Online Continuous DR-Submodular Maximization with Long-Term Budget Constraints

Omid Sadeghi
University of Washington

Maryam Fazel
University of Washington

Abstract

In this paper, we study a class of online optimization problems with long-term budget constraints where the objective functions are not necessarily concave (nor convex), but they instead satisfy the Diminishing Returns (DR) property. In this online setting, a sequence of monotone DR-submodular objective functions and linear budget functions arrive over time and assuming a limited total budget, the goal is to take actions at each time, before observing the utility and budget function arriving at that round, to achieve sub-linear regret bound while the total budget violation is sub-linear as well. Prior work has shown that achieving sub-linear regret and total budget violation simultaneously is impossible if the utility and budget functions are chosen adversarially. Therefore, we modify the notion of regret by comparing the agent against the best fixed decision in hindsight which satisfies the budget constraint proportionally over any window of length W. We propose the Online Saddle Point Hybrid Gradient (OSPHG) algorithm to solve this class of online problems. For $W = T$, we recover the aforementioned impossibility result. However, if W is sub-linear in T, we show that it is possible to obtain sub-linear bounds for both the regret and the total budget violation.

Proceedings of the 23rdInternational Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108.

1 Introduction

1.1 Motivating Application: Online Ad Placement

Consider the following online ad placement problem: At round $t \in [T]$, an advertiser should choose an investment vector $x_t \in R_+^n$ over n different websites where i-th entry of x_t denotes the amount that the advertiser is willing to pay per each click on the ad on the i-th website (i.e., cost per click). In other words, each website has different tiers of ads and choosing x_t corresponds to ordering a certain type of ad. The aggregate cost of investment would be determined when the number of clicks the ad receives is revealed. In other words, the cost of such an investment would be $\langle p_t, x_t \rangle$ where the i-th entry of the vector p_t is the number of clicks the ad on the i-th website receives. Note that the vector p_t is not known ahead of time and could be adversarial. For instance, competing advertisers may click on the ads to deplete their rival's budget. The advertiser needs to balance her total investment against an allotted long-term budget (daily, monthly, etc.), i.e., $\sum_{t=1}^{T} \langle p_t, x_t \rangle \leq B_T$ where B_T is the total targeted budget. At round $t \in [T]$, the advertiser's utility function $f_t(x_t)$ is a monotone DR-submodular function with respect to the vector of investments and this function quantifies the overall amount of impressions of the ads. DR-submodularity of the utility function characterizes the diminishing returns property of the impressions (Diminishing Returns (DR) property and continuous DR-submodular functions are defined in Section 2.2 at page 4). In other words, making an ad more visible will attract proportionally fewer extra viewers because each website shares a portion of its visitors with other websites. Liakopoulos et al. (2019) considered the online portfolio management problem (with online ad placement problem as their running example) and in order to model diminishing returns, they assumed that the utility functions are separable and concave. Note that concavity is equivalent to DR-submodularity for separable functions. However, for non-separable utility functions, DR-submodularity is the property that captures the diminishing returns of the objective functions (rather than the concavity property) and therefore, in this work, we focus on DR-submodular utility functions that are not necessarily concave. Our results resolve the open problem posed by Liakopoulos et al. (2019) in the footnote of the third page of their paper. See Appendix A for more applications.

In this paper, we propose an algorithm for this class of online non-convex problems such that the algorithm has no regret, i.e., a sub-linear regret bound with respect to the horizon T, while the total budget violation is sub-linear as well.

1.2 Related Work

Online convex optimization with constraints. Consider an online problem where at step $t \in [T]$, the player chooses $x_t \in \mathcal{X}$. Then, cost function $f_t : \mathcal{X} \to \mathbb{R}$ and constraint function $g_t : \mathcal{X} \to \mathbb{R}$ are revealed and the player incurs a loss of $f_t(x_t)$ and her budget is impacted by the amount $g_t(x_t)$. $\mathcal{X}$ is assumed to be convex and compact and the functions f_t, g_t are convex for all $t \in [T]$. The overall goal is to design an algorithm whose output is asymptotically feasible, i.e., the constraint residual $\sum_{t=1}^{T} g_t(x_t)$ is sub-linear, and has a sub-linear regret. Mahdavi et al. (2012) considered the case where all constraint functions are equal and are given offline, i.e., $g_t(x) = g(x)\ \forall t \in [T], x \in \mathcal{X}$. For this setting, they achieved $\mathcal{O}(\sqrt{T})$ regret and $\mathcal{O}(T^{\frac{2}{3}})$ constraint residual (i.e., $\sum_{t=1}^{T} g(x_t)$) bounds. Jenatton et al. (2016) studied the exact same framework as Mahdavi et al. (2012) and generalized their result by obtaining $\mathcal{O}(T^{\max\{\beta,1-\beta\}})$ regret and $\mathcal{O}(T^{1-\frac{\beta}{2}})$ constraint residual bounds where $\beta \in (0, 1)$ is a tunable parameter. More recently, Yuan and Lamperski (2018) considered an alternative notion of constraint residual defined as the sum of squares of clipped residuals, $\sum_{t=1}^{T}(\max\{g(x_t), 0\})^2$, and achieved $\mathcal{O}(T^{\max\{\beta,1-\beta\}})$ regret and $\mathcal{O}(T^{1-\beta})$ constraint residual bounds for this setting. Also, they obtained logarithmic regret bounds for the case that cost functions are strongly convex. The new constraint residual form considered in Yuan and Lamperski (2018) heavily penalizes large constraint violations and strictly feasible solutions of some rounds cannot cancel out the effect of violated constraints at other rounds.

