The Symmetry between Arms and Knapsacks: A Primal-Dual Approach for Bandits with Knapsacks

Xiaocheng Li[1] Chunlin Sun[2] Yinyu Ye[2]

Abstract

In this paper, we study the bandits with knapsacks (BwK) problem and develop a primal-dual based algorithm that achieves a problem-dependent logarithmic regret bound. The BwK problem extends the multi-arm bandit (MAB) problem to model the resource consumption associated with playing each arm, and the existing BwK literature has been mainly focused on deriving asymptotically optimal distribution-free regret bounds. We first study the primal and dual linear programs underlying the BwK problem. From this primal-dual perspective, we discover symmetry between arms and knapsacks, and then propose a new notion of sub-optimality measure for the BwK problem. The sub-optimality measure highlights the important role of knapsacks in determining algorithm regret and inspires the design of our two-phase algorithm. In the first phase, the algorithm identifies the optimal arms and the binding knapsacks, and in the second phase, it exhausts the binding knapsacks via playing the optimal arms through an adaptive procedure. Our regret upper bound involves the proposed sub-optimality measure and it has a logarithmic dependence on length of horizon T and a polynomial dependence on m (the numbers of arms) and d (the number of knapsacks). To the best of our knowledge, this is the first problem-dependent logarithmic regret bound for solving the general BwK problem.

1. Introduction

The Multi-Armed Bandit (MAB) problem is a problem in which a limited amount of resource must be allocated between competing (alternative) choices in a way that maximizes the expected gain (Gittins et al., 2011). It is a benchmark problem for decision making under uncertainty that has been studied for nearly a century. As a prototypical reinforcement learning problem, MAB problem exemplifies the exploration–exploitation tradeoff dilemma (Weber et al., 1992). The original problem first formulated in its predominant version in (Robbins, 1952), has inspired a recent line of research that considers additional constraints that reflect more accurately the reality of the online decision making process. *Bandits with Knapsacks* (BwK) was introduced by (Badanidiyuru et al., 2013) to allow more general constraints on the decisions across time, in addition to the customary limitation on the time horizon. The BwK problem, as a general framework, encompasses a wide range of applications, including dynamic pricing and revenue management (Besbes & Zeevi, 2012), online advertisement (Mehta et al., 2005), network and routing (Agrawal et al., 2014), etc.

While the existing BwK literature (Badanidiyuru et al., 2013; Agrawal & Devanur, 2014) has derived algorithms that achieve optimal problem-independent regret bounds, a problem-dependent bound that captures the optimal performance of an algorithm on a specific BwK problem instance remains an open question. For the setting of standard MAB problem, the problem-dependent bound has been well understood, and an upper bound with logarithmic dependence on T can be achieved by both UCB-based algorithm (Auer et al., 2002) and Thompson sampling-based algorithm (Agrawal & Goyal, 2012). In this paper, we focus on developing a problem-dependent bound for the BwK problem and identify parameters that characterize the hardness of a BwK problem instance.

Two existing works along this line are (Flajolet & Jaillet, 2015) and (Sankararaman & Slivkins, 2020). The paper (Flajolet & Jaillet, 2015) considers several specific settings for the problem, and for the general BwK problem, it achieves an $O(2^{m+d} \log T)$ regret bound (with other problem-dependent parameters omitted) where m is the number of arms and d is the number of the knapsacks/resource constraints. In addition, the results in (Flajolet & Jaillet, 2015) require the knowledge of some parameters of the problem instance a priori. The recent work (Sankararaman & Slivkins, 2020) considers the BwK problem under the assumptions that there is only one single

[*]Equal contribution [1]Imperial College London, London, United Kingdom [2]Stanford University, California, USA. Correspondence to: Chunlin Sun <chunlin@stanford.edu>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

knapsack/resource constraint and that there is only one single optimal arm. In contrast to these two pieces of work, we consider the problem in its full generality and do not assume any prior knowledge of the problem instance. We will further compare with their results after we present our regret bound.

Specifically, we adopt a primal-dual perspective to study the BwK problem. Our treatment is new in that we highlight the effects of resources/knapsacks on regret from the dual perspective and define the sub-optimality measure based on the primal and dual problems jointly. Specifically, we first derive a generic upper bound that works for all BwK algorithms. The upper bound consists of two elements: (i) the number of times for which a sub-optimal arm is played; (ii) the remaining knapsack resource at the end of the horizon. It emphasizes that the arms and knapsacks are of equal importance in determining the regret of a BwK algorithm. By further exploiting the structure of the primal and dual LPs, we develop a new sub-optimality measure for the BwK problem which can be viewed as a generalization of the sub-optimality measure for the MAB problem first derived in (Lai & Robbins, 1985). The sub-optimality measure accounts for both arms and knapsacks, and it aims to distinguish optimal arms from non-optimal arms, and binding knapsacks from non-binding knapsacks. We use this measure as a key characterization of the hardness of a BwK problem instance.

Inspired by these findings, we propose a two-phase algorithm for the problem. The first phase of our algorithm is elimination-based and its objective is to identify the optimal arms and binding knapsacks of the BwK problem. The second phase of our algorithm utilizes the output of the first phase and it uses the optimal arms to exhaust the remaining resources through an adaptive procedure. Our algorithm and analysis feature for its full generality and we only make a mild assumption on the non-degeneracy of the underlying LP. In addition, the algorithm requires no prior knowledge of the underlying problem instance.

Other related literature: (Agrawal & Devanur, 2015; Agrawal et al., 2016) study the contextual BwK problem where the reward and resource consumption are both linear in a context vector. (Immorlica et al., 2019; Kesselheim & Singla, 2020) study the BwK problem under an adversarial setting. (Ferreira et al., 2018) analyzes Thompson sampling-based algorithms for the BwK problem.

2. Model and Setup

The problem of *bandits with knapsacks* (BwK) was first defined in (Badanidiyuru et al., 2013), and the notations presented here are largely consistent with (Badanidiyuru et al., 2013; Agrawal & Devanur, 2014). Consider a fixed and known finite set of m arms (possible actions) available to the decision maker, henceforth called the algorithm. There are d type of resources and a finite time horizon T, where T is known to the algorithm. In each time step t, the algorithm plays an arm of the m arms, receives reward r_t, and consumes amount $C_{j,t} \in [0,1]$ of each resource $j \in [d]$. The reward r_t and consumption $\boldsymbol{C}_t = (C_{1,t}, ..., C_{d,t})^\top \in \mathbb{R}^d$ are revealed to the algorithm after choosing arm $i_t \in [m]$. The rewards and costs in every round are generated i.i.d. from some unknown fixed underlying distribution. More precisely, there is some fixed but unknown $\boldsymbol{\mu} = (\mu_1, ..., \mu_m)^\top \in \mathbb{R}^m$ and $\boldsymbol{C} = (\boldsymbol{c}_1, ..., \boldsymbol{c}_m) \in \mathbb{R}^{d\times m}$ such that

