The Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI-21)

Online DR-Submodular Maximization: Minimizing Regret and Constraint Violation

Prasanna Raut*[1], Omid Sadeghi*[2], Maryam Fazel[2]

[1]Department of Mechanical Engineering, University of Washington

[2] Department of Electrical and Computer Engineering, University of Washington

{raut, omids, mfazel}@uw.edu

Abstract

In this paper, we consider online continuous DR-submodular maximization with linear stochastic long-term constraints. Compared to the prior work on online submodular maximization, our setting introduces the extra complication of stochastic linear constraint functions that are i.i.d. generated at each round. In particular, at each time step a DR-submodular utility function and a constraint vector, i.i.d. generated from an unknown distribution, are revealed after committing to an action and we aim to maximize the overall utility while the expected cumulative resource consumption is below a fixed budget. Stochastic long-term constraints arise naturally in applications where there is a limited budget or resource available and resource consumption at each step is governed by stochastically time-varying environments. We propose the Online Lagrangian Frank-Wolfe (OLFW) algorithm to solve this class of online problems. We analyze the performance of the OLFW algorithm and we obtain sub-linear regret bounds as well as sub-linear cumulative constraint violation bounds, both in expectation and with high probability.

1 Introduction

The Online Convex Optimization (OCO) problem has been extensively studied in the literature (Hazan 2016; Shalev-Shwartz 2012; Zinkevich 2003; Orabona 2019). In this problem, a sequence of arbitrary convex cost functions $\{f_t(\cdot)\}_{t=1}^T$ are revealed one by one by "nature" and at each round $t \in [T]$, the decision maker chooses an action $x_t \in \mathcal{X}$, where $\mathcal{X}$ is the fixed domain set, before the corresponding function $f_t(\cdot)$ is revealed. The goal is to minimize the regret defined as (Zinkevich 2003)

$$\sum_{t=1}^{T} f_t(x_t) - \min_{x\in\mathcal{X}} \sum_{t=1}^{T} f_t(x).$$

In other words, regret characterizes the difference between the overall cost incurred by the decision maker and that of a fixed benchmark action which has access to all the cost functions $\{f_t\}_{t=1}^T$.

In many applications, however, in addition to maximizing the total reward (minimizing the overall cost), there are restrictions on the sequence of decisions made by the learner that need to be satisfied on average (Agrawal and Devanur 2015; Badanidiyuru, Kleinberg, and Slivkins 2018; Agrawal and Devanur 2014; Rivera, Wang, and Xu 2018). Therefore, it may be beneficial to sacrifice some of the reward to meet other desired goals or restrictions over the time horizon. Such long-term constraints arise naturally in applications with limited budget (resource) availability (Balseiro and Gur 2019; Besbes and Zeevi 2012; Ferreira, Simchi-Levi, and Wang 2018).

As an illustrative example, consider the online ad allocation problem for an advertiser. At each round $t \in [T]$, the advertiser should choose her investment on ads to be placed on n different websites. Beyond the immediate goal of maximizing the overall impressions of the ads, the advertiser needs to balance her total investment against an allotted budget on a daily, monthly or yearly basis (Balseiro and Gur 2019). However, the cost of ad placement in each round depends on the number of clicks the ads receive, so they are not known ahead of time. Therefore, the advertiser needs to strike the right balance between the total reward and budget used. See Appendix B for a number of other motivating applications that can be naturally cast in our framework (see Raut, Sadeghi, and Fazel (2021) for the full version of the paper including all the appendices).

In this paper, we study a new class of online allocation problems with long-term resource constraints where the utility functions are DR-submodular (and not necessarily concave) and the constraint functions are linear with coefficient vectors drawn i.i.d. from some unknown underlying distribution. The problem has been extensively studied in the convex setting (Yuan and Lamperski 2018; Wei, Yu, and Neely 2020; Neely and Yu 2017; Liakopoulos et al. 2019); furthermore, Sadeghi and Fazel (2020) considered a similar framework under the assumption that the linear constraint functions are chosen adversarially. However, Sadeghi and Fazel (2020) do not provide any high probability bounds for the regret and constraint violation with random i.i.d. linear constraints, and their expected constraint violation bound is worse than ours as well (see Section 3.2 and the Appendix for an overview of related work and comparison of our results with the existing bounds respectively). In this paper, we provide the *first* sub-linear bounds for the regret and total budget violation that hold in expectation as well as with high probability. Specifically, our contributions are as follows:

*Equal contribution

- In Section 4.1, We propose the Online Lagrangian Frank-Wolfe (OLFW) algorithm for this class of online continuous DR-submodular maximization problems with stochastic cumulative constraints. The OLFW algorithm is inspired by the quadratic penalty method in constrained optimization literature (Nocedal and Wright 1999) and it generalizes a Frank-Wolfe variant proposed by Chen, Hassani, and Karbasi (2018) for solving online continuous DR-submodular maximization problems to take into account the additional stochastically time-varying linear constraints. Note that this extension is not straightforward and the choice of the penalty function and the update rule for the dual variable are crucial in obtaining bounds for the total budget violation as well as the regret (see Section 4.1 for more details).
- We analyze the performance of the OLFW algorithm with high probability and in expectation in Section 4.2 and Section 4.3 respectively and we establish the first sub-linear *expected* and *high probability* bounds on both the *regret* and *total budget violation* of the algorithm.

Finally, in Section 5, we demonstrate the effectiveness of our proposed algorithm on simulated and real-world problem instances, and compare the performance of the OLFW algorithm with several baseline algorithms. All the missing proofs, motivating applications and examples of DR-submodular functions are provided in the Appendix of the full version of the paper (Raut, Sadeghi, and Fazel 2021).