For the setting with constraints arriving online[1], Mannor et al. (2009) considered the notion of regret with window length $W = T$ and provided a simple counterexample showing that the regret of any algorithm with sub-linear constraint residual is lower bounded by $\Omega(T)$. Neely and Yu (2017) assumed that there exists an action $x^* \in \mathcal{X}$ such that $g_t(x^*) < 0\ \forall t \in [T]$ (Slater condition) and under this assumption, they obtained $\mathcal{O}(\sqrt{T})$ bounds for both regret with window size $W = 1$ and constraint residual. However, the fixed decision benchmark action considered in this paper is restricted to be feasible for all constraint functions $g_t\ \forall t \in [T]$ which heavily restricts the performance of the benchmark action and thus, the obtained regret guarantees could be loose. Sun et al. (2017) considered the same notion of regret as Neely and Yu (2017) and using online mirror descent as a subroutine (and without assuming the Slater condition), they obtained a similar $\mathcal{O}(\sqrt{T})$ regret bound and a looser $\mathcal{O}(T^{\frac{3}{4}})$ constraint residual bound. Liakopoulos et al. (2019) considered the exact same framework and algorithm as Neely and Yu (2017), however, they constrained the benchmark action to be feasible in all windows of size $W = T^{\beta}$ where $\beta \in [0, 1)$ (as opposed to Neely and Yu, 2017 where $W = 1$). They obtained $\mathcal{O}(\frac{WT}{V} + \sqrt{T})$ regret bound and $\mathcal{O}(\sqrt{VT})$ residual bound where $V \in [W, T)$ is a tunable parameter which captures the trade-off between the regret and constraint residual bounds. Note that for $W = 1$, since the Slater condition is not assumed by Liakopoulos et al. (2019), their bound does not achieve the $\mathcal{O}(\sqrt{T})$ regret and constraint residual bound of Neely and Yu (2017).

Note that in all these works, the objective functions are assumed to be convex. In contrast, we consider a more general class of non-convex/non-concave DR-submodular objective functions to which the aforementioned results are not applicable.

Online submodular maximization. An orthogonal research direction considers the following problem: At step $t \in [T]$, the online algorithm chooses a feasible point $x_t \in \mathcal{P}$. Once the algorithm commits to this choice, a monotone continuous DR-submodular function f_t is revealed and the reward $f_t(x_t)$ is received. The goal is to minimize the regret defined as the difference between the total reward obtained by the algorithm and that of the $(1 - \frac{1}{e})$-approximation to the best fixed decision in hindsight with $(1 - \frac{1}{e})$ being the optimal polynomial time approximation ratio for an *offline* monotone continuous DR-submodular maximization problem (Bian et al., 2017a). Note that although similar to our framework (the objective functions are assumed to be continuous DR-submodular in this setting), there are no time-varying constraints arriving online and therefore, they do not deal with the considerable complication of bounding the constraint residual.

The meta-algorithm for online submodular maximization problem is presented in Algorithm 1. The intuition for using K online maximization subroutines to obtain x_t; $t \in [T]$ is the Frank-Wolfe variant proposed by Bian et al. (2017a) to obtain the optimal approximation guar-

[1] In this setting, the regret metric with window length W is defined as $R_T = \sum_{t=1}^{T} f_t(x_t) - \sum_{t=1}^{T} f_t(x_W^*)$ where $x_W^* = \arg\min_{x\in\mathcal{X}_W} \sum_{t=1}^{T} f_t(x)$ and $\mathcal{X}_W = \{x \in \mathcal{X} : \sum_{\tau=t}^{t+W-1} g_\tau(x) \le 0,\ 1 \le t \le T - W + 1\}$.

antee of $(1-\frac{1}{e})$ for solving the offline DR-submodular maximization problem. To be more precise, consider the first iteration $t=1$ of the online setting 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 $x_1 \in \mathcal{P}$. If f_1 were available offline, the mentioned Frank-Wolfe variant of Bian et al. (2017a) could have been used for K iterations to maximize f_1 over $\mathcal{P}$. Starting from $x_1^{(1)}=0$, for all $k \in [K]$, a vector $v_1^{(k)}$ would have been found that maximizes $\langle x, \nabla f_1(x_1^{(k)})\rangle$ over $x \in \mathcal{P}$ and using the update $x_1^{(k+1)} = x_1^{(k)} + \frac{1}{K} v_1^{(k)}$, $x_1 = x_1^{(K+1)}$ would have been derived 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 $k \in [K]$, a separate instance of a no-regret online linear maximization algorithm is used instead to obtain $v_1^{(k)}$. The same process is repeated for the subsequent utility functions $f_t;\ t>1$ as well.

Golovin et al. (2014) considered the case that the continuous DR-submodular function f_t is the multilinear extension of a discrete submodular function and $\mathcal{P}$ is the matroid polytope. Using the Perturbed Follow the Leader (PFTL) as the online algorithm, they achieved an $\mathcal{O}(\sqrt{T})$ bound for the $(1-\frac{1}{e})$-regret. Chen et al. (2018a) used Regularized Follow The Leader (RFTL) online algorithm and achieved a similar regret bound for general continuous DR-submodular functions. In Chen et al. (2018b), they further generalized their result and developed a projection-free algorithm which only requires stochastic gradient estimates of $f_t\ \forall t \in [T]$ and achieves the same regret guarantees. See Krause and Golovin (2014) for a detailed overview of online maximization of submodular set functions.