$$\mathbb{E}[r_t|i_t] = \mu_{i_t},\ \mathbb{E}[\boldsymbol{C}_t|i_t] = \boldsymbol{c}_{i_t}$$

where $\mu_i \in \mathbb{R}$ and $\boldsymbol{c}_i = (c_{1i}, ..., c_{di})^\top \in \mathbb{R}^d$ are the expected reward and the expected resource consumption of arm $i \in [m]$. In the beginning of every time step t, the algorithm needs to pick an arm i_t, using only the history of plays and outcomes until time step $t-1$. There is a hard constraint capacity B_j for the j-th type of resource. The algorithm stops at the earliest time τ when one or more of the constraints is violated, i.e. $\sum_{t=1}^{\tau+1} c_{j,t} > B_j$ for some $j \in [d]$ or if the time horizon ends, if i.e. $\tau \geq T$. Its total reward is given by the sum of rewards in all rounds preceding τ, i.e $\sum_{t=1}^{\tau} r_t$. The goal of the algorithm is to maximize the expected total reward. The values of B_j are known to the algorithm, and without loss of generality, we make the following assumption.

Assumption 1. *We assume $B_j = B = \min_j B_j$ for all $j \in [d]$ (by scaling the consumption matrix $\boldsymbol{C}$). Let $\boldsymbol{B} = (B, ..., B)^\top \in \mathbb{R}^d$. Moreover, we assume the resource capacity $\boldsymbol{B}$ scales linearly with T, i.e., $\boldsymbol{B} = T \cdot \boldsymbol{b} = T \cdot (b, ..., b)^\top \in \mathbb{R}^d$ for some $b > 0$.*

The assumption is mainly for notation simplicity and it will not change the nature of the analysis in this paper. Given that our focus is to derive asymptotic problem-dependent bound, it is natural to have the resource capacity scales linearly with the length of horizon. Throughout this paper, we use bold symbols to denote vectors/matrices and normal symbols to denote scalars.

Furthermore, without loss of generality, a "null" arm is introduced to represent the time constraint (time horizon T). Specifically, let $\mu_m = 0$, $\boldsymbol{c}_m = (b, 0, ..., 0)^\top$, and $c_{1,i} = b$ for all $i \in [m]$. In this way, the first constraint captures the constraint of finite time horizon T and the "null" arm can be played with no reward achieved and with no cost induced to the other factual constraints except for the time constraint.

Regret is defined as the difference in the total reward obtained by the algorithm and OPT, where OPT denotes the total expected reward for the optimal dynamic policy. In this paper, we are interested in the (expected) problem-dependent regret,

$$\text{Regret}_T^{\pi}(\mathcal{P}, \boldsymbol{B}) := \text{OPT} - \mathbb{E}\left[\sum_{t=1}^{\tau} r_t\right]$$

where π denotes the algorithm, and $\mathcal{P}$ encapsulates all the parameters related to the distributions of reward and resource consumption, including $\boldsymbol{\mu}$ and $\boldsymbol{C}$. The expectation is taken with respect to the randomness of the reward and resource consumption.

Consider the following linear program:

$$\begin{aligned} \text{OPT}_{\text{LP}} := \max_{\boldsymbol{x}} \ & \boldsymbol{\mu}^\top \boldsymbol{x} \\ \text{s.t. } & \boldsymbol{C}\boldsymbol{x} \le \boldsymbol{B} \\ & \boldsymbol{x} \ge \boldsymbol{0} \end{aligned} \tag{1}$$

where the decision variables are $\boldsymbol{x} = (x_1, ..., x_m)^\top \in \mathbb{R}^m$. One can show that (see (Badanidiyuru et al., 2013))

$$\text{OPT}_{\text{LP}} \ge \text{OPT}$$

so that OPT_{LP} provides a deterministic upper bound for the expected reward under the optimal dynamic policy. Let $\boldsymbol{x}^* = (x_1^*, ..., x_m^*)^\top$ denote the optimal solution to (1).

3. Primal-dual Perspective for BwK

In this section, we present a generic regret upper bound and explore the properties of the underlying primal and dual LPs.

Let $\mathcal{I}^*$ and $\mathcal{I}'$ denote the set of optimal basic variables and the set of optimal non-basic variables of (1), respectively. Let $\mathcal{J}^*$ and $\mathcal{J}'$ denote the set of binding and non-binding constraints of (1), respectively. That is,

$$\begin{aligned} \mathcal{I}^* &:= \left\{x_i^* > 0, i \in [m]\right\}, \\ \mathcal{I}' &:= \left\{x_i^* = 0, i \in [m]\right\}, \\ \mathcal{J}^* &:= \left\{B - \sum_{i=1}^{m} c_{ji} x_i^* = 0, j \in [d]\right\}, \\ \mathcal{J}' &:= \left\{B - \sum_{i=1}^{m} c_{ji} x_i^* > 0, j \in [d]\right\}. \end{aligned}$$

So, we know $\mathcal{I}^* \cap \mathcal{I}' = [m]$ and $\mathcal{J}^* \cap \mathcal{J}' = [d]$. Accordingly, we call an arm $i \in \mathcal{I}^*$ as an optimal arm and $i \in \mathcal{I}'$ as a sub-optimal arm. Here and hereafter, we will refer to knapsack as constraint so that the terminology is more aligned with the LPs.

We make the following assumption on the LP's optimal solution.

Assumption 2. *The LP (1) has an unique optimal solution. Moreover, the optimal solution is non-degenerate, i.e.,*

$$|\mathcal{I}^*| = |\mathcal{J}^*|.$$

The assumption is a standard one in LP's literature, and any LP can satisfy the assumption with an arbitrarily small perturbation (Megiddo & Chandrasekaran, 1989). To interpret the non-degeneracy, consider if $|\mathcal{I}^*| = |\mathcal{J}^*| = l$, then the optimal solution to LP (1) is to play the only l arms in $\mathcal{I}^*$. When there is no linear dependency between the columns of $\boldsymbol{C}$, that will result in a depletion of l resource constraints.

The dual problem of (1) is

$$\begin{aligned} \min_{\boldsymbol{y}} \ & \boldsymbol{B}^\top \boldsymbol{y} \\ \text{s.t. } & \boldsymbol{C}^\top \boldsymbol{y} \ge \boldsymbol{\mu} \\ & \boldsymbol{y} \ge \boldsymbol{0}. \end{aligned} \tag{2}$$

Denote its optimal solution as $\boldsymbol{y}^* = (y_1^*, ..., y_d^*)$. From LP's complementarity condition, we know the following relation holds under Assumption 2,

$$j \in \mathcal{J}^* \Leftrightarrow y_j^* > 0, \ \ j \in \mathcal{J}' \Leftrightarrow y_j^* = 0.$$

The following lemma summarizes the LP's properties.

