Journal of Machine Learning Research 13 (2012) 2903-2922 Submitted 12/11; Revised 7/12; Published 10/12

Online Submodular Minimization

Elad Hazan[*] EHAZAN@IE.TECHNION.AC.IL
Technion - Israel Inst. of Tech.
Technion City
Haifa, 32000, Israel

Satyen Kale[†] SCKALE@US.IBM.COM
IBM T.J. Watson Research Center
P. O. Box 218
Yorktown Heights, NY 10598

Editor: Francis Bach

Abstract

We consider an online decision problem over a discrete space in which the loss function is submodular. We give algorithms which are computationally efficient and are Hannan-consistent in both the full information and partial feedback settings.

Keywords: submodular optimization, online learning, regret minimization

1. Introduction

Online decision-making is a learning problem in which one needs to choose a decision repeatedly from a given set of decisions, in an effort to minimize costs over the long run, even in the face of complete uncertainty about future outcomes. The performance of an online learning algorithm is measured in terms of its *regret*, which is the difference between the total cost of the decisions it chooses, and the cost of the optimal decision chosen in hindsight. A *Hannan-consistent* algorithm is one that achieves sublinear regret (as a function of the number of decision-making rounds). Hannan-consistency implies that the average per round cost of the algorithm converges to that of the optimal decision in hindsight.

In the past few decades, a variety of Hannan-consistent algorithms have been devised for a wide range of decision spaces and cost functions, including well-known settings such as prediction from expert advice (Littlestone and Warmuth, 1989), online convex optimization (Zinkevich, 2003), and more (see the book by Cesa-Bianchi and Lugosi, 2006 for an extensive survey of prediction algorithms). Most of these algorithms are based on an online version of convex optimization algorithms. Despite this success, many online decision-making problems still remain open, especially when the decision space is discrete and large (say, exponential size in the problem parameters) and the cost functions are non-linear.

In this paper, we consider just such a scenario. Our decision space is now the set of all subsets of a ground set of n elements, and the cost functions are assumed to be *submodular*. This property is widely seen as the discrete analogue of convexity, and has proven to be a ubiquitous property in various machine learning tasks (see Guestrin and Krause, 2008 for references). A crucial compo-

∗. Supported by ISF Grant 810/11 and a Google Research Award.
†. Part of this work was done while the author was at Microsoft Research.

nent in these latter results are the celebrated polynomial time algorithms for submodular function minimization (Iwata et al., 2001).

To motivate the online decision-making problem with submodular cost functions, here is an example from the survey by McCormick (2006). Consider a factory capable of producing any subset from a given set of n products E. Let $f : 2^E \mapsto \mathbb{R}$ be the cost function for producing any such subset (here, 2^E stands for the set of all subsets of E). Economics tells us that this cost function should satisfy the law of diminishing returns: that is, the additional cost of producing an additional item is lower the more we produce. Mathematically stated, for all sets $S, T \subseteq E$ such that $T \subseteq S$, and for all elements $i \in E$, we have

$$f(T \cup \{i\}) - f(T) \ \geq \ f(S \cup \{i\}) - f(S).$$

Such cost functions are called *submodular*, and frequently arise in real-world economic and other scenarios. Now, for every item i, let p_i be the market price of the item, which is only determined in the future based on supply and demand. Thus, the profit from producing a subset S of the items is $P(S) = \sum_{i \in S} p_i - f(S)$. Maximizing profit is equivalent to minimizing the function $-P$, which is submodular as well.

The online decision problem which arises is now to decide which set of products to produce, to maximize profits in the long run, without knowing in advance the cost function or the market prices. A more difficult version of this problem, perhaps more realistic, is when the only information obtained is the actual profit of the chosen subset of items, and no information on the profit possible for other subsets.

In general, the Online Submodular Minimization problem is the following. In each iteration, we choose a subset of a ground set of n elements, and then observe a submodular cost function which gives the cost of the subset we chose. The goal is to minimize the regret, which is the difference between the total cost of the subsets we chose, and the cost of the best subset in hindsight. Depending on the feedback obtained, we distinguish between two settings, full-information and bandit. In the full-information setting, we can query each cost function at as many points as we like. In the bandit setting, we only get to observe the cost of the subset we chose, and no other information is revealed.