1.3 Contributions

In this paper, we design an algorithm for online continuous DR-submodular maximization problem with long-term budget constraints to achieve sub-linear regret and budget violation bounds simultaneously. Specifically, we make the following contributions:

- We introduce the online continuous DR-submodular maximization problem with long-term budget constraints. The online ad placement example mentioned in section 1.1 is an application of this framework. We also provide a number of other motivating applications for this framework in Appendix A.
- We propose the Online Saddle Point Hybrid Gradient (OSPHG) algorithm to solve this class of online problems. Our algorithm is inspired by that of Sun et al. (2017) and Chen et al. (2018a). We consider a refined notion of static regret where the agent's utility is compared against a $(1-\frac{1}{e})$-approximation to the best fixed decision in hindsight which satisfies the budget constraint proportionally over any window of length W. For $W=T$, we recover the known impossibility result obtained by Mannor et al. (2009). However, for $W=o(T)$, we obtain sub-linear bounds for both the $(1-\frac{1}{e})$-regret and the total budget violation. In particular, if $W=T^{1-\epsilon}$ for $0<\epsilon\leq 1$, we obtain a $(1-\frac{1}{e})$-regret bound of $\mathcal{O}(T^{1-\frac{\epsilon}{2}})$ while the total budget violation is $\mathcal{O}(T^{1-\frac{\epsilon}{4}})$.

Algorithm 1 Online submodular maximization meta-algorithm (Chen et al., 2018a)

Input: $\mathcal{P}$ is a convex set and T is the horizon.
Output: $\{x_t : 1 \leq t \leq T\}$.
Choose an off-the-shelf online linear maximization algorithm and initialize K instances $\mathcal{E}_k;\ k \in [K]$ of it for online maximization of linear utility functions over $\mathcal{P}$.
for $t=1$ **to** T **do**
  Set $x_t^{(1)}=0$.
  for $k=1$ **to** K **do**
    Let $v_t^{(k)}$ be the vector selected by $\mathcal{E}_k$.
    $x_t^{(k+1)} = x_t^{(k)} + \frac{1}{K} v_t^{(k)}$.
  end for
  Play $x_t = x_t^{(K+1)}$, observe the function f_t and the reward $f_t(x_t)$.
  For all $k \in [K]$, feedback $\langle v_t^{(k)}, \nabla f_t(x_t^{(k)})\rangle$ as the payoff to be received by $\mathcal{E}_k$.
end for

Finally, we illustrate the performance of our algorithm through numerical examples for a class of non-convex/non-concave continuous DR-submodular objective functions.

2 Preliminaries

2.1 Notation

We will use $[T]$ to denote the set $\{1,2,\ldots,T\}$. For $u \in \mathbb{R}$, we define $[u]_+ := \max\{u,0\}$. 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. 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 $f^*(y) = \sup_x (x^T y - f(x))$.

2.2 Diminishing Returns (DR) property

Definition 2.1 *A differentiable function $f : \mathcal{X} \to \mathbb{R}$, $\mathcal{X} \subset \mathbb{R}^n_+$, satisfies the Diminishing Returns (DR) property if:*

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

In other words, ∇f is an anti-tone mapping from $\mathbb{R}^n$ to $\mathbb{R}^n$.
If f is twice differentiable, 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 semi-definiteness of the Hessian matrix which is not equivalent to the Hessian matrix being element-wise non-positive.

A similar property is introduced by Vondrák (2008) and Bian et al. (2017a) as well and functions satisfying this property are called "smooth submodular" and "DR-submodular" there respectively. Additionally, Eghbali and Fazel (2016) defined the DR property for concave functions with respect to a partial ordering induced by a cone and showed that by taking the cone to be $\mathbb{R}^n_+$, Definition 2.1 is recovered. Eghbali and Fazel (2016) also showed that if the cone of positive semi-definite matrices is considered, the DR property generalizes to matrix ordering as well. Bian et al. (2017a) showed that DR-submodular functions are concave along any non-negative direction, and any non-positive direction. In other words, for a DR-submodular function f, if $t \geq 0$ and $v \in \mathbb{R}^n$ satisfies $v \succeq 0$ or $v \preceq 0$, we have:

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

2.3 Examples of continuous non-concave DR-submodular functions

Multilinear extension of discrete submodular functions. (Calinescu et al., 2007) A discrete function $F : \{0, 1\}^V \to \mathbb{R}$ is submodular if for all $j \in V$ and $A \subseteq B \subseteq V \setminus \{j\}$, the following holds:

$$F(A \cup \{j\}) - F(A) \geq F(B \cup \{j\}) - F(B).$$

The multilinear extension $f : [0, 1]^V \to \mathbb{R}$ of F is defined as:

$$f(x) = \sum_{S \subset V} F(S) \prod_{i \in S} x_i \prod_{j \notin S} (1 - x_j) = \mathbb{E}_{S \sim x}[F(S)].$$

Multilinear extensions are extensively used for maximizing their corresponding submodular set function and are known to be a special case of non-concave DR-submodular functions. The Hessian matrix of this class of functions has non-positive off-diagonal entries and all its diagonal entries are zero. It has been shown that for a large class of submodular set functions, their multilinear extension could be efficiently computed. Weighted matroid rank function, set cover function, probabilistic coverage function, graph cut function and concave over modular function are all examples of such submodular functions (see Iyer et al., 2014; Bian et al., 2019 for more examples and details).
Non-convex/non-concave quadratic functions. Consider the quadratic function $f(x) = \frac{1}{2}x^T Hx + h^T x + c$. If the matrix H is element-wise non-positive, f is a DR-submodular function. We use this class of non-concave DR-submodular functions for the numerical examples.
See Bian et al. (2017a,b) for more examples of continuous DR-submodular objective functions.