Lemma 1. *Under Assumption 2, we have the primal LP (1) and the dual LP (2) share the same optimal objective value. Also,*

$$\begin{aligned} |\mathcal{I}^*| + |\mathcal{J}'| &= d, \\ |\mathcal{I}'| + |\mathcal{J}^*| &= m. \end{aligned}$$

3.1. A Generic Regret Upper Bound

We begin our discussion with deriving a new upper bound for a generic BwK algorithm. First, we define the knapsack process as the remaining resource capacity at each time t. Specifically, we define $\boldsymbol{B}^{(0)} := \boldsymbol{B}$ and

$$\boldsymbol{B}^{(t+1)} := \boldsymbol{B}^{(t)} - \boldsymbol{C}_t$$

for $t \in [T]$. Recall that $\boldsymbol{C}_t$ is the (random) resource consumption at time t. The process $\boldsymbol{B}^{(t)}$ is pertaining to the BwK algorithm. In addition, we follow the convention of the bandits literature and define the count process $n_i(t)$ as the number of times the i-th arm is played up to the end of time period t.

Proposition 1. *The following inequality holds for any BwK algorithm,*

$$\textit{Regret}_T^{\pi}(\mathcal{P}, \boldsymbol{B}) \le \sum_{i \in \mathcal{I}'} n_i(t)\Delta_i + \mathbb{E}\left[\boldsymbol{B}^{(\tau)}\right]^\top \boldsymbol{y}^*. \tag{3}$$

where $\Delta_i = \boldsymbol{c}_i^\top \boldsymbol{y}^* - \mu_i$ *for* $i \in [m]$.

Here Δ_i is known as reduced cost/profit in LP literature and it quantifies the cost-efficiency of each arm (each basic variable in LP). The upper bound in Proposition 1 is new to the existing literature and it can be generally applicable to all BwK algorithms. It consists of two parts: (i) the number of times for which a sub-optimal arm is played multiplied by the corresponding reduced cost; (ii) the remaining resource at time τ, either when any of the resource is depleted or at the end of time horizon T. The first part is consistent with the classic bandits literature in that we always want to upper bound the number of sub-optimal arms being played throughout the horizon. At each time a sub-optimal arm $i \in \mathcal{I}'$ is played, a cost of Δ_i will be induced. Meanwhile, the second part is particular to the bandits with knapsacks setting and can easily be overlooked. Recall that the definition of τ refers to the first time that any resource $j \in [d]$ is exhausted (or the end of the horizon). It tells that the left-over of resources when the process terminates at time τ may also induce regret. For example, for two binding resources $j, j' \in \mathcal{J}^*$, it would be less desirable for one of them j to have a lot of remaining while the other one j' is exhausted. Since the binding resources are critical in determining optimal objective value for LP (1), intuitively, it is not profitable to waste any of them at the end of the procedure.

3.2. Symmetry between Arms and Bandits

From Proposition 1, we see the importance of dual problem (2) in bounding an algorithm's regret. Now, we pursue further along the path and propose a new notion of sub-optimality for the BwK problem. Our sub-optimality measure is built upon both the primal and dual LPs, and it reveals the combinatorial structure of the problem. In the following, we define two classes of LPs, one for the arm $i \in [m]$ and the other for the constraints $j \in [d]$.

First, for each arm $i \in [m]$, define

$$\begin{aligned} \mathrm{OPT}_i := \max_{\boldsymbol{x}} \ & \boldsymbol{\mu}^\top \boldsymbol{x}, \\ \text{s.t.} \ & \boldsymbol{C}\boldsymbol{x} \le \boldsymbol{B}, \\ & x_i = 0, \boldsymbol{x} \ge \boldsymbol{0}. \end{aligned} \tag{4}$$

By definition, OPT_i denotes the optimal objective value of an LP that takes the same form as the primal LP (1) except with an extra constraint $x_i = 0$. It represents the optimal objective value if the i-th arm is not allowed to use. For a sub-optimal arm $i \in \mathcal{I}'$, $\mathrm{OPT}_i = \mathrm{OPT}_{\mathrm{LP}}$, while for an optimal arm $i \in \mathcal{I}^*$, $\mathrm{OPT}_i < \mathrm{OPT}_{\mathrm{LP}}$. In this way, OPT_i characterizes the importance of arm i.

Next, for each constraint $j \in [d]$, define

$$\begin{aligned} \mathrm{OPT}_j := \min_{\boldsymbol{y}} \ & \boldsymbol{B}^\top \boldsymbol{y} - B, \\ \text{s.t.} \ & \boldsymbol{C}^\top \boldsymbol{y} \ge \boldsymbol{\mu} + \boldsymbol{C}_{j,\cdot}, \\ & \boldsymbol{y} \ge \boldsymbol{0}, \end{aligned} \tag{5}$$

where $\boldsymbol{C}_{j,\cdot}$ denotes the j-th row of the constraint matrix $\boldsymbol{C}$. Though it may not be as obvious as the previous case of OPT_i, the definition of OPT_j aims to characterize the bindingness/non-bindingness of a constraint j. The point can be illustrated by looking at the primal problem for (5). From LP's strong duality, we know

$$\begin{aligned} \mathrm{OPT}_j = \max_{\boldsymbol{x}} \ & \boldsymbol{\mu}^\top \boldsymbol{x} - (B - \boldsymbol{C}_{j,\cdot}^\top \boldsymbol{x}), \\ \text{s.t.} \ & \sum_{i=1}^{m} \boldsymbol{C}\boldsymbol{x} \le \boldsymbol{B}, \\ & \boldsymbol{x} \ge 0. \end{aligned} \tag{6}$$

Compared to the original primal LP (1), there is an extra term in the objective function in LP (6). The extra term is a penalization for the left-over of the j-th constraint, and thus it encourages the usage of the j-th constraint. For a binding constraint $j \in \mathcal{J}^*$, it will be exhausted under the optimal solution $\boldsymbol{x}^*$ to LP (1) so the penalization term does not have any effect, i.e., $\mathrm{OPT}_j = \mathrm{OPT}_{\mathrm{LP}}$. In contrast, for a non-binding constraint $j \in \mathcal{J}'$, the extra term will result in a reduction in the objective value, i.e., $\mathrm{OPT}_j < \mathrm{OPT}_{\mathrm{LP}}$. We note that one can introduce any positive weight to the penalization term so as to trade off between the reward and the left-over of the j-th constraint in (6), but its current version suffices our discussion.

The following proposition summarizes the properties of OPT_i and OPT_j.