1.1 Notations

$[T]$ is used to denote the set $\{1, 2, \dots, T\}$. For $u \in \mathbb{R}$, we define $[u]_+ := \max\{u, 0\}$. For a vector $x \in \mathbb{R}^n$, we use x_i to denote the i-th entry of x. The inner product of two vectors $x, y \in \mathbb{R}^n$ is denoted by either $\langle x, y\rangle$ or $x^T y$. Also, for two vectors $x, y \in \mathbb{R}^n$, $x \preceq y$ implies that $x_i \leq y_i \ \forall i \in [n]$. A function $f : \mathbb{R}^n \to \mathbb{R}$ is called monotone if for all x, y such that $x \preceq y$, $f(x) \leq f(y)$ holds. For a vector $x \in \mathbb{R}^n$, we use $\|x\|$ to denote the Euclidean norm of x. The unit ball of the Euclidean norm is denoted by $\mathcal{B}$, i.e., $\mathcal{B} = \{x \in \mathbb{R}^n \mid \|x\| \leq 1\}$. For a convex set $\mathcal{X}$, we will use $\mathcal{P}_{\mathcal{X}}(y) = \arg\min_{x\in\mathcal{X}} \|x-y\|$ to denote the projection of y onto set $\mathcal{X}$. The Fenchel conjugate of a function $f : \mathbb{R}^n \to \mathbb{R}$ is defined as $f^*(y) = \sup_x (x^T y - f(x))$.

2 Preliminaries

2.1 DR-Submodular Functions

Definition 1. *A differentiable function $f : \mathcal{X} \to \mathbb{R}$, $\mathcal{X} \subset \mathbb{R}^n_+$, is called DR-submodular if*

$$x \succeq y \Rightarrow \nabla f(x) \preceq \nabla f(y).$$

In other words, ∇f is element-wise decreasing and satisfies the DR (Diminishing Returns) property.

If f is twice differentiable, the DR property is equivalent to the Hessian matrix being element-wise non-positive. Note that for $n = 1$, the DR property is equivalent to concavity. However, for $n > 1$, concavity corresponds to negative semidefiniteness of the Hessian matrix (which is not equivalent to the Hessian matrix being element-wise non-positive). DR-submodular functions are also known as "smooth submodular" in the submodularity literature (e.g., see Vondrak (2008)). Bian et al. (2017) showed that a DR-submodular function f is concave along any non-negative and any non-positive direction; that is, if $t \geq 0$ and $v \in \mathbb{R}^n$ satisfies $v \succeq 0$ or $v \preceq 0$, we have

$$f(x + tv) \leq f(x) + t\langle \nabla f(x), v\rangle.$$

See Appendix A for examples of non-concave DR-submodular functions.

3 Problem Statement

Consider the following overall offline optimization problem:

$$\begin{array}{ll} \text{maximize} & \sum_{t=1}^T f_t(x_t) \\ \text{subject to} & x_t \in \mathcal{X},\ \forall t \in [T] \\ & \sum_{t=1}^T \langle p, x_t\rangle \leq B_T. \end{array} \tag{1}$$

The online setup is as follows: At each round $t \in [T]$, the algorithm chooses an action $x_t \in \mathcal{X}$, where $\mathcal{X} \subset \mathbb{R}^n_+$ is a fixed, known set. Upon committing to this action, the utility function $f_t : \mathcal{X} \to \mathbb{R}_+$ and a random i.i.d. sample $p_t \sim \mathcal{D}(p, \Sigma)$ are revealed and the algorithm receives a reward of $f_t(x_t)$ while using $\langle p, x_t\rangle$ of its fixed total allotted budget B_T. The overall goal is to maximize the total obtained reward while satisfying the budget constraint asymptotically (i.e., $\sum_{t=1}^T \langle p, x_t\rangle - B_T$ being sub-linear in T). Mahdavi, Yang, and Jin (2012) considered a similar setup and performance metric for the special case of linear utility functions.

Note that our proposed algorithm can handle multiple linear constraints as well, and similar regret and constraint violation bounds can be derived. However, for ease of notation, we focus on the case with only one linear constraint.

We make the following assumptions about our problem framework:

A1. The domain $\mathcal{X} \subset \mathbb{R}^n_+$ is a closed, bounded, convex set containing the origin, i.e., $0 \in \mathcal{X}$. We denote the diameter of $\mathcal{X}$ with R; i.e., $R := \max_{x,y\in\mathcal{X}} \|y - x\|$.

A2. For all $t \in [T]$, the utility function $f_t(\cdot)$ is normalized (i.e., $f_t(0) = 0$), monotone, DR-submodular, β_f-Lipschitz and L-smooth. In other words, for all $x, y \in \mathcal{X}$ and $u \in \mathbb{R}^n$ where $u \succeq 0$ or $u \preceq 0$, the following holds:

$$f_t(x + u) - f_t(x) \geq \langle u, \nabla f_t(x)\rangle - \frac{L}{2}\|u\|^2$$
$$|f_t(y) - f_t(x)| \leq \beta_f \|y - x\|.$$

A3. For all $t \in [T]$, $p_t \in \mathbb{R}^n_+$ is i.i.d. generated from the distribution $\mathcal{D}$ with bounded support $\beta_p\mathcal{B} \cap \mathbb{R}^n_+$, mean $p \succeq 0$ and covariance matrix Σ, i.e., $p_t \sim \mathcal{D}(p, \Sigma)$.

Let $\beta = \max\{\beta_f, \beta_p\}$. Under the above assumptions, we have:

$$F := \max_{t\in[T]} \max_{x,y\in\mathcal{X}} |f_t(x) - f_t(y)| \leq \beta R < \infty$$

$$G := \max_{p'\sim\mathcal{D}(p,\Sigma)} \max_{x\in\mathcal{X}} |\langle p', x\rangle - \frac{B_T}{T}| \leq \beta R - \frac{B_T}{T} < \infty.$$

3.1 Performance Metric

We characterize the performance of our proposed algorithm through bounding the notions of regret and cumulative constraint violation which are defined below:

Definition 2. *The* $(1-\frac{1}{e})$*-regret is defined as:*

$$R_T = (1-\frac{1}{e})\max_{x\in\mathcal{X}^*}\sum_{t=1}^{T} f_t(x) - \sum_{t=1}^{T} f_t(x_t),$$

where:

$$\mathcal{X}^* = \{x\in\mathcal{X} : \sum_{t=1}^{T}\langle p, x\rangle \leq B_T\} = \{x\in\mathcal{X} : \langle p, x\rangle \leq \frac{B_T}{T}\}.$$

The regret metric R_T quantifies the difference of the reward obtained by the algorithm and the $(1-\frac{1}{e})$-approximation of the reward of the best fixed benchmark action that has access to all the utility functions $f_t\ \forall t\in[T]$, the *mean* p of the linear constraint functions, and satisfies the cumulative budget constraint. Note that $1-\frac{1}{e}$ is the optimal approximation ratio for offline continuous DR-submodular maximization; in other words, even if all the online input were available beforehand, we could only obtain a $(1-\frac{1}{e})$ fraction of the maximum reward in polynomial time. The $(1-\frac{1}{e})$-regret is commonly used in the online submodular maximization literature (Chen, Hassani, and Karbasi 2018).