3 Problem Statement

The overall offline optimization problem is the following:

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

The online framework is as follows: At step $t \in [T]$, the player chooses $x_t \in \mathcal{X}$. Then, utility function $f_t : \mathcal{X} \to \mathbb{R}$ and p_t are revealed, the player obtains the reward $f_t(x_t)$ and her budget is impacted by the amount $\langle p_t, x_t\rangle$. It is assumed that $\mathcal{X} \subset \mathbb{R}^n_+$ is convex and compact. For all $t \in [T]$, $f_t : \mathcal{X} \to \mathbb{R}$ is a differentiable normalized monotone continuous DR-submodular function and the constraint function $g_t : \mathcal{X} \to \mathbb{R}$, where $g_t(x) = \langle p_t, x\rangle - \frac{B_T}{T}$, is linear and monotone, i.e., $p_t \succeq 0$.
As it was mentioned before, there are existing works on online continuous submodular maximization without online constraints (e.g., Chen et al., 2018a) and online convex optimization with constraints (e.g., Sun et al., 2017). However, our results are *not* a straightforward combination of the mentioned works. First of all, since $p_t \succeq 0 \; \forall t \in [T]$, the Lagrangian $f_t(x) - \lambda_t g_t(x)$ is not monotone non-decreasing and therefore, we cannot simply apply the algorithm of Chen et al. (2018a) to the Lagrangian to obtain the desired results. In this work, we exploit other properties such as the linearity and non-negativity of the constraint functions to obtain the results. Secondly, the techniques and algorithms for online convex optimization are quite different from online continuous DR-submodular maximization and thus, the analysis in Sun et al. (2017) could not be simply adapted to our framework.

3.1 Performance Metric

In order to quantify the performance of our proposed algorithm, we first define our notion of regret and total budget violation below:

Definition 3.1 (Regret Metric) *The $(1 - \frac{1}{e})$-regret* *is defined as:*

$$R_T = (1 - \frac{1}{e}) \sum_{t=1}^{T} f_t(x_W^*) - \sum_{t=1}^{T} f_t(x_t),$$

where:

$$x_W^* = \arg\max_{x \in \mathcal{X}_W} \sum_{t=1}^{T} f_t(x),$$
$$\mathcal{X}_W = \{x \in \mathcal{X} : \sum_{\tau=t}^{t+W-1} g_\tau(x) \leq 0,\ 1 \leq t \leq T - W + 1\}.$$

Note that very recently, Liakopoulos et al. (2019) first introduced the notion of a "K-benchmark", i.e., a comparator which meets the problem's allotted budget over any window of length K, and used this notion for online convex problems with time-varying constraints.

Definition 3.2 (Total Budget Violation) *The total budget violation is defined as follows:*

$$C_T = \sum_{t=1}^{T} g_t(x_t) = \sum_{t=1}^{T} \langle p_t, x_t \rangle - B_T.$$

The regret metric R_T measures the difference between the output of the algorithm and the $(1 - \frac{1}{e})$-approximation to the best fixed decision in hindsight which satisfies the budget constraint proportionally over any window of length W. The $(1 - \frac{1}{e})$-approximation ratio in the definition is the optimal polynomial time approximation ratio for an offline monotone continuous DR-submodular maximization problem (Bian et al., 2017a) and it is commonly used in online submodular maximization literature (e.g., Chen et al., 2018a). If we choose $W = T$ as the window size, we obtain the usual notion of regret where the benchmark action is only required to satisfy the long-term budget constraint and thus, the benchmark action is more aggressive. However, Mannor et al. (2009) provided a simple counterexample showing that the regret of *any algorithm* with window length $W = T$ which has a sub-linear total budget violation is lower bounded by $\Omega(T)$. Hence, inspired by Liakopoulos et al. (2019), we restricted the benchmark action to satisfy the budget constraint proportionally over any window of size W.
We design online algorithms which achieve sub-linear bounds for both the $(1 - \frac{1}{e})$-regret R_T and the total budget violation C_T. We obtain results showing that although it is impossible to obtain sub-linear regret and total budget violation bounds simultaneously for $W = T$, such sub-linear bounds could be guaranteed against a weaker benchmark with window length $W = o(T)$.

3.2 Assumptions

We make the following assumptions:

- $\mathcal{X} \subset \mathbb{R}_+^n$ is a compact and convex set and it contains the origin, i.e., $0 \in \mathcal{X}$.

- The bounded diameter of the compact set $\mathcal{X}$ is R, i.e., we have:

  $$R := \max_{x,y \in \mathcal{X}} \|y - x\|.$$

- Both the utility functions f_t, $\forall t \in [T]$ and constraint functions g_t, $\forall t \in [T]$ are Lipschitz continuous with parameters β_f and β_g respectively and $\beta = \max\{\beta_f, \beta_g\}$. In other words, for all $x, y \in \mathcal{X}$ and $t \in [T]$, we have:

  $$|f_t(y) - f_t(x)| \leq \beta_f \|y - x\|,$$
  $$|g_t(y) - g_t(x)| \leq \beta_g \|y - x\|.$$

  Note that since g_t is linear for all $t \in [T]$, $\beta_g = \max_{t \in [T]} \|p_t\|$ holds.