Proposition 2. *Under Assumption 2, we have*

$$\begin{aligned} & OPT_i < OPT_{LP} \Leftrightarrow i \in \mathcal{I}^*, \\ & OPT_i = OPT_{LP} \Leftrightarrow i \in \mathcal{I}', \\ & OPT_j = OPT_{LP} \Leftrightarrow j \in \mathcal{J}^*, \\ & OPT_j < OPT_{LP} \Leftrightarrow j \in \mathcal{J}'. \end{aligned}$$

In this way, the definition of OPT_i distinguishes optimal arms $\mathcal{I}^*$ from sub-optimal arms $\mathcal{I}'$, while the definition of OPT_j distinguishes the binding constraints $\mathcal{J}^*$ from non-binding constraints $\mathcal{J}'$. The importance of such a distinguishment arises from the upper bound in Proposition 1: on one hand, we should avoid playing sub-optimal arms, and on the other hand, we should exhaust the binding resources. A second motivation for defining both OPT_i and OPT_j can be seen after we present our algorithm. Furthermore, we remark that a measurement of the sub-optimality of the arms has to be defined through the lens of LP due to the combinatorial nature of the problem. The effect of the i-th arm's removal on the objective value can only be gauged by solving an alternative LP of OPT_i. Similarly, a measurement of the bindingness of the constraints should also take into account the combinatorial relation between constraints.

Next, define

$$\delta := \frac{1}{T}\left(\text{OPT}_{\text{LP}} - \max\left\{\max_{i\in\mathcal{I}^*}\text{OPT}_i, \max_{j\in\mathcal{J}'}\text{OPT}_j\right\}\right)$$

where the factor $\frac{1}{T}$ is to normalize the optimality gap by the number of time periods. Under Assumption 1, all the objective values in above should scale linearly with T.

To summarize, δ characterizes the hardness of distinguishing optimal arms from non-optimal arms (and binding constraints from non-binding constraints). It can be viewed as a generalization of the sub-optimality measure $\delta_{\text{MAB}} = \min_{i\neq i^*}\mu_{i^*} - \mu_i$ for the MAB problem (Lai & Robbins, 1985). δ_{MAB} characterizes the hardness of an MAB problem instance, i.e., the hardness of distinguishing the optimal arm from sub-optimal arms. In the context of BwK, δ is a more comprehensive characterization in that it takes into account both the arms and the constraints. Imaginably, it will be critical in both algorithm design and analysis for the BwK problem.

3.3. Key Parameters of the LPs

Now, we define two LP-related quantities that will appear in our regret bound:

- Linear Dependency between Arms: Define σ be the minimum singular value of the matrix $\boldsymbol{C}_{\mathcal{I}^*,\mathcal{J}^*}$. Specifically,

  $$\sigma := \sigma_{\min}\left(\boldsymbol{C}_{\mathcal{J}^*,\mathcal{I}^*}\right) = \sigma_{\min}\left((c_{ji})_{j\in\mathcal{J}^*, i\in\mathcal{I}^*}\right).$$

  In this light, σ represents the linear dependency between optimal arms across the binding constraints. For a smaller value of σ, the optimal arms are more linearly dependent, and then it will be harder to identify the optimal numbers of plays. Under Assumption 2, the uniqueness of optimal solution implies $\sigma > 0$.

- Threshold on the optimal solution:

  $$\chi := \frac{1}{T}\cdot\min\{x_i^* \neq 0, i \in [m]\}$$

  If the total resource $\boldsymbol{B} = T\cdot\boldsymbol{b}$, both the optimal solution $(x_1^*, ..., x_m^*)$ should scale linearly with T. The factor $\frac{1}{T}$ normalizes the optimal solution into a probability vector. χ denotes the smallest non-zero entry for the optimal solution. Intuitively, a small value of χ implies that the optimal proportion of playing an arm $i \in \mathcal{I}^*$ is small and thus it is more prone to "overplay" the arm.

Remarks. By the definition, it seems that the above parameters δ and χ both involve a factor of T. But if we replace $\boldsymbol{B}$ (the right-hand-side of LP) with $\boldsymbol{b}$ from Assumption 1, then the factor T disappears, and the parameters χ and δ are essentially dependent on $\boldsymbol{\mu}$, $\boldsymbol{C}$, and $\boldsymbol{b}$ which are inherent to the problem instance but bear no dependency on the horizon T. In other words, Assumption 1 frees the dependency on T by introducing the quantity b. Practically, the assumption states the resource capacity should be sufficiently large and it is natural in many application contexts (for example, the small bids assumption in AdWords problem (Mehta et al., 2005)). Theoretically, in two previous works (Flajolet & Jaillet, 2015; Sankararaman & Slivkins, 2020), either a factor of $1/T$ appears in the parameter definition (Sankararaman & Slivkins, 2020) or the assumption is explicitly imposed (Flajolet & Jaillet, 2015). Such an assumption might be inevitable for a logarithmic regret to be derived.

3.4. LCB and UCB

Throughout this paper, we denote the reward and resource consumption of the s-th play of the i-th arm as $r_{i,s}$ and $\boldsymbol{C}_{i,s} = (C_{1i,s}, ..., C_{di,s})^\top$ respectively, for $i \in [m]$ and $s \in [T]$. Let $n_i(t)$ be the number of times the i-th arm is played in the first t time periods. Accordingly, we denote the estimators at time t for the i-th arm as

$$\hat{\mu}_i(t) := \frac{1}{n_i(t)}\sum_{s=1}^{n_i(t)} r_{i,s},$$

$$\hat{C}_{ji}(t) := \frac{1}{n_i(t)}\sum_{s=1}^{n_i(t)} C_{ji,s}$$

for $i \in [m]$ and $j \in [d]$. In a similar manner, we define the estimator for the i-th arm's resource consumption vector as $\hat{\boldsymbol{C}}_i(t) := (\hat{C}_{1i}(t), ..., \hat{C}_{di}(t))^\top$, and the resource consumption matrix as $\hat{\boldsymbol{C}}(t) := (\hat{\boldsymbol{C}}_1(t), ..., \hat{\boldsymbol{C}}_m(t))$. Specifically, without changing the nature of the analysis, we ignore the case that when $n_i(t) = 0$. Then, we define the lower confidence bound (LCB) and upper confidence bound (UCB) for the parameters as

$$\mu_i^L(t) := \mathit{proj}_{[0,1]}\left(\hat{\mu}_i(t) - \sqrt{\frac{2\log T}{n_i(t)}}\right)$$

$$\mu_i^U(t) := \mathit{proj}_{[0,1]}\left(\hat{\mu}_i(t) + \sqrt{\frac{2\log T}{n_i(t)}}\right)$$

$$C_{ji}^L(t) := \mathit{proj}_{[0,1]}\left(\hat{C}_{ji}(t) - \sqrt{\frac{2\log T}{n_i(t)}}\right)$$

$$C_{ji}^U(t) := \mathit{proj}_{[0,1]}\left(\hat{C}_{ji}(t) + \sqrt{\frac{2\log T}{n_i(t)}}\right)$$

where $proj_{[0,1]}(\cdot)$ projects a real number to interval $[0,1]$. The following lemma is standard in bandits literature and it characterizes the relation between the true values and the LCB/UCB estimators. It states that all the true values will fall into the intervals defined by the corresponding estimators with high probability.