Definition 3. *The cumulative constraint violation is defined as follows:*

$$C_T = \sum_{t=1}^{T}\langle p, x_t\rangle - B_T.$$

Note that since $p_t\ \forall t\in[T]$ is i.i.d. drawn from the distribution $\mathcal{D}$ with mean p, our cumulative constraint violation metric C_T is defined with respect to the true underlying fixed linear constraint p (as opposed to p_t).

3.2 Related Work

Consider the following general framework of online problems with long-term constraints: At round $t\in[T]$, the player chooses $x_t\in\mathcal{X}$. Then, cost (utility) function $f_t:\mathcal{X}\to\mathbb{R}$ ($\mathcal{X}$ is a fixed convex set) and constraint function $g_t:\mathcal{X}\to\mathbb{R}$ are revealed, the player incurs a loss (obtains a reward) of $f_t(x_t)$ and uses the amount $g_t(x_t)$ of her budget (with the long-term constraint $\sum_{t=1}^{T} g_t(x_t)\leq 0$). This problem has been extensively studied under various assumptions where the cost (utility) functions are adversarially chosen and are assumed to be linear, convex or DR-submodular and the constraint functions are linear or convex and are either fixed (i.e., $g_t(\cdot)=g(\cdot)\ \forall t\in[T]$), stochastic and i.i.d drawn from some unknown distribution, or adversarial. For the setting with adversarial utility and constraint functions, Mannor, Tsitsiklis, and Yu (2009) provided a simple counterexample to show that regardless of the decisions of the algorithm, it is impossible to guarantee sub-linear regret against the benchmark action while the overall budget violation is sub-linear. Therefore, prior works in this setting have further restricted the fixed comparator action to be chosen from $\mathcal{X}_W = \{x\in\mathcal{X} : \sum_{\tau=t}^{t+W-1} g_\tau(x)\leq 0,\ 1\leq t\leq T-W+1\}$. In other words, in addition to merely satisfying the overall cumulative constraint (which corresponds to the $W=T$ case), the benchmark action is required to satisfy the budget constraint proportionally over any window of length W. On the other hand, for fixed or stochastic constraint functions, sub-linear regret and constraint violation bounds have been derived in the literature. A summary of the state of the art results for online problems with long-term constraints is provided in Table 1.

4 Online Lagrangian Frank-Wolfe (OLFW) Algorithm

In this section of the paper, we first introduce our proposed algorithm, namely the Online Lagrangian Frank-Wolfe (OLFW) algorithm, in Section 4.1 and subsequently, we analyze the performance of the algorithm with high probability and in expectation in Section 4.2 and Section 4.3 respectively.

4.1 Algorithm

The Online Lagrangian Frank-Wolfe (OLFW) algorithm is presented in Algorithm 1. First, note that for all $t\in[T]$, $x_t=\frac{1}{K}\sum_{k=1}^{K} v_t^{(k)}$ is the average of vectors in the convex domain $\mathcal{X}$ and hence, $x_t\in\mathcal{X}$. The intuition for using K online maximization subroutines to update x_t is the Frank-Wolfe variant proposed in Bian et al. (2017) to obtain the optimal approximation guarantee of $1-\frac{1}{e}$ for solving the offline DR-submodular maximization problem without the additional linear constraints. To be more precise, consider the first iteration $t=1$ of our online setting (ignoring the linear cumulative constraints) and the corresponding DR-submodular utility function $f_1(\cdot)$ arriving at this step. Note that f_1 is not revealed until the algorithm commits to an action $x_1\in\mathcal{X}$. If we were in the offline setting, we could use the mentioned Frank-Wolfe variant of Bian et al. (2017), run it for K iterations and maximize f_1 over $\mathcal{X}$. Starting from $x_1^{(1)}=0$, for all $k\in[K]$, we would find a vector $v_1^{(k)}$ that maximizes $\langle x, \nabla f_1(x_1^{(k)})\rangle$ over $x\in\mathcal{X}$, perform the update $x_1^{(k+1)}=x_1^{(k)}+\frac{1}{K}v_1^{(k)}$ and derive $x_1=x_1^{(K+1)}$ as the output. However, in the online setting, the utility function f_1 is not available before committing to the action x_1. Therefore, for each $k\in[K]$, we instead use a separate instance of a no-regret online linear maximization algorithm to obtain $v_1^{(k)}$. We repeat the same process for the subsequent utility functions $\{f_t\}_{t>1}$. This intuition was first provided in Chen, Hassani, and Karbasi (2018) and they managed to obtain an $\mathcal{O}(\sqrt{T})$ regret bound for the unconstrained online monotone submodular maximization problem.

Our choice of Lagrangian function is inspired by the quadratic penalty method in constrained optimization (Nocedal and Wright 1999). The penalized formulation of the overall optimization problem (1) with quadratic penalty

Paper	Cost (utility)	Constraint	Window size	R_T	C_T
Yuan and Lamperski (2018)	convex	convex (fixed)	—	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(T^{\frac{3}{4}})$
Wei, Yu, and Neely (2020)	convex	convex (stochastic)	—	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\sqrt{T})$
Neely and Yu (2017)	convex	convex (adversarial)	1	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(\sqrt{T})$
Liakopoulos et al. (2019)$^{(a)}$	convex	convex (adversarial)	W	$\mathcal{O}(\sqrt{T}+\frac{WT}{V})$	$\mathcal{O}(\sqrt{VT})$
Sadeghi and Fazel (2020)	DR-submodular	linear (adversarial)	W	$\mathcal{O}(\sqrt{WT})$	$\mathcal{O}(W^{\frac{1}{4}}T^{\frac{3}{4}})$