- For all $t \in [T]$, the utility functions f_t are L-smooth, i.e., for all $x \in \mathcal{X}$ and $u \in \mathbb{R}^n$ where $u \succeq 0$ or $u \preceq 0$, the following holds:

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

Using the above assumptions, we have:

$$F := \max_{t \in [T]} \max_{x,y \in \mathcal{X}} |f_t(x) - f_t(y)| \leq \beta_f R,$$
$$G := \max_{t \in [T]} \max_{x \in \mathcal{X}} |g_t(x)| \leq \beta_g R - \frac{B_T}{T}.$$

4 Online Saddle Point Hybrid Gradient (OSPHG): Algorithm and Analysis

We first introduce our proposed algorithm, the Online Saddle Point Hybrid Gradient (OSPHG) algorithm, in Section 4.1 and then, the analysis for obtaining the regret and total budget violation bounds is provided in Section 4.2.

4.1 Algorithm

Consider the Online Saddle Point Hybrid Gradient (OSPHG) algorithm presented in Algorithm 2.

Algorithm 2 Online Saddle Point Hybrid Gradient (OSPHG) algorithm

Input: $\mathcal{X}$ is the constraint set, T is the horizon, $\mu > 0$, $\delta > 0$ 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 $\mathcal{X}$.
$\lambda_1 = 0$.
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 $\mathcal{E}_k$.
    $x_t^{(k+1)} = x_t^{(k)} + \frac{1}{K} v_t^{(k)}$.
  end for
  Play $x_t = x_t^{(K+1)}$ and observe the Lagrangian function $\mathcal{L}_t(x_t, \lambda_t) = f_t(x_t) - \lambda_t g_t(x_t) + \frac{\delta\mu}{2}\lambda_t^2$.
  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
  $\lambda_{t+1} = [\lambda_t - \mu \nabla_\lambda \mathcal{L}_t(x_t, \lambda_t)]_+$.
end for

First, note that x_t is the convex combination (average) of vectors in the convex set $\mathcal{X}$ and hence, $x_t \in \mathcal{X}$ as well.
The OSPHG algorithm could be interpreted as running two no-regret procedures:

1. K instances $\mathcal{E}_k$ of Online Gradient Ascent where for each $k \in [K]$, at online step $t \in [T]$, the algorithm chooses the point $v_t^{(k)}$ and after committing to this choice, it receives a reward of $\langle v_t^{(k)}, \nabla_x \mathcal{L}_t(x_t^{(k)}, \lambda_t)\rangle$. Note that each instance $\mathcal{E}_k$ $\forall k \in [K]$ corresponds to an online linear maximization problem. The update for $v_{t+1}^{(k)}$ is as follows:
$$v_{t+1}^{(k)} = \mathcal{P}_{\mathcal{X}}\big(v_t^{(k)} + \mu \nabla_x \mathcal{L}_t(x_t^{(k)}, \lambda_t)\big),$$
where $\mathcal{P}_{\mathcal{X}}$ is the projection onto set $\mathcal{X}$. Note that in our applications, the domain set $\mathcal{X}$ is usually a box constraint or the simplex and therefore, projection on $\mathcal{X}$ can be efficiently computed.

2. Online Gradient Descent for the sequence of losses $\{\mathcal{L}_t(x_t, \lambda)\}_{t=1}^T$ where at each online step $t \in [T]$, the algorithm chooses $\lambda_t \geq 0$ and then, observes the loss $-\lambda_t g_t(x_t) + \frac{\delta\mu}{2}\lambda_t^2$. Note that this is an online quadratic minimization problem.

Therefore, the OSPHG algorithm is in fact solving an online saddle point problem at each step and hence the name. It is noteworthy that although we used Online Gradient Descent/Ascent as subroutines in the OSPHG algorithm, *any* other off-the-shelf no-regret online optimization algorithm (such as Online Mirror Descent, Regularized Follow the Leader, etc.) could have been used instead and similar bounds would have been derived. Potential advantages of any such no-regret algorithm over the other could indeed be an interesting research direction.
Our choice of Lagrangian function is inspired by the quadratic penalty method in constrained optimization (Nocedal and Wright, 2006). The penalized formulation of the overall optimization problem (1) with quadratic penalty function could be written as follows:

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

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:

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

Therefore, the corresponding Lagrangian function at round $t \in [T]$ is $\mathcal{L}_t(x, \lambda) = f_t(x) - \lambda(\langle p_t, x\rangle - \frac{B_T}{T}) + \frac{\delta\mu}{2}\lambda^2$.

4.2 Analysis

In order to prove the regret and budget violation bounds, we first provide Lemmas 4.1, 4.2 and 4.3.

Lemma 4.1 *For all $t \in [T]$, the following holds:*

$$\mu \sum_{s=1}^{t} \gamma^{t-s} g_s(x_s) \leq \lambda_{t+1} \leq \mu \sum_{s=1}^{t} \gamma^{t-s} |g_s(x_s)|$$

where $\gamma = 1 - \delta\mu^2$.