Lemma 2 (Concentration)**.** *The following event holds with probability no less than* $1-\frac{4md}{T^2}$*,*

$$\begin{aligned} \mu_i &\in \left(\mu_i^L(t), \mu_i^U(t)\right), \\ c_{ji} &\in \left(C_{ji}^L(t), C_{ji}^U(t)\right) \end{aligned}$$

for all $i \in [m], j \in [d], t \in T$.

With the UCB/LCB estimators for the parameters, we can construct UCB/LCB estimators for the objective of the primal LP. Specifically,

$$\begin{aligned} \text{OPT}_{\text{LP}}^{U} := \max_{\boldsymbol{x}} \ & \left(\boldsymbol{\mu}^U\right)^\top \boldsymbol{x}, \\ \text{s.t. } & \boldsymbol{C}^L \boldsymbol{x} \le \boldsymbol{B}, \\ & \boldsymbol{x} \ge \boldsymbol{0}. \end{aligned} \tag{7}$$

$$\begin{aligned} \text{OPT}_{\text{LP}}^{L} := \max_{\boldsymbol{x}} \ & \left(\boldsymbol{\mu}^L\right)^\top \boldsymbol{x}, \\ \text{s.t. } & \boldsymbol{C}^U \boldsymbol{x} \le \boldsymbol{B}, \\ & \boldsymbol{x} \ge \boldsymbol{0}. \end{aligned} \tag{8}$$

The following lemma states the relation between $\text{OPT}_{\text{LP}}^{U}$, $\text{OPT}_{\text{LP}}^{L}$, and OPT_{LP}. Intuitively, if we substitute the original constraint matrix $\boldsymbol{C}$ with its LCB (or UCB) and the objective coefficient $\boldsymbol{\mu}$ with its UCB (or LCB), the resultant optimal objective value will be a UCB (or LCB) for OPT_{LP}.

Lemma 3. *The following inequality holds with probability no less* $1-\frac{4md}{T^2}$,

$$\mathit{OPT}_{\mathit{LP}}^{L} \le \mathit{OPT}_{\mathit{LP}} \le \mathit{OPT}_{\mathit{LP}}^{U}.$$

A similar approach is used in (Agrawal & Devanur, 2014) to develop an UCB-based algorithm for the BwK problem. For our algorithm presented in the following section, we will construct estimates not only for the primal LP (1), but also for the LPs (4) and (5). By comparing the estimates of OPT_{LP}, OPT_i, and OPT_j, we will be able to identify the optimal arms $\mathcal{I}^*$ and the non-binding constraints $\mathcal{J}'$.

4. Two-Phase Algorithm

In this section, we describe our two-phase algorithm for the BwK problem. The main theme is to use the underlying LP's solution to guide the plays of the arms. The two phases in the algorithm correspond to the two parts of the regret upper bound in Proposition 1. In the following, we describe the two phases of the algorithm and their intuitions respectively.

Phase I of Algorithm 1 is an elimination algorithm and it aims to identify the optimal arms $\mathcal{I}^*$ and the non-binding constraints $\mathcal{J}'$. In each round of the while loop in Phase I, all the arms are played once to improve the estimators for μ and $\boldsymbol{C}$. After each round of plays, an identification procedure is conducted by comparing the LCB of the original optimal value ($\text{OPT}_{\text{LP}}^{L}$) against the UCBs ($\text{OPT}_i^U$ and OPT_j^U). Recall that Proposition 2, there will be a non-zero sub-optimality gap if $i \in \mathcal{I}^*$ or $j \in \mathcal{J}'$. So, if the algorithm observes a gap between the corresponding LCBs/UCBs, it will assert $i \in \mathcal{I}^*$ or $j \in \mathcal{J}'$, and the assertion will be true with high probability.

The stopping rule in Phase I originates from the complementary condition in Lemma 1, i.e., $|\hat{\mathcal{I}}^*| + |\hat{\mathcal{J}}'| < d$. This is a key in the primal-dual design of the algorithm and it further justifies the consideration of the dual problem. Without maintaining the set $\hat{\mathcal{J}}'$, we cannot decide whether we have obtain the true primal set, i.e., $\hat{\mathcal{I}}^* = \mathcal{I}^*$. Specifically, since there is no precise knowledge of the number of optimal arms $|\mathcal{I}^*|$, while we keep adding arms into $\hat{\mathcal{I}}^*$, we do not know when to stop. The complementarity in Lemma 1 provides a condition on the number of arms in $\mathcal{I}^*$ and the number of constraints in $\mathcal{J}'$. Accordingly, Phase I is terminated when this condition is met. Moreover, we emphasize that a by-product of the Phase I is a *best arm identification* procedure. To the best of our knowledge, this is the first result on identifying optimal arms for the BwK problem.

Phase II of Algorithm 1 is built upon the output of Phase I. At each time t, the algorithm solves an adaptive version LP (9) and normalizes its optimal solution into a sampling scheme on arms. Also, in Phase II, the algorithm will only play arms $i \in \hat{\mathcal{I}}^*$; this is achieved by enforcing $x_i = 0$ for $i \in \hat{\mathcal{I}}^*$ in (9). The adaptive design is exemplified on the right-hand-side of the LP (9), where instead of the static resource capacity $\boldsymbol{B}$, it uses the remaining resource at the end of last time period $\boldsymbol{B}^{(t-1)}$. To see its intuition, consider if a binding resource $j \in \mathcal{J}^*$ is over-used in the first t time periods, then it will result in a smaller value of $B_j^{(t-1)}$, and then the adaptive mechanism will tend to be more reluctant to consume the j-th resource in the future, and vice versa for the case of under-use.

