}{\eta(\rho+\beta)^2} + 4\eta T m \\
&\le \frac{2}{\rho+\beta} + \frac{6m\rho}{(\rho+\beta)^2}\sqrt{T} + \frac{6m}{\rho}\sqrt{T} \\
&= \frac{2}{\rho+\beta} + 6m\left(\frac{\rho}{(\rho+\beta)^2} + \frac{1}{\rho}\right)\sqrt{T}.
\end{aligned}$$

This concludes the proof. □

Theorem 4.4. *In the adversarial setting, assuming the replenishment parameter β is known, Algorithm 1 with learning rate $\eta := \frac{3}{2(\rho+\beta)}\sqrt{\frac{1}{T}}$ guarantees:*

$$\alpha\text{-}R_T \le \frac{2+12m\sqrt{T}}{\rho+\beta}.$$

If β is unknown, Algorithm 1 with learning rate $\eta := \frac{3}{2\rho}\sqrt{\frac{1}{T}}$ guarantees:

$$\alpha\text{-}R_T \le \frac{2}{\rho+\beta} + 6m\left(\frac{\rho}{(\rho+\beta)^2} + \frac{1}{\rho}\right)\sqrt{T}.$$

Proof. If $\eta > 1/2m$, both claims follow from $\alpha\text{-}R_T \le T$ by the same argument as in the proof of Theorem 4.2. Thus, assume $0 < \eta \le 1/2m$. Let $(x_t^*)_{t=1}^T$ be the optimal sequence of actions and let

$\alpha := \frac{\rho+\beta}{1+\beta}$. For each round $t \in [T]$, we define $\xi_t^\diamond$ as the strategy that plays x_t^* with probability α and $\varnothing$ with probability $1-\alpha$. By construction, for each $i \in [m]$, it holds:

$$\mathbb{E}_{x\sim\xi_t^\diamond}[b_{t,i}(x)] = \alpha \cdot b_{t,i}(x_t^*) + (1-\alpha) b_{t,i}(\varnothing) \leq \alpha \cdot 1 + (1-\alpha)(-\beta) = \rho. \tag{11}$$

Moreover, since rewards are non-negative, it holds:

$$\mathbb{E}_{x\sim\xi_t^\diamond}[f_t(x)] = \alpha \cdot f_t(x_t^*) + (1-\alpha) f_t(\varnothing) \geq \alpha \cdot f_t(x_t^*). \tag{12}$$

Exploiting these properties, we can bound the α-regret as:

$$\begin{aligned}
\alpha\text{-}R_T &= \sum_{t\in[T]} (\alpha f_t(x_t^*) - f_t(x_t)) \\
&\leq \sum_{t\in[T]} (\mathbb{E}_{x\sim\xi_t^\diamond}[f_t(x)] - f_t(x_t)) && (13)\\
&= \sum_{t\in\mathcal{T}_{\text{active}}} (\mathbb{E}_{x\sim\xi_t^\diamond}[f_t(x)] - f_t(x_t)) + \sum_{t\in\mathcal{T}_{\text{forced}}} (\mathbb{E}_{x\sim\xi_t^\diamond}[f_t(x)] - f_t(\varnothing)) \\
&\leq \sum_{t\in\mathcal{T}_{\text{active}}} (\mathbb{E}_{x\sim\xi_t^\diamond}[f_t(x)] - f_t(x_t)) + |\mathcal{T}_{\text{forced}}| \\
&\leq \sum_{t\in\mathcal{T}_{\text{active}}} \left(\langle \boldsymbol{\lambda}_t, \mathbb{E}_{x\sim\xi_t^\diamond}[\boldsymbol{b}_t(x) - \boldsymbol{\rho}] - (\boldsymbol{b}_t(x_t) - \boldsymbol{\rho})\rangle\right) + |\mathcal{T}_{\text{forced}}| && (14)\\
&\leq -\sum_{t\in\mathcal{T}_{\text{active}}} \langle \boldsymbol{\lambda}_t, \boldsymbol{b}_t(x_t) - \boldsymbol{\rho}\rangle + |\mathcal{T}_{\text{forced}}| && (15)\\
&\leq -|\mathcal{T}_{\text{forced}}| + \frac{2}{\rho+\beta} + \frac{9m}{\eta(\rho+\beta)^2} + 4\eta T m + |\mathcal{T}_{\text{forced}}| && (16)\\
&= \frac{2}{\rho+\beta} + \frac{9m}{\eta(\rho+\beta)^2} + 4\eta T m,
\end{aligned}$$

where Inequality (13) follows from Inequality (12), Inequality (14) holds since x_t deterministically maximizes the instantaneous Lagrangian over $\mathcal{X}$, Inequality (15) follows from Inequality (11) and Inequality (16) holds by Lemma B.4. The proof is concluded by tuning η according to the reasoning in Theorem 4.2. □