Proof See Appendix B for the proof. ■

Using Lemma 4.1 and the inequality $1 - \delta\mu^2 \leq 1$, we can conclude that for all $t \in [T]$, $\lambda_{t+1} \leq \mu t G$ holds. We will use this fact multiple times in the proofs.

Lemma 4.2 *For a fixed $t \in \{1, \ldots, T - W + 1\}$, if δ and μ are chosen such that $\delta\mu^2 \leq \frac{1}{2}$, the following holds:*

$$\sum_{\tau=0}^{W-1} \lambda_{t+\tau} g_{t+\tau}(x_W^*) \leq \lambda_t \sum_{\tau=0}^{W-1} g_{t+\tau}(x_W^*) + G^2\mu W(W-1).$$

Proof See Appendix C for the proof. ■

Lemma 4.3 *For $\mu = \frac{R}{\beta\sqrt{WT}}$, $\delta = 4\beta^2$ and any $\lambda \geq 0$, if T is large enough such that $T \geq \frac{16R^2}{W}$ holds, we have:*

$$R_T + C_T\lambda - \frac{\delta\mu}{2}T\lambda^2 - \frac{\lambda^2}{\mu} \leq (F + \beta R)(W-1) + \frac{R^2}{\mu} + (G^2+\beta^2)\mu T + \frac{G}{2}(G+\beta R)\mu(W-1)(T-1) + G^2\mu(W-1)(T-W+1) + \frac{LR^2}{2K}(T-W+1). \quad (2)$$

Proof See Appendix D for the proof. ■

Lemmas 4.1, 4.2 and 4.3 are all new results. Note that although Sun et al. (2017) use the same Lagrangian function $\mathcal{L}_t$ as our work, they study online *convex* problems (as opposed to our non-convex problem) with window length $W = 1$ and their algorithm and results are not applicable to our framework. Liakopoulos et al. (2019) provide results similar to our lemmas 4.1 and 4.2 for the convex problem, however, their algorithm and choice of Lagrangian function $\mathcal{L}_t(x,\lambda)$, $\forall t \in [T]$ is totally different from ours and therefore, their results are not applicable to our work.
Now, we have all the required tools to prove the performance bounds of the OSPHG algorithm.

Theorem 4.1 (Regret bound) *For $W = o(T)$, if we choose $\mu = \frac{R}{\beta\sqrt{WT}} = \mathcal{O}(\frac{1}{\sqrt{WT}})$, $K = \mathcal{O}(\sqrt{\frac{T}{W}})$ and T is large enough such that $T \geq \frac{16R^2}{W}$ holds, the $(1-\frac{1}{e})$-regret R_T satisfies the following:*

$$R_T \leq \mathcal{O}(\sqrt{WT}).$$

Thus, for $W = T^{1-\epsilon}$, $\forall\epsilon > 0$, the $(1-\frac{1}{e})$-regret of the OSPHG algorithm is $\mathcal{O}(T^{1-\frac{\epsilon}{2}})$ and hence sub-linear.

Proof If we plug in $\lambda = 0$, $\mu = \frac{R}{\beta\sqrt{WT}} = \mathcal{O}(\frac{1}{\sqrt{WT}})$ and $K = \mathcal{O}(\sqrt{\frac{T}{W}})$ in inequality (2), the dominating terms on the right hand side of the inequality are $\frac{G}{2}(G+\beta R)\mu(W-1)(T-1) = \mathcal{O}(\sqrt{WT})$, $\frac{R^2}{\mu} = \mathcal{O}(\sqrt{WT})$, $G^2\mu(W-1)(T-W+1) = \mathcal{O}(\sqrt{WT})$ and $\frac{LR^2}{2K}(T-W+1) = \mathcal{O}(\sqrt{WT})$ and therefore, the result follows. ■

Theorem 4.2 (Budget violation bound) *For $W = o(T)$, if we choose $\mu = \frac{R}{\beta\sqrt{WT}} = \mathcal{O}(\frac{1}{\sqrt{WT}})$, $K = \mathcal{O}(\sqrt{\frac{T}{W}})$ and T is large enough such that $T \geq \frac{16R^2}{W}$ holds, C_T is bounded as follows:*

$$C_T \leq \mathcal{O}(W^{\frac{1}{4}}T^{\frac{3}{4}}).$$

Therefore, for $W = T^{1-\epsilon}$ $\forall\epsilon > 0$, the OSPHG algorithm achieves a sub-linear budget violation bound of $\mathcal{O}(T^{1-\frac{\epsilon}{4}})$.

Proof First, we observe that by assumption, $R_T \geq -FT$ holds where F is defined at the end of Section 3.2. Assume that $C_T \geq 0$ (otherwise, we are done). Setting $\lambda = \frac{C_T}{\delta\mu T + \frac{2}{\mu}}$ in inequality (2), we obtain:

$$\frac{C_T^2}{2\delta\mu T + \frac{4}{\mu}} \leq FT + (F+\beta R)(W-1) + \frac{G}{2}(G+\beta R)\mu(W-1)(T-1) + \frac{R^2}{\mu} + (G^2+\beta^2)\mu T + G^2\mu(W-1)(T-W+1) + \frac{LR^2}{2K}(T-W+1).$$

Plugging in $\mu = \frac{R}{\beta\sqrt{WT}} = \mathcal{O}(\frac{1}{\sqrt{WT}})$ and $K = \mathcal{O}(\sqrt{\frac{T}{W}})$ in the above inequality and multiplying both sides by $2\delta\mu T + \frac{4}{\mu}$, the dominating term on the right hand side of the inequality is $FT(2\delta\mu T + \frac{4}{\mu}) = \mathcal{O}(W^{\frac{1}{2}}T^{\frac{3}{2}})$. Therefore, $C_T^2 \leq \mathcal{O}(W^{\frac{1}{2}}T^{\frac{3}{2}})$ holds. Taking the square root of both sides, we obtain the desired result. ■

Theorems 4.1 and 4.2 provide the first sub-linear regret and total budget violation bounds respectively for the online DR-submodular maximization problem with long-term budget constraints.