We emphasize that the adaptive design is not only intuitive but also necessary to achieve a regret that is logarithmic in T. If we adopt a static right-hand-side $\boldsymbol{B}$ in (9) (as in (Badanidiyuru et al., 2013; Agrawal & Devanur, 2014)), then the fluctuation of the process $\boldsymbol{B}_t$ will be on the order of $\Omega(\sqrt{t})$. Consequently, when it approaches to the end of the horizon, certain type of binding resource may be exhausted while other binding resources still have $\Omega(\sqrt{T})$ left-over, and this may result in an $\Omega(\sqrt{T})$ upper bound in Proposition 1. The intuition is made rigorous by (Arlotto & Gurvich, 2019); the paper establishes that without an adaptive design,

Algorithm 1 Primal-dual Adaptive Algorithm for BwK

```
Input: Resource capacity $\boldsymbol{B}, T$
%% Phase I: Identification of $\mathcal{I}^*$ and $\mathcal{J}'$
Initialize $\hat{\mathcal{I}}^* = \hat{\mathcal{J}}' = \emptyset, t = 0$
Initialize the knapsack process $\boldsymbol{B}^{(0)} = \boldsymbol{B}$
while $|\hat{\mathcal{I}}^*| + |\hat{\mathcal{J}}'| < d$ do
    Play each arm $i \in [m]$ once
    Update $t = t + m$ and the knapsack process $\boldsymbol{B}^{(t)}$
    Update the estimates $\hat{\boldsymbol{\mu}}(t)$ and $\hat{\boldsymbol{C}}(t)$
    Solve the LCB problem (8) and obtain $\mathrm{OPT}_{\mathrm{LP}}^{L}(t)$
    for $i \notin \hat{\mathcal{I}}^*$ do
        Solve the following UCB problem for $\mathrm{OPT}_i$
            $\mathrm{OPT}_i^U(t) := \max_{\boldsymbol{x}} \ \left(\boldsymbol{\mu}^U(t)\right)^\top \boldsymbol{x},$
                $\text{s.t.} \ \boldsymbol{C}^L(t)\boldsymbol{x} \le \boldsymbol{B},$
                $x_i = 0, \boldsymbol{x} \ge \boldsymbol{0}.$
        if $\mathrm{OPT}_{\mathrm{LP}}^{L}(t) > \mathrm{OPT}_i^{\mathrm{U}}(t)$ then
            Update $\hat{\mathcal{I}}^* = \hat{\mathcal{I}}^* \cup \{i\}$
        end if
    end for
    for $j \notin \hat{\mathcal{J}}'$ do
        Solve the following UCB problem for $\mathrm{OPT}_j$
            $\mathrm{OPT}_j^U(t) := \min_{\boldsymbol{y}} \ \boldsymbol{B}^\top \boldsymbol{y} - B,$
                $\text{s.t.} \ (\boldsymbol{C}^L(t))^\top \boldsymbol{y} \ge \boldsymbol{\mu}^U(t) + \boldsymbol{C}_{j,\cdot}^U(t),$
                $\boldsymbol{y} \ge 0.$
        if $\mathrm{OPT}_{\mathrm{LP}}^{L}(t) > \mathrm{OPT}_j^{\mathrm{U}}(t)$ then
            Update $\hat{\mathcal{J}}' = \hat{\mathcal{J}}' \cup \{j\}$
        end if
    end for
end while
Update $t = t + 1$
%% Phase II: Exhausting the binding resources
while $t \le \tau$ do
    Solve the following LP
        $\max_{\boldsymbol{x}} \ \left(\boldsymbol{\mu}^U(t-1)\right)^\top \boldsymbol{x},$    (9)
            $\text{s.t.} \ \boldsymbol{C}^L(t-1)\boldsymbol{x} \le \boldsymbol{B}^{(t-1)},$
            $x_i = 0, \ i \notin \mathcal{I}^*,$
            $\boldsymbol{x} \ge \boldsymbol{0}.$
    Denote its optimal solution as $\tilde{\boldsymbol{x}}$
    Normalize $\tilde{\boldsymbol{x}}$ into a probability and randomly play an
      arm according to the probability
    Update estimates $\hat{\boldsymbol{\mu}}(t)$, $\hat{\boldsymbol{C}}(t)$, and $\boldsymbol{B}^{(t)}$
    Update $t = t + 1$
end while
```

the regret is at least $\Omega(\sqrt{T})$ for the multi-secretary problem (can be viewed as a one-constraint BwK problem) even if the underlying distribution is known.

5. Regret Analysis

In this section, we derive an regret upper bound for Algorithm 1 by analyzing the two phases separately.

5.1. Analysis of Phase I

Proposition 3 provides an upper bound on the number of time periods within which Phase I will terminate. It also states that the identification of $\mathcal{I}^*$ and $\mathcal{J}'$ will be precise conditional on the high probability event in Lemma 2.

Proposition 3. *In Phase I of Algorithm 1, each arm $i \in [m]$ will be played for no more than $\left(2 + \frac{1}{b}\right)^2 \cdot \frac{72 \log T}{\delta^2}$ times where b is defined in Assumption 1. If the resources are not exhausted in Phase I, then its output satisfies*

$$\mathbb{P}\left(\hat{\mathcal{I}}^* = \mathcal{I}^*, \hat{\mathcal{J}}' = \mathcal{J}'\right) \Bigg(\ge 1 - \frac{4md}{T^2}.$$

The surprising point of Proposition 3 lies in that there are $O(2^{m+d})$ possible configurations of $(\mathcal{I}^*, \mathcal{J}')$ and, without any prior knowledge, the true configuration can be identified within $O(\log T)$ number of plays for each arm. In contrast, (Flajolet & Jaillet, 2015) does not utilize the primal-dual structure of the problem and conducts a brute-force search in all possible configurations which results in an $O(2^{m+d} \log T)$ regret. In addition, the search therein requires the knowledge of a non-degeneracy parameter a priori. The result also explains why (Sankararaman & Slivkins, 2020) imposes a single-best-arm condition for the BwK problem, which assumes $\mathcal{I}^* = 1$. This additional greatly simplifies the combinatorial structure and reduces the BwK problem more closely to the standard MAB problem. In this light, Proposition 3 can be viewed as a best-arm-identification result (Audibert & Bubeck, 2010) for the BwK problem in full generality and without any prior knowledge.

5.2. Analysis of Phase II

Proposition 4 provides an upper bound on the remaining resource for binding constraints when the procedure terminate, which corresponds to the second part of Proposition 1. Notably, the upper bound has no dependency on T. In a comparison with the $\Omega(\sqrt{T})$ fluctuation under the static design, it demonstrates the effectiveness of our adaptive design.

Proposition 4. *For each binding constraint $j \in \mathcal{J}^*$, we have*

$$\mathbb{E}\left[B_j^{(\tau)}\right] \Bigg(= O\left(\frac{d^3}{b \min\{\chi^2, \delta^2\} \min\{1, \sigma^2\}}\right)$$

where χ and σ are defined in Section 3.3, and b is defined in Assumption 1.