Table 1: State of the art results for online problems with cumulative constraints in various settings. Note that in (a), $V \in (W, T)$ is a tunable parameter.

function could be written as follows:

$$\max_{x_t} \sum_{t=1}^{T} f_t(x_t) - \frac{1}{2\delta\mu}\Big(\sum_{t=1}^{T}\langle p, x_t\rangle - B_T\Big)^2$$
$$\text{subject to } \; x_t \in \mathcal{X} \; \forall t \in [T].$$

Considering that the Fenchel conjugate of the function $h(\cdot) = \frac{1}{2\delta\mu}(\cdot)^2$ is $h^*(\cdot) = \frac{\delta\mu}{2}(\cdot)^2$, we can write the above problem in the following equivalent form:

$$\max_{x_t}\min_{\lambda} \sum_{t=1}^{T} f_t(x_t) - \lambda\Big(\sum_{t=1}^{T}\langle p, x_t\rangle - B_T\Big) + \frac{\delta\mu}{2}\lambda^2$$
$$\text{subject to } \; x_t \in \mathcal{X} \; \forall t \in [T].$$

Therefore, the corresponding Lagrangian function at round $t \in [T]$ is $\mathcal{L}_t(x, \lambda) = f_t(x) - \lambda(\langle p, x\rangle - \frac{B_T}{T}) + \frac{\delta\mu}{2}\lambda^2$. However, p is unknown to the online algorithm. Therefore, we alternatively use $\widehat{p}_t := \frac{1}{t-1}\sum_{s=1}^{t-1} p_s$ instead of p in the Lagrangian function. Note that $\widehat{p}_t$ is the empirical estimation of p at round t.

We first provide a lemma which is central to obtaining the regret and constraint violation bounds both in expectation and with high probability.

Lemma 1. *Let $x \in \mathcal{X}$ be a fixed vector. In the OLFW algorithm, set $\delta = \beta^2$. We then have:*

$$\sum_{t=1}^{T}\Big((1-\frac{1}{e})f_t(x) - f_t(x_t)\Big) \leq \frac{LR^2T}{2K} + \frac{R^2}{\mu} + \beta^2\mu T + \sum_{t=1}^{T}\lambda_t \widetilde{g}_t(x). \quad (2)$$

4.2 Performance Analysis with High Probability

In order to analyze the performance of the OLFW algorithm with high probability, the following lemmas detailing the concentration inequalities for the stochastic linear constraints are used.

Lemma 2. *The following holds with probability at least $1-\epsilon$:*

$$\sum_{t=2}^{T}\|\widehat{p}_t - p\| \leq C\sigma\sqrt{T\log\big(\frac{2nT}{\epsilon}\big)}.$$

Lemma 3. *Let $x \in \mathcal{X}$ be fixed. Define $\widehat{g}_t(x) := \langle\widehat{p}_t, x\rangle - \frac{B_T}{T}$ and $g(x) := \langle p, x\rangle - \frac{B_T}{T}$. For a fixed $t \in \{2, 3, \ldots, T\}$ and $\{\gamma_t := \sqrt{\frac{2G^2\log(\frac{2T}{\epsilon})}{t}}\}_{t=2}^{T}$, $|\widehat{g}_t(x) - g(x)| \leq \gamma_t$ holds with probability at least $1-\frac{\epsilon}{T}$.*

Algorithm 1 Online Lagrangian Frank-Wolfe (OLFW)

Input: $\mathcal{X}$ is the constraint set, T is the horizon, $\mu > 0$, $\delta > 0$, $\{\gamma_t\}_{t=1}^{T}$ and K.
Output: $\{x_t : 1 \leq t \leq T\}$.
Initialize K instances $\{\mathcal{E}_k\}_{k\in[K]}$ of Online Gradient Ascent with step size μ for online maximization of linear functions over $\mathcal{X}$.
for $t = 1$ **to** T **do**
  $x_t^{(1)} = 0$.
  for $k = 1$ **to** K **do**
    Let $v_t^{(k)}$ be the output of oracle $\mathcal{E}_k$ from round $t-1$.
    $x_t^{(k+1)} = x_t^{(k)} + \frac{1}{K}v_t^{(k)}$.
  end for
  Set $x_t = x_t^{(K+1)}$.
  Let $\widehat{p}_t := \frac{1}{t-1}\sum_{s=1}^{t-1} p_s$ for $t > 1$.
  Let
$$\widetilde{g}_t(\cdot) = \begin{cases}\langle\widehat{p}_t, \cdot\rangle - \frac{B_T}{T} & \text{(I)}\\ \langle\widehat{p}_t, \cdot\rangle - \frac{B_T}{T} - \gamma_t & \text{(II)}\end{cases}.$$
  Set $\lambda_t = \frac{[\widetilde{g}_t(x_t)]_+}{\delta\mu}$ for $t > 1$ and 0 otherwise.
  Play x_t and observe the Lagrangian function $\mathcal{L}_t(x_t, \lambda_t) = f_t(x_t) - \lambda_t\widetilde{g}_t(x_t) + \frac{\delta\mu}{2}\lambda_t^2$.
  for $k = 1$ **to** K **do**
    Feedback $\langle v_t^{(k)}, \nabla_x\mathcal{L}_t(x_t^{(k)}, \lambda_t)\rangle$ as the payoff to be received by $\mathcal{E}_k$.
  end for
end for

Proof. First, note that $\mathbb{E}[\widehat{g}_t(x)] = \mathbb{E}[\langle\widehat{p}_t, x\rangle - \frac{B_T}{T}] = \langle p, x\rangle - \frac{B_T}{T} = g(x)$. If $y_t = g_t(x)$ is a random variable, then by assumption, $y_t \in [-G, G]$ holds for each t, i.e., y_t is a bounded random variable. Therefore we can apply Hoeffding's inequality and get $\mathbb{P}\{|\widehat{g}_t(x) - g(x)| > \gamma_t\} \leq 2\exp(-\frac{t\gamma_t^2}{2G^2})$. Substituting the value of γ_t in the right hand side, we get that $\mathbb{P}\{|\widehat{g}_t(x) - g(x)| > \gamma_t\} \leq \frac{\epsilon}{T}$. The result follows immediately. ■

Now, we have all the machinery to obtain the high probability performance bounds of the OLFW algorithm.