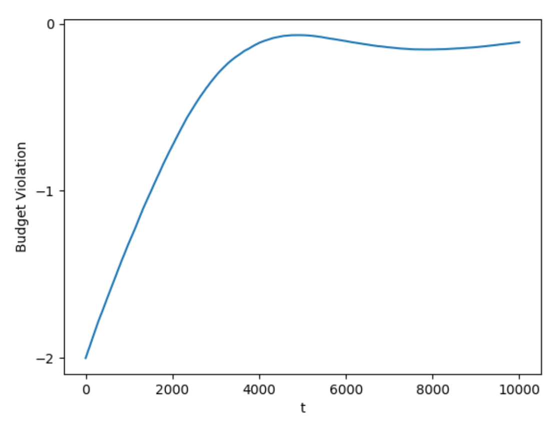

Figure 1: Budget violation running average $\frac{\sum_{\tau=1}^{t} g_\tau(x_\tau)}{t}$ of OSPHG algorithm for $W = \sqrt{T}$.

5 Numerical Examples

We defined $\mathcal{X} = \{x \in \mathbb{R}^n : 0 \preceq x \preceq \mathbf{1}\}$ and for all $t \in [T]$, we randomly generated monotone non-convex/non-concave quadratic utility functions of the form $f_t(x) = \frac{1}{2}x^T H_t x + h_t^T x$ (see section 2.3) where

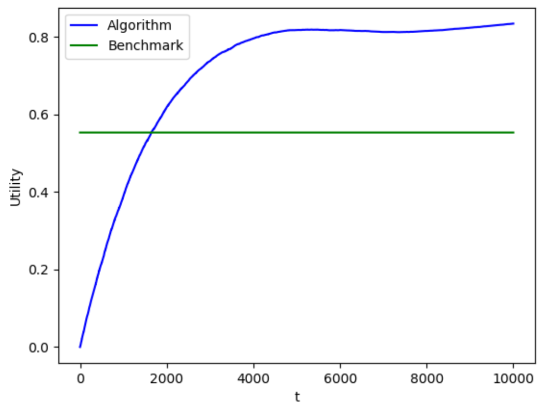

Figure 2: Running average of the utility $\frac{\sum_{\tau=1}^{t} f_\tau(x_\tau)}{t}$ of OSPHG algorithm for $W = \sqrt{T}$ vs. utility of the benchmark $(1-\frac{1}{e})\frac{1}{T}\sum_{t=1}^{T} f_t(x_W^*)$.

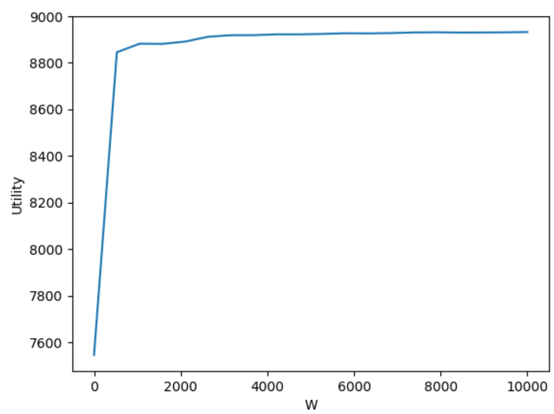

Figure 3: Utility of the benchmark for different window lengths $1 \leq W \leq T$.

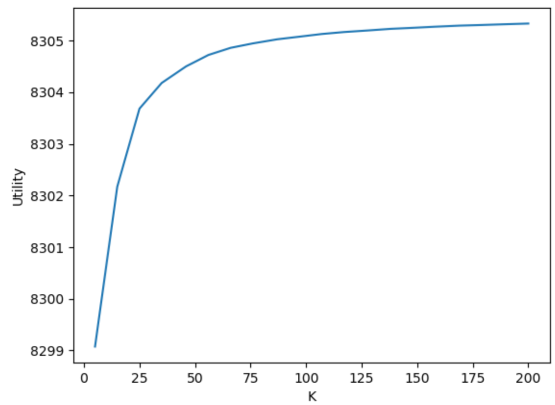

Figure 4: Overall utility of the algorithm $\sum_{t=1}^{T} f_t(x_t)$ for different values of K for $W = \sqrt{T}$.