The idea of proof is to introduce an auxiliary process $b_j^{(t)} = \frac{B_j^{(t)}}{T-t}$ for $t \in [T-1]$ and $j \in \mathcal{J}^*$. Recall that $B_j^{(t)}$ is the j-th component of the knapsack process $\boldsymbol{B}^{(t)}$, we know its initial value $b_j^{(0)} = b$. Then define

$$\tau_j = \min\{t : b_j^{(t)} \notin [b-\epsilon, b+\epsilon]\} \cup \{T\}$$

for a fixed $\epsilon > 0$. With the definition, $b_j^{(t)}$ can be interpreted as average remaining resource (per time period) and τ_j can be interpreted as the first time that $b_j^{(t)}$ deviates from its initial value b by a small amount. It is easy to see that $\tau_j \le \tau$ with a proper choice of ϵ. Next, we aim to upper bound $\mathbb{E}[T-\tau_j]$ by analyzing the process $\{b_j^{(t)}\}_{t=0}^T$. From the dynamic of the knapsack process $\boldsymbol{B}_t$, we know that

$$\begin{aligned} b_j^{(t)} &= \frac{B_j^{(t)}}{T-t} = \frac{B_j^{(t-1)} - C_{j,t}}{T-t} \\ &= b_j^{(t-1)} - \frac{1}{T-t}\left(C_{j,t} - b_j^{(t-1)}\right) \qquad (10) \end{aligned}$$

where $C_{j,t}$ as defined earlier is the resource consumption of j-th constraint at time t. The above formula (10) provides a technical explanation for the motivation of the adaptive design in (9). Intuitively, when the right-hand-side of (9) is $\boldsymbol{B}^{(t-1)}$, it will lead to a solution that (approximately and on expectation) consumes $b_j^{(t-1)}$ of the j-th resource for each of the following time periods. Ideally, this will make the second term in (10) have a zero expectation. However, due to estimation error for the LP's parameters, this may not be the case. The idea is to first provide an upper bound for the "bias" term

$$\mathbb{E}\left[C_{j,t} - b_j^{(t-1)} | \mathcal{H}_{t-1}\right]$$

where $\mathcal{H}_{t-1} = \{(r_s, \boldsymbol{C}_s, i_s)\}_{s=1}^{t-1}$ encapsulates all the information up to time $t-1$. Unsurprisingly, the upper bound is on the order of $O\left(\frac{1}{\sqrt{t}}\right)$. Next, with this bias upper bound, we can construct a super-martingale (sub-martingale) based on the dynamic (10) and employ Azuma–Hoeffding inequality to provide a concentration result for the value of the martingale. Through the analysis of the process $b_j^{(t)}$, we can derive an upper bound for $\mathbb{E}[T-\tau_j]$, and consequently, it leads to an upper bound on $\mathbb{E}[B_j^{(\tau)}]$.

The importance of the adaptive design has been widely recognized in other constrained online learning problems, such as online matching problem (Manshadi et al., 2012), online assortment problem (Golrezaei et al., 2014), online linear programming problem (Li & Ye, 2019), network revenue management problem (Jasin & Kumar, 2012), etc. The common pattern of these problem is to allocate limited resources in a sequential manner, and the idea of adaptive design is to adjust the allocation rule dynamically according to the remaining resource/inventory. This is in parallel with LP (9) where the solution at each time t is contingent on the remaining resources $\boldsymbol{B}^{(t)}$. The significance of our algorithm design and analysis lies in that (i) to the literature of BwK, our paper is the first application of the idea of adaptive design; (ii) to the existing literature of adaptive design in constrained online learning problems, our work provides its first application and analysis in a partial-information environment. For the second aspect, all the existing analysis on the adaptive design fall in the paradigm of "first-observe-then-decide" while the BwK problem is "first-decide-then-observe". Specifically, in matching/resource allocation/revenue management problems, at each time period, a new agent arrives, and upon the observation, we decide the matching for the agent; or a new customer arrives, and upon the observation of her preference, we decide the assortment decision for the customer. So, the existing analyses are analogous to a BwK "setting" where the reward and resource consumption of playing an arm are first observed (magically), and then we decide whether we want to play the arm or not.

5.3. Regret Upper Bound for Algorithm 1

Combining the two parts, we have the following result on the regret of Algorithm 1.

Proposition 5. *The regret of Algorithm 1 has the following upper bound,*

$$O\left(\left(2+\frac{1}{b}\right)^2 \frac{md\log T}{b\delta^2} + \frac{d^4}{b^2\min\{\chi^2,\delta^2\}\min\{1,\sigma^2\}}\right)$$

where b is defined in Assumption 1, δ is defined in Section 3.2, and σ and χ are defined in Section 3.3.

The result reduces the exponential dependence on m and d in (Flajolet & Jaillet, 2015) to polynomial, and also it does not rely on any prior knowledge. Specifically, the authors consider several settings for BwK problem, many of which assume special structures such as one or two resource constraints. The most general settings therein, which is comparable to ours, allows arbitrary number of constraints and number of optimal arms. In terms of the key parameters, the way we define δ is the same as their definition of Δ_x. However, the regret bound (Theorem 8 therein) involves a summation of exponentially many $\frac{1}{\Delta_x}$'s (the same as the total number of the bases of the LP). Our parameter σ is related to their ϵ (Assumption 8 therein) while the latter is more restrictive. Because σ in our paper represents the minimal singular value of the matrix corresponding to only the optimal basis of the primal LP, whereas the parameter ϵ therein represents a lower bound of the determinant of the matrices corresponding to all the possible (exponentially many) bases of the primal LP. In this light, if they adopt our parameter σ, their bound would be improved by a factor of $d!$ ($C!$ therein). Moreover, ϵ is a lower bound for our parameter χ and (Flajolet & Jaillet, 2015) explicitly requires the knowledge of ϵ a priori.

The proposition also generalizes the one-constraint bound in (Sankararaman & Slivkins, 2020) and relaxes the deterministic resource consumption assumption therein. Specifically, the authors assume there is one single optimal arm and one single resource (other than time), i.e., the optimal solution to the primal LP has only one non-zero entry ($|\mathcal{I}^*| = |\mathcal{J}^*| = 1$ and $d = 2$). They also assume the underlying LP is non-degenerate. Our results generalize their work in allowing arbitrary d, $|\mathcal{I}^*|$ and $|\mathcal{J}^*|$. In terms of the key parameters, under their assumption, our parameter $\sigma = 1$ because it is defined by a 1-by-1 matrix. Our parameter χ is deemed as a constant in their paper. For δ, under their assumptions, our definition of OPT_i can be adjusted accordingly. Specifically, we can replace the constraint $x_i = 0$ in defining OPT_i with $x_{i'} = 0, i' \neq i$. Then our definition of δ would reduce to their definition of $G_{LAG}(a)$, both of which characterize the sub-optimality gap of an arm.

The above comparison against the existing literature highlights that the parameters σ, δ, and χ or other LP-based parameters might be inevitable in characterizing logarithmic regret bound. Our paper makes some preliminary efforts along this line, but we do not believe our bound is tight: parameters such as χ and σ may be replaced with some tighter parameter through a better algorithm and sharper analysis. As remarked in Section 3.3, the parameters are not dependent on T under Assumption 1. But to characterize the dependency of these parameters (such as δ and χ) on the problem size m and d remains an open question. Moreover, our algorithm and analysis highly rely on the structural properties of the underlying LP, which might not be the unique method to handle the BwK problem.