Theorem 1. ***(High probability regret bound)*** *Let $\epsilon \in (0, 1)$ be given. Set $\mu = \frac{R}{\beta\sqrt{T}}$, $K = \sqrt{T}$, $\delta = \beta^2$ and $\{\gamma_t\}_{t=2}^{T}$ be chosen according to Lemma 3. Then, the OLFW algorithm* *with update (II) for $\widetilde{g}_t(\cdot)$ obtains the following regret bound with probability at least $1-\epsilon$.*

$$R_T \le \big(\frac{LR^2}{2} + 2R\beta\big)\sqrt{T}.$$

Proof. We begin from Lemma 1. Substitute the benchmark $x = x^*$ as the fixed vector in (2) and the constants as given in the hypothesis. We get: $R_T \le (\frac{LR^2}{2} + 2R\beta)\sqrt{T} + \sum_{t=1}^T \lambda_t \widetilde{g}_t(x^*)$. Now let us bound $\sum_{t=1}^T \lambda_t \widetilde{g}_t(x^*)$. From Lemma 3, we have that with probability at least $1-\frac{\epsilon}{T}$, $\widehat{g}_t(x^*) - \gamma_t \le g(x^*)$ holds, i.e., $\widetilde{g}_t(x^*) \le g(x^*)$. Also, $g(x^*) \le 0$ holds according to the definition of the benchmark action. Therefore, we have: $\widetilde{g}_t(x^*) \le 0$. As $\lambda_t \ge 0$, $\lambda_t \widetilde{g}_t(x^*) \le 0$ holds. Now, taking union bound over all $t \in [T]$, we have with probability at least $1-\epsilon$ that $\sum_{t=1}^T \lambda_t \widetilde{g}_t(x^*) \le 0$. The result follows immediately. ∎

We will use the following lemma to get performance bounds for the constraint violation.

Lemma 4. *Let $\{\gamma_t\}_{t=2}^T$ be defined as in Lemma 3, then the following holds.*

$$C_T \le \sum_{t=1}^T [\widetilde{g}_t(x_t)]_+ + R\sum_{t=1}^T \|\widehat{p}_t - p\| + \sum_{t=2}^T \gamma_t,$$

where $\widetilde{g}_t(\cdot)$ is derived using update (II).

Theorem 2. ***(High probability constraint violation bound)*** *Let $\epsilon \in (0,1)$ be given. Set $\mu = \frac{R}{\beta\sqrt{T}}$, $K = \sqrt{T}$, $\delta = \beta^2$ and $\{\gamma_t\}_{t=2}^T$ be chosen according to Lemma 3. Then the following holds with probability at least $1-\epsilon$ for the OLFW algorithm with update rule (II).*

$$C_T \le \sqrt{2G^2T\log(\frac{2T}{\epsilon})} + CR\sigma\sqrt{T\log(\frac{2nT}{\epsilon})} + \frac{T}{B_T}R\beta F\sqrt{T} + \frac{TR\beta}{B_T}(\frac{LR^2}{2} + 2R\beta) + R\beta.$$

So, we obtain $\tilde{\mathcal{O}}(\sqrt{T})$ constraint violation bound with high probability.

Proof. We begin with Lemma 1 again but now substitute $x = 0$ as the fixed vector in (2).

$$\frac{B_T}{T}\sum_{t=1}^T \lambda_t + \sum_{t=1}^T \lambda_t\gamma_t \le \underbrace{\sum_{t=1}^T f_t(x_t)}_{\le FT} + \frac{LR^2T}{2K} + \frac{R^2}{\mu} + \beta^2\mu T. \tag{3}$$

Rearranging and substituting the values of input parameters as given in the hypothesis, we get:

$$\sum_{t=1}^T [\widetilde{g}_t(x_t)]_+ + \frac{T\delta\mu}{B_T}\sum_{t=1}^T \lambda_t\gamma_t \le \frac{T}{B_T}R\beta F\sqrt{T} + \frac{TR\beta}{B_T}(\frac{LR^2}{2} + 2R\beta).$$

Both terms in the left hand side of the above equation are positive. Thus, we can drop the second term. We have:

$$\sum_{t=1}^T [\widetilde{g}_t(x_t)]_+ \le \frac{T}{B_T}R\beta F\sqrt{T} + \frac{TR\beta}{B_T}(\frac{LR^2}{2} + 2R\beta).$$

Combining Lemma 4 and the equation above, we obtain:

$$C_T \le \frac{T}{B_T}R\beta F\sqrt{T} + \frac{TR\beta}{B_T}(\frac{LR^2}{2} + 2R\beta) + R\sum_{t=1}^T \|\widehat{p}_t - p\| + \sum_{t=2}^T \gamma_t.$$

Therefore, we can conclude:

$$C_T \le \frac{T}{B_T}R\beta F\sqrt{T} + \frac{TR\beta}{B_T}(\frac{LR^2}{2} + 2R\beta) + \sqrt{2G^2T\log(\frac{2T}{\epsilon})} + \underbrace{R\sum_{t=1}^T \|\widehat{p}_t - p\|}_{\text{(A)}},$$

where the last inequality follows from summing γ_t's. Now, Lemma 2 tells us that (A) $\le R\beta + CR\sigma\sqrt{T\log(\frac{2nT}{\epsilon})}$ holds with probability at least $1-\epsilon$. Thus, we get the result. ∎

Theorem 1 and Theorem 2 are indeed the first high probability bounds obtained for the online DR-submodular maximization problem with stochastic cumulative constraints. Note that the $\mathcal{O}(\sqrt{T})$ regret bound obtained in Theorem 1 is known to be optimal.

4.3 Performance Analysis in Expectation

We first provide a simple lemma that will be used throughout the analysis in expectation.

Lemma 5. *For $t > 1$, we have:*

$$\mathbb{E}\|\widehat{p}_t - p\|^2 = \frac{Tr(\Sigma)}{t-1},$$

where $Tr(\Sigma)$ denotes the trace of the covariance matrix Σ.