Obviously, if we ignore the special structure of these problems, standard algorithms for learning with expert advice and/or with bandit feedback can be applied to this setting. However, the computational complexity of these algorithms would be proportional to the number of subsets, which is 2^n. In addition, for the submodular bandits problem, even the regret bounds have an exponential dependence on n. It is hence of interest to design *efficient* algorithms for these problems. For the bandit version an even more basic question arises: does there exist an algorithm with regret which depends only polynomially on n?

In this paper, we answer these questions in the affirmative. We give efficient algorithms for both problems, with regret which is bounded by a polynomial in n, the underlying dimension, and sublinearly in the number of iterations. For the full information setting, we give two different randomized algorithms.

One of these algorithms is based on the follow-the-perturbed-leader approach (Hannan, 1957; Kalai and Vempala, 2005). We give a new way of analyzing such an algorithm. We hope this analysis technique will be applicable to other problems with large decision spaces as well. This algorithm is combinatorial, strongly polynomial, and can be generalized to arbitrary distributive lattices, rather than just all subsets of a given set.

The second algorithm is based on convex analysis. We make crucial use of a continuous extension of a submodular function known as the *Lovász extension*. We obtain our regret bounds by running a (sub)gradient descent algorithm in the style of Zinkevich (2003). The expected regret of this latter algorithm is shown to be bounded by $O(\sqrt{nT})$, and we show this to be optimal.

For the bandit setting, we give a randomized algorithm with expected regret at most $O(nT^{2/3})$. This algorithm also makes use of the Lovász extension and gradient descent. The algorithm folds exploration and exploitation steps into a single sample and obtains the stated regret bound. We also give high-probability bounds on regret of the same order for both settings of online submodular minimization.

An extended abstract of the results of this paper was originally presented in NIPS 2009 (Hazan and Kale, 2009). The present paper contains additional results with detailed proofs and tighter bounds, as well as several corrections.

1.1 Related Work

Submodular optimization has found numerous applications in machine learning and optimization in recent years, see, for example, the survey of Krause and Guestrin (2011). The prediction framework of online convex optimization was put forth by Zinkevich (2003), and found numerous applications since. Flaxman et al. (2005) show how to obtain sub linear regret bounds in the bandit setting. The latter technique is applicable to our setting when applied to the Lovász extension of a submodular function, although this gives weaker regret bounds than the ones presented hereby. Following our work, Jegelka and Bilmes (2011) study constrained submodular minimization over specific combinatorial structures.

2. Preliminaries and Problem Statement

In this section we review the basic concepts of submodular functions, prediction and online convex optimization, and state our main results.

2.1 Submodular Functions

The decision space is the set of all subsets of a universe of n elements, $[n] = \{1,2,\dots,n\}$. The set of all subsets of $[n]$ is denoted $2^{[n]}$. For a set $S \subseteq [n]$, denote by χ_S its characteristic vector in $\{0,1\}^n$, that is, $\chi_S(i) = 1$ if $i \in S$, and 0 otherwise.

A function $f : 2^{[n]} \to \mathbb{R}$ is called *submodular* if for all sets $S,T \subseteq [n]$ such that $T \subseteq S$, and for all elements $i \in [n]$, we have

$$f(T+i) - f(T) \;\geq\; f(S+i) - f(S).$$

Here, we use the shorthand notation $S+i$ to indicate $S \cup \{i\}$. An explicit description of f would take exponential space. We assume therefore that the only way to access f is via a *value oracle*, that is, an oracle that returns the value of f at any given set $S \subseteq [n]$.