$H_t \in \mathbb{R}^{n\times n}$ is a random matrix with uniformly distributed non-positive entries in $[-1, 0]$ and $h_t = -H_t^T\mathbf{1}$ to make the gradient non-negative. Therefore, the utility functions are of the form $f_t(x) = (\frac{1}{2}x - \mathbf{1})^T H_t x$. For all $t \in [T]$, we generated random linear budget functions such that p_t has uniformly distributed entries in $[2, 4]$. We set $T = 10000$, $n = 2$, $B_T = 2T$ and $K = 100$. We ran the OSPHG algorithm for $W = \sqrt{T}$. All codes were implemented in Python 3.7. The running average of the budget violation and utility of the OSPHG algorithm is depicted in Figure 1 and Figure 2 respectively which verifies the sub-linearity of the total budget violation and regret of our algorithm (note that the average total budget violation is negative and also, the algorithm achieves higher utilities compared to the benchmark). Additionally, we used the Frank-Wolfe variant algorithm of Bian et al. (2017a) with $K = 100$ for solving offline constrained DR-submodular optimization problems to obtain the utility of the benchmark for different window lengths. As it could be seen in Figure 3, choosing larger window sizes leads to higher utility for the corresponding benchmark and hence, tighter regret guarantees are obtained. However, for a large enough W, there is merely a small difference between the obtained benchmark utility versus the case that $W = T$. Overall obtained utility of the algorithm for different values of K is plotted in Figure 4 and it could be seen that choosing larger values of K leads to a very small increase in the performance which is not significant.

Note that the algorithm of Chen et al. (2018a) does not consider online constraints and the algorithms of Sun et al. (2017); Liakopoulos et al. (2019) are designed for online convex problems and therefore, neither of these algorithms (nor the vanilla online gradient descent) could be simply adapted to our framework to be used as the baseline algorithm for comparison. Our work indeed provides the first algorithm for online DR-submodular maximization with long-term budget constraints.

6 Conclusion

In this paper, we studied a class of online optimization problems with long-term linear budget constraints where the utility functions are monotone continuous DR-submodular. We proposed the Online Saddle Point Hybrid Gradient (OSPHG) algorithm to solve such problems. We considered a refined notion of static regret and proved sub-linear $(1-\frac{1}{e})$-regret and budget violation bounds. Finally, we demonstrated our theoretical findings through a numerical example on a class of continuous DR-submodular functions.

References

Nikolaos Liakopoulos, Apostolos Destounis, Georgios Paschos, Thrasyvoulos Spyropoulos, and Panayotis Mertikopoulos. Cautious regret minimization: Online optimization with long-term budget constraints. In *International Conference on Machine Learning*, pages 3944–3952, 2019.

Mehrdad Mahdavi, Rong Jin, and Tianbao Yang. Trading regret for efficiency: online convex optimization with long term constraints. *Journal of Machine Learning Research*, 13(Sep):2503–2528, 2012.

Rodolphe Jenatton, Jim Huang, and Cedric Archambeau. Adaptive algorithms for online convex optimization with long-term constraints. In *International Conference on Machine Learning*, pages 402–411, 2016.

Jianjun Yuan and Andrew Lamperski. Online convex optimization for cumulative constraints. In *Advances in Neural Information Processing Systems*, pages 6137–6146, 2018.

Shie Mannor, John N Tsitsiklis, and Jia Yuan Yu. Online learning with sample path constraints. *Journal of Machine Learning Research*, 10(Mar):569–590, 2009.

Michael J Neely and Hao Yu. Online convex optimization with time-varying constraints. *arXiv preprint arXiv:1702.04783*, 2017.

Wen Sun, Debadeepta Dey, and Ashish Kapoor. Safety-aware algorithms for adversarial contextual bandit. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3280–3288. JMLR. org, 2017.

Andrew An Bian, Baharan Mirzasoleiman, Joachim Buhmann, and Andreas Krause. Guaranteed non-convex optimization: Submodular maximization over continuous domains. In *Artificial Intelligence and Statistics*, pages 111–120, 2017a.

Daniel Golovin, Andreas Krause, and Matthew Streeter. Online submodular maximization under a matroid constraint with application to learning assignments. *arXiv preprint arXiv:1407.1082*, 2014.

Lin Chen, Hamed Hassani, and Amin Karbasi. Online continuous submodular maximization. In *International Conference on Artificial Intelligence and Statistics*, pages 1896–1905, 2018a.

Lin Chen, Christopher Harshaw, Hamed Hassani, and Amin Karbasi. Projection-free online optimization with stochastic gradient: From convexity to submodularity. In *International Conference on Machine Learning*, pages 813–822, 2018b.

Andreas Krause and Daniel Golovin. Submodular function maximization., 2014.

Jan Vondrák. Optimal approximation for the submodular welfare problem in the value oracle model. In *Proceedings of the fortieth annual ACM symposium on Theory of computing*, pages 67–74. ACM, 2008.

Reza Eghbali and Maryam Fazel. Designing smoothing functions for improved worst-case competitive ratio in online optimization. In *Advances in Neural Information Processing Systems*, pages 3287–3295, 2016.

Gruia Calinescu, Chandra Chekuri, Martin Pál, and Jan Vondrák. Maximizing a submodular set function subject to a matroid constraint. In *International Conference on Integer Programming and Combinatorial Optimization*, pages 182–196. Springer, 2007.

Rishabh Iyer, Stefanie Jegelka, and Jeff Bilmes. Monotone closure of relaxed constraints in submodular optimization: Connections between minimization and maximization: Extended version. In *UAI*, 2014.

Yatao Bian, Joachim Buhmann, and Andreas Krause. Optimal continuous dr-submodular maximization and applications to provable mean field inference. In *International Conference on Machine Learning*, pages 644–653, 2019.

An Bian, Kfir Levy, Andreas Krause, and Joachim M Buhmann. Continuous dr-submodular maximization: Structure and algorithms. In *Advances in Neural Information Processing Systems*, pages 486–496, 2017b.

Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.