Now, we present the main performance bounds in expectation, namely the expected regret bound and the expected cumulative constraint violation bound. In the Appendix, we have also considered the case where we only have access to unbiased stochastic gradient estimates of the utility functions $\{f_t\}_{t=1}^T$ and exact gradient computation is not possible. For this setting, we modify the OLFW algorithm through incorporating the variance reduction technique introduced by Chen et al. (2018) and we obtain similar regret and constraint violation bounds in expectation for the modified algorithm.

Theorem 3. ***(Expected regret bound)*** *The regret bound of the OLFW algorithm with update rule (I) is the following:*

$$\mathbb{E}[R_T] \le \tilde{\mathcal{O}}(T^{\frac{3}{4}}).$$

Proof. We first observe from (3) that $\sum_{t=1}^{T}\lambda_t \leq \mathcal{O}(T)$. Now, substitute $x = x^*$, the benchmark, in (2) and take expectation on both sides to obtain:

$$\begin{aligned}\mathbb{E}[R_T] \leq &\frac{LR^2T}{2K} + \frac{R^2}{\mu} + \beta^2\mu T \\ &+ \mathbb{E}[\sum_{t=1}^{T}\lambda_t(\widehat{g}_t(x^*) - g(x^*))] + \sum_{t=1}^{T}\lambda_t g(x^*) \\ \leq &\frac{LR^2T}{2K} + \frac{R^2}{\mu} + \beta^2\mu T + \underbrace{\mathbb{E}\,[\sum_{t=1}^{T}\lambda_t(\widehat{g}_t(x^*) - g(x^*))]}_{\text{(B)}}.\end{aligned}$$

Now we bound (B) as follows:

$$\begin{aligned}\text{(B)} &= \sum_{t=1}^{T}\lambda_t(\widehat{g}_t(x^*) - g(x^*)) \\ &\leq \sqrt{\sum_{t=1}^{T}\lambda_t^2}\sqrt{\sum_{t=1}^{T}(\widehat{g}_t(x^*) - g(x^*))^2} \\ &= \sqrt{\|\lambda\|^2}\sqrt{\sum_{t=1}^{T}(\langle\widehat{p}_t - p, x^*\rangle)^2} \\ &\leq \|\lambda\|\sqrt{\sum_{t=1}^{T}\|\widehat{p}_t - p\|^2R^2} \\ &= R\|\lambda\|\sqrt{\sum_{t=1}^{T}\|\widehat{p}_t - p\|^2}.\end{aligned}$$

Both the inequalities above are obtained using Cauchy-Schwarz inequality, where $\lambda := [\lambda_1, \lambda_2, \ldots, \lambda_T]^T$.

Using the Cauchy-Schwarz inequality again, we have $\|\lambda\| \leq \sqrt{\|\lambda\|_1\|\lambda\|_\infty}$. Thus, we obtain $\|\lambda\| \leq \sqrt{(\sum_{t=1}^{T}\lambda_t)(\frac{G}{\delta\mu})} \leq \mathcal{O}(T^{3/4})$. Therefore, the following holds:

$$\|\lambda\| \leq \mathcal{O}(T^{3/4}). \tag{4}$$

Using Jensen's inequality, we have

$$\mathbb{E}\sqrt{\sum_{t=1}^{T}\|\widehat{p}_t - p\|^2} \leq \sqrt{\sum_{t=1}^{T}\mathbb{E}\|\widehat{p}_t - p\|^2}.$$

We can use Lemma 5 and write:

$$\mathbb{E}\sqrt{\sum_{t=1}^{T}\|\widehat{p}_t - p\|^2} \leq \sqrt{Tr(\Sigma)\log(T)}. \tag{5}$$

Thus, combining (4) and (5), we obtain:

$$\mathbb{E}[\text{(B)}] \leq \mathcal{O}(T^{3/4}\sqrt{Tr(\Sigma)\log(T)}).$$

The result thus follows. ■

Remark. The main challenge in bounding R_T in expectation is the fact that in our algorithm, the choice of λ_t is dependent on $\widehat{p}_t$ and thus, we cannot use

$$\mathbb{E}[\lambda_t\widehat{g}_t(x^*)] = \mathbb{E}[\lambda_t\mathbb{E}[\widehat{g}_t(x^*)]] = \mathbb{E}[\lambda_t g(x^*)] \leq 0$$

and this term is indeed the dominating term in the regret bound. However, as we saw earlier, we do not encounter this problem in the high probability setting due to subtracting γ_t from all the constraint functions and using the concentration inequalities, and thus we were able to obtain $\mathcal{O}(\sqrt{T})$ high probability regret bound.

Theorem 4. ***(Expected cumulative constraint violation bound)*** *For the OLFW algorithm with update rule (I), we have:*

$$\begin{aligned}\mathbb{E}[C_T] \leq &\frac{T}{B_T}R\beta F\sqrt{T} + R\sqrt{Tr(\Sigma)}\sqrt{T} \\ &+ \frac{TR\beta}{B_T}(\frac{LR^2}{2} + 2R\beta) + R\beta.\end{aligned}$$

Therefore, $\mathbb{E}[C_T] \leq \tilde{\mathcal{O}}(\sqrt{T})$ *holds.*

Theorem 3 and Theorem 4 provide the first sub-linear expectation bounds on the regret and cumulative constraint violation of the online DR-submodular maximization problem with stochastic cumulative constraints.

5 Numerical Results

We conduct numerical experiments, over simulated and real-world datasets in the following.

Joke Recommendation We look at the problem of DR-submodular function maximization over the *Jester* dataset[1]. We consider a fraction of the dataset where there are 100 jokes and user ratings from 10000 users are available for these jokes. The ratings take values in $[-10, 10]$, we re-scale them to be in $[0, 10]$. Let $R_{u,j}$ be the rating of user u for joke j. As some of the user ratings are missing in the dataset, we set such ratings to be 5. In the online setting, a user arrives and we have to recommend at most $M = 15$ jokes to her. The utility function for each round $t \in [T]$ is of the form $f_t(x) = \sum_{i=1}^{100} R^t_{u_t,i}x_i + \sum_{i,j:i\neq j}\theta_{ij}x_ix_j$, where u_t is the user being served in the current round. $\{\theta_{ij}\}_{i\neq j}$ are chosen such that the function is monotone. These DR-submodular utility functions capture the overall impression of the displayed jokes on the user. There is a limited total time (denoted by $B_T = 1.5T$) available to recommend the jokes to the users. For all $i \in [n]$, p_i denotes the average time it takes to read joke i. As some jokes are relatively longer, we do not want the user to spend more time on jokes which do not lead to larger utility. The linear budget functions are chosen randomly with entries uniformly drawn from $[0.03, 0.35]$. We compare the performance of our algorithm against the following strategies:

- *Uniform*: At every round, we assign 15 randomly chosen jokes to the user.