6. Conclusion

In this paper, we introduce a new BwK algorithm and derive problem-dependent bound for the algorithm. In the Phase I of the algorithm, it involves a round-robin design and may result in playing sub-optimal arms for inefficiently many times. Our regret bound can be large when the parameters σ, δ, and χ are small and the inefficient plays of the sub-optimal arms may prevent the algorithm from achieving an $O(\sqrt{T})$ worst-case regret. In the extreme case, these parameters may scale with $O(\frac{1}{T})$, though Assumption 1 prevents such a possibility. So the question is whether Assumption 1 is necessary in admitting a logarithmic problem-dependent bound.

We conclude our discussion with a new one-phase algorithm – Algorithm 2. The algorithm is also LP-based, and at each time t, it solves a UCB version of the primal LP to sample the arm. The algorithm has an adaptive design to exhaust the resources. On one hand, the algorithm naturally incorporates the Phase I of Algorithm 1 as a part of its Phase II. It directly enters the Phase II of Algorithm 1 and lets the adaptive LP to fully determine the arm(s) to play (without the extra constraint in (9)). On the other hand, the algorithm can be viewed as an adaptive version of the algorithm in (Agrawal & Devanur, 2014). Our conjecture is that Algorithm 2 is the optimal algorithm for BwK: it is optimal in the sense that it achieves optimal problem-dependent bound, but also admits $O(\sqrt{T})$ problem-independent bound. Unfortunately, its analysis is more challenging than Algorithm 1, which we leave as an open question.

Algorithm 2 One-Phase Adaptive Algorithm for BwK

```
Input: Resource capacity $\boldsymbol{B}, T$
Initialize the knapsack process $\boldsymbol{B}^{(0)} = \boldsymbol{B}$
Initialize the estimates $\hat{\boldsymbol{\mu}}(0)$ and $\hat{\boldsymbol{C}}(0)$
Set $t = 1$
while $t \leq \tau$ do
    Solve the following LP
```

$$\begin{aligned} \max_{\boldsymbol{x}} \ & \left(\boldsymbol{\mu}^U(t-1)\right)^\top \boldsymbol{x}, \\ \text{s.t. } & \boldsymbol{C}^L(t-1)\boldsymbol{x} \leq \boldsymbol{B}^{(t-1)}, \\ & \boldsymbol{x} \geq \boldsymbol{0}. \end{aligned} \tag{11}$$

```
    Denote its optimal solution as $\tilde{\boldsymbol{x}}$
    Normalize $\tilde{\boldsymbol{x}}$ into a probability and randomly play an
    arm according to the probability
    Update estimates $\hat{\boldsymbol{\mu}}(t)$, $\hat{\boldsymbol{C}}(t)$, and $\boldsymbol{B}^{(t)}$
    Update $t = t + 1$
end while
```

Acknowledgements

We would like to thank anonymous reviewers for their helpful suggestions and insightful comments.

References

Agrawal, S. and Devanur, N. R. Bandits with concave rewards and convex knapsacks. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pp. 989–1006, 2014.

Agrawal, S. and Devanur, N. R. Linear contextual bandits with knapsacks. *arXiv preprint arXiv:1507.06738*, 2015.

Agrawal, S. and Goyal, N. Analysis of thompson sampling for the multi-armed bandit problem. In *Conference on learning theory*, pp. 39–1. JMLR Workshop and Conference Proceedings, 2012.

Agrawal, S., Wang, Z., and Ye, Y. A dynamic near-optimal algorithm for online linear programming. *Operations Research*, 62(4):876–890, 2014.

Agrawal, S., Devanur, N. R., and Li, L. An efficient algorithm for contextual bandits with knapsacks, and an extension to concave objectives. In *Conference on Learning Theory*, pp. 4–18. PMLR, 2016.

Arlotto, A. and Gurvich, I. Uniformly bounded regret in the multisecretary problem. *Stochastic Systems*, 2019.

Audibert, J.-Y. and Bubeck, S. Best arm identification in multi-armed bandits. In *COLT-23th Conference on Learning Theory-2010*, pp. 13–p, 2010.

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

Badanidiyuru, A., Kleinberg, R., and Slivkins, A. Bandits with knapsacks. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pp. 207–216. IEEE, 2013.

Besbes, O. and Zeevi, A. Blind network revenue management. *Operations research*, 60(6):1537–1550, 2012.

Ferreira, K. J., Simchi-Levi, D., and Wang, H. Online network revenue management using thompson sampling. *Operations research*, 66(6):1586–1602, 2018.

Flajolet, A. and Jaillet, P. Logarithmic regret bounds for bandits with knapsacks. *arXiv preprint arXiv:1510.01800*, 2015.

Gittins, J., Glazebrook, K., and Weber, R. *Multi-armed bandit allocation indices*. John Wiley & Sons, 2011.

Golrezaei, N., Nazerzadeh, H., and Rusmevichientong, P. Real-time optimization of personalized assortments. *Management Science*, 60(6):1532–1551, 2014.

Immorlica, N., Sankararaman, K. A., Schapire, R., and Slivkins, A. Adversarial bandits with knapsacks. In *2019 IEEE 60th Annual Symposium on Foundations of Computer Science (FOCS)*, pp. 202–219. IEEE, 2019.

Jasin, S. and Kumar, S. A re-solving heuristic with bounded revenue loss for network revenue management with customer choice. *Mathematics of Operations Research*, 37 (2):313–345, 2012.

Kesselheim, T. and Singla, S. Online learning with vector costs and bandits with knapsacks. In *Conference on Learning Theory*, pp. 2286–2305. PMLR, 2020.

Lai, T. L. and Robbins, H. Asymptotically efficient adaptive allocation rules. *Advances in applied mathematics*, 6(1): 4–22, 1985.

Li, X. and Ye, Y. Online linear programming: Dual convergence, new algorithms, and regret bounds. *arXiv preprint arXiv:1909.05499*, 2019.

Manshadi, V. H., Gharan, S. O., and Saberi, A. Online stochastic matching: Online actions based on offline statistics. *Mathematics of Operations Research*, 37(4): 559–573, 2012.

Megiddo, N. and Chandrasekaran, R. On the ε-perturbation method for avoiding degeneracy. *Operations Research Letters*, 8(6):305–308, 1989.

Mehta, A., Saberi, A., Vazirani, U., and Vazirani, V. Adwords and generalized on-line matching. In *46th Annual IEEE Symposium on Foundations of Computer Science (FOCS'05)*, pp. 264–273. IEEE, 2005.

Robbins, H. Some aspects of the sequential design of experiments. *Bulletin of the American Mathematical Society*, 58(5):527–535, 1952.

Sankararaman, K. A. and Slivkins, A. Advances in bandits with knapsacks. *arXiv preprint arXiv:2002.00253*, 2020.

Weber, R. et al. On the gittins index for multiarmed bandits. *The Annals of Applied Probability*, 2(4):1024–1033, 1992.