[1]http://eigentaste.berkeley.edu/dataset/

- *Greedy*: We deploy an exploration-exploitation strategy where with probability 0.1, we randomly assign 15 jokes and with probability 0.9, we present the top 15 jokes based on the ratings observed so far.
- *Meta-FW* (Chen, Hassani, and Karbasi 2018): This corresponds to solving the unconstrained DR-submodular maximization problem (i.e., ignoring the budget constraints).
- *Budget-Cautious*: At each round, we assign 15 jokes which have the lowest average budget consumption observed so far.

The results are presented in Figure 1. As it can be seen in the plots, our OLFW algorithm obtains a reasonable utility while approximately satisfying the budget constraint as well.

Indefinite quadratic functions. We choose $\mathcal{X} = \{x \in \mathbb{R}^2 : 0 \preceq x \preceq 1\}$ and for each $t \in [T]$, we generate quadratic functions of the form $f_t(x) = \frac{1}{2}x^T H_t x + h_t^T x$ where $H_t \in \mathbb{R}^{2\times 2}$ is a random matrix whose entries are chosen uniformly from $[-1, 0]$. We let $h_t = -H_t^T \mathbf{1}$ to ensure the monotonocity of the objective. We let $T = 1000$. At each round, we randomly generate linear budget functions whose entries are chosen uniformly from $[0.5, 2.5]$ and the mean vector is $p = [1, 2]^T$. Also, we set the total budget to be $B_T = 2T$. We run the OLFW algorithm 10 times and take the respective averages for the cumulative utility and total remaining budget. We vary δ, the parameter of the penalty function, in the range $[0.1, 1000]$ and plot the trade-off curve $\big($i.e., $\sum_{t=1}^{1000} f_t(x_t)$ versus $B_T - \sum_{t=1}^{1000}\langle p, x_t\rangle\big)$ for 100 chosen values of δ in Figure 2. In this example, our choice of δ in the OLFW algorithm, highlighted in the plot, achieves the highest possible cumulative utility while satisfying the total budget constraint.

Log-determinant functions. We choose $\mathcal{X} = \{x \in \mathbb{R}^{10} : 0 \preceq x \preceq 1\}$ and for each $t \in [T]$, we generate log-determinant functions of the form $f_t(x) = \log\det\big(\mathrm{diag}(x)(L_t - I) + I\big)$, where each L_t is a random positive definite matrix with eigenvalues falling in the range $[2, 3]$. The choice of eigenvalues ensures the monotonocity of the function. Let $T = 4900$. At each round, we generate linear budget functions whose entries are chosen uniformly from the range $[0.3, 5.7]$. We run the OLFW algorithm for different choices of the step size μ and plot the cumulative utility and the total budget violation in Figure 3. Our OLFW algorithm chooses μ such that the overall utility and cumulative budget consumption are balanced.

6 Conclusion and Future Work

In this work, we studied online continuous DR-submodular maximization with stochastic linear cumulative constraints. We proposed the Online Lagrangian Frank-Wolfe (OLFW) algorithm to solve this problem and we obtained the first sub-linear bounds, both in expectation and with high probability, for the regret and constraint violation of this algorithm. The current work could be further extended in a number of interesting directions. First, it is yet to be seen whether the online DR-submodular maximization setting could handle general, stochastic or adversarial, convex long-term constraints. Furthermore, it is interesting to see whether it is possible to improve the expected regret bound to match the $\mathcal{O}(\sqrt{T})$ high probability regret bound. Finally, studying this problem under bandit feedback (as opposed to the full information setting considered in this paper) is left to future work.

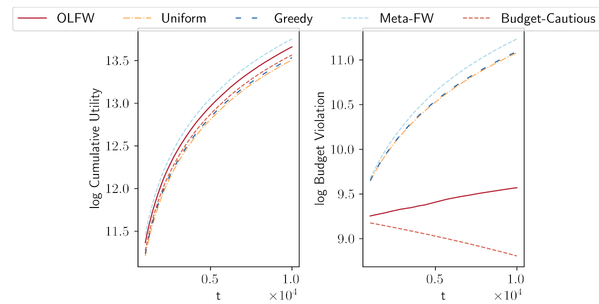

Figure 1: Comparison of the overall utility $\sum_{t=1}^{T} f_t(x_t)$ and cumulative budget violation $\sum_{t=1}^{T}\langle p, x_t\rangle - B_T$ for the Jester dataset.

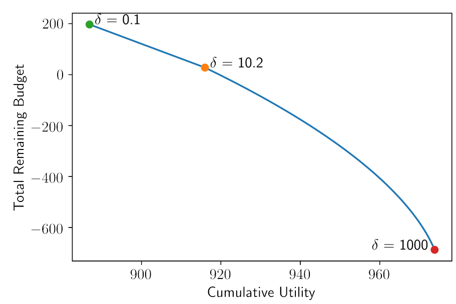

Figure 2: Trade-off between the overall utility $\sum_{t=1}^{T} f_t(x_t)$ and the total remaining budget $B_T - \sum_{t=1}^{T}\langle p, x_t\rangle$ of quadratic functions for different choices of parameter δ. $\delta = 10.2$ is our choice of the penalty parameter.

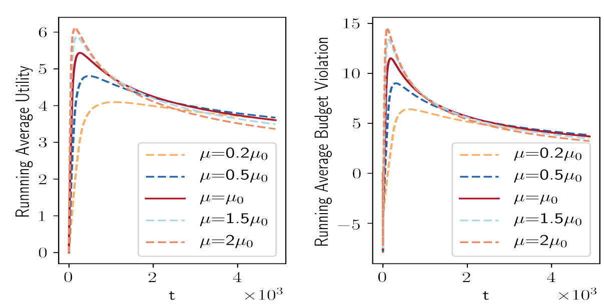

Figure 3: Running average of cumulative utility $\frac{1}{t}\sum_{\tau=1}^{t} f_\tau(x_\tau)$ and running average of budget violation $\frac{1}{t}\sum_{\tau=1}^{t}(\langle p, x_\tau\rangle - \frac{B_T}{T})$ for different choices of the step size μ, where $\mu_0 := \frac{R}{\beta\sqrt{T}}$ is our choice of step size in the OLFW algorithm.

Acknowledgements

This work was supported in part by the following grants: NSF TRIPODS grant 1740551, DARPA Lagrange grant FA8650-18-2-7836, ONR MURI grant N0014-16-1-2710.

References

Agrawal, S.; and Devanur, N. R. 2014. Bandits with Concave Rewards and Convex Knapsacks. In *Proceedings of the Fifteenth ACM Conference on Economics and Computation*, EC '14, 989–1006. New York, NY, USA: Association for Computing Machinery. ISBN 9781450325653. doi: 10.1145/2600057.2602844. URL https://doi.org/10.1145/2600057.2602844.

Agrawal, S.; and Devanur, N. R. 2015. Fast Algorithms for Online Stochastic Convex Programming. In *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, SODA '15, 1405–1424. USA: Society for Industrial and Applied Mathematics.

Badanidiyuru, A.; Kleinberg, R.; and Slivkins, A. 2018. Bandits with Knapsacks. *J. ACM* 65(3). ISSN 0004-5411. doi:10.1145/3164539. URL https://doi.org/10.1145/3164539.

Balseiro, S. R.; and Gur, Y. 2019. Learning in Repeated Auctions with Budgets: Regret Minimization and Equilibrium. *Management Science* 65(9): 3952–3968. doi:10.1287/mnsc.2018.3174. URL https://doi.org/10.1287/mnsc.2018.3174.

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

Bian, A. A.; Mirzasoleiman, B.; Buhmann, J.; and Krause, A. 2017. Guaranteed Non-Convex Optimization: Submodular Maximization over Continuous Domains. In Singh, A.; and Zhu, J., eds., *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, volume 54 of *Proceedings of Machine Learning Research*, 111–120. Fort Lauderdale, FL, USA: PMLR.

Chen, L.; Harshaw, C.; Hassani, H.; and Karbasi, A. 2018. Projection-free online optimization with stochastic gradient: From convexity to submodularity. In *International Conference on Machine Learning*, 814–823. PMLR.

Chen, L.; Hassani, H.; and Karbasi, A. 2018. Online Continuous Submodular Maximization. In Storkey, A.; and Perez-Cruz, F., eds., *Proceedings of the Twenty-First International Conference on Artificial Intelligence and Statistics*, volume 84 of *Proceedings of Machine Learning Research*, 1896–1905. Playa Blanca, Lanzarote, Canary Islands: PMLR.

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

Hazan, E. 2016. Introduction to Online Convex Optimization. *Found. Trends Optim.* 2(3–4): 157–325. ISSN 2167-3888. doi:10.1561/2400000013.

Liakopoulos, N.; Destounis, A.; Paschos, G.; Spyropoulos, T.; and Mertikopoulos, P. 2019. Cautious Regret Minimization: Online Optimization with Long-Term Budget Constraints. In Chaudhuri, K.; and Salakhutdinov, R., eds., *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, 3944–3952. Long Beach, California, USA: PMLR.

Mahdavi, M.; Yang, T.; and Jin, R. 2012. Efficient Constrained Regret Minimization. *arXiv:1205.2265 [cs]* ZSCC: 0000002.

Mannor, S.; Tsitsiklis, J. N.; and Yu, J. Y. 2009. Online learning with sample path constraints. *Journal of Machine Learning Research* 10(Mar): 569–590.

Neely, M. J.; and Yu, H. 2017. Online Convex Optimization with Time-Varying Constraints. *arXiv:1702.04783 [math]* .

Nocedal, J.; and Wright, S. J. 1999. *Numerical Optimization*. Springer Series in Operations Research. New York: Springer. ISBN 978-0-387-98793-4. ZSCC: 0000956.

Orabona, F. 2019. A modern introduction to online learning. *arXiv preprint arXiv:1912.13213* .

Raut, P. S.; Sadeghi, O.; and Fazel, M. 2021. Online DR-Submodular Maximization with Stochastic Cumulative Constraints. *arXiv:2005.14708v2 [math.OC]* .

Rivera, A.; Wang, H.; and Xu, H. 2018. The Online Saddle Point Problem and Online Convex Optimization with Knapsacks. *arXiv preprint arXiv:1806.08301* .

Sadeghi, O.; and Fazel, M. 2020. Online continuous DR-submodular maximization with long-term budget constraints. In *International Conference on Artificial Intelligence and Statistics*, 4410–4419. PMLR.

Shalev-Shwartz, S. 2012. Online Learning and Online Convex Optimization. *Found. Trends Mach. Learn.* 4(2): 107–194. ISSN 1935-8237. doi:10.1561/2200000018. URL https://doi.org/10.1561/2200000018.

Vondrak, J. 2008. Optimal Approximation for the Submodular Welfare Problem in the Value Oracle Model. In *Proceedings of the Fourtieth Annual ACM Symposium on Theory of Computing - STOC 08*, 67. Victoria, British Columbia, Canada: ACM Press. ISBN 978-1-60558-047-0. doi: 10.1145/1374376.1374389.

Wei, X.; Yu, H.; and Neely, M. J. 2020. Online primal-dual mirror descent under stochastic constraints. *Proceedings of the ACM on Measurement and Analysis of Computing Systems* 4(2): 1–36.

Yuan, J.; and Lamperski, A. 2018. Online convex optimization for cumulative constraints. In Bengio, S.; Wallach, H.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems 31*, 6137–6146. Curran Associates, Inc.

Zinkevich, M. 2003. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th international conference on machine learning (icml-03)*, 928–936.