Given access to a value oracle for a submodular function, it is possible to minimize it in polynomial time (Grötschel et al., 1988), and indeed, even in strongly polynomial time (Grötschel et al., 1988; Iwata et al., 2001; Schrijver, 2000; Iwata, 2003; Orlin, 2009; Iwata and Orlin, 2009). The current fastest strongly polynomial algorithm is due to Orlin (2009) and takes time $O(n^5\mathrm{EO} + n^6)$, where EO is the time taken to run the value oracle. The fastest weakly polynomial algorithm are given by Iwata (2003) and Iwata and Orlin (2009) and run in time $\tilde{O}(n^4\mathrm{EO} + n^5)$.

2.2 Online Submodular Minimization

In the Online Submodular Minimization problem, over a sequence of iterations $t = 1, 2, \ldots$, an online decision maker has to repeatedly chose a subset $S_t \subseteq [n]$. In each iteration, after choosing the set S_t, the cost of the decision is specified by a submodular function $f_t : 2^{[n]} \to [-M, M]$. The decision maker incurs cost $f_t(S_t)$. The *regret* of the decision maker is defined to be

$$\text{Regret}_T \ := \ \sum_{t=1}^{T} f_t(S_t) - \min_{S \subseteq [n]} \sum_{t=1}^{T} f_t(S).$$

If the sets S_t are chosen by a randomized algorithm, then we consider the expected regret over the randomness in the algorithm.

An online algorithm to choose the sets S_t will be said to be Hannan-consistent if it ensures that $\text{Regret}_T = o(T)$. The algorithm will be called *efficient* if it computes each decision S_t in $\text{poly}(n,t)$ time. Depending on the kind of feedback the decision maker receives, we distinguish between two settings of the problem:

- *Full information setting.* In this case, in each round t, the decision maker has unlimited access to the value oracles of the previously seen cost function $f_1, f_2, \ldots f_{t-1}$.
- *Bandit setting.* In this case, in each round t, the decision maker only observes the cost of her decision S_t, viz. $f_t(S_t)$, and receives no other information.

2.3 Statement of Main Results

In the setup of the Online Submodular Minimization, we have the following results:

Theorem 1 *In the full information setting of Online Submodular Minimization, there is an efficient randomized algorithm that attains the following regret bound:*

$$\mathbf{E}[\textit{Regret}_T] \ = \ O(M\sqrt{nT}).$$

Furthermore, $\textit{Regret}_T = O(M(\sqrt{n} + \sqrt{\log(1/\varepsilon)})\sqrt{T})$ *with probability at least* $1 - \varepsilon$*.*

We also prove a lower bound that shows that the algorithm of Theorem 1 has optimal regret up to constants:

Theorem 2 *In the full information setting of Online Submodular Minimization, for any algorithm, there is a sequence of submodular cost functions such that the algorithm has regret at least* $\Omega(M\sqrt{nT})$.

Theorem 3 *In the bandit setting of Online Submodular Minimization, there is an efficient randomized algorithm that attains the following regret bound:*

$$\mathbf{E}[\textit{Regret}_T] \ = \ O(MnT^{2/3}).$$

Furthermore, $\textit{Regret}_T = O(M(n + \sqrt{n\log(1/\varepsilon)})T^{2/3})$ *with probability at least* $1 - \varepsilon$*.*

Both Theorem 1 and Theorem 3 hold against both oblivious as well as adaptive adversaries, that is, the cost functions can be chosen adversarially with knowledge of the distribution over subsets chosen by the decision maker.

2.4 The Lovász Extension

A major technical construction we need for the algorithms is the *Lovász extension* $\hat{f}$ of the submodular function f. This is defined on the unit hypercube $\mathcal{K} = [0,1]^n$ and takes real values. Before defining the Lovász extension, we need the concept of a chain of subsets of $[n]$:

Definition 4 *A* **chain** *of subsets of $[n]$ is a collection of sets $A_0, A_1, \ldots, A_p$ such that*

$$A_0 \subset A_1 \subset A_2 \subset \cdots \subset A_p.$$

A **maximal chain** *is one where $p = n$. For a maximal chain, we have $A_0 = \emptyset$, $A_n = [n]$, and there is a unique associated permutation $\pi : [n] \to [n]$ such that for all $i \in [n]$, we have $A_{\pi(i)} = A_{\pi(i)-1} + i$. For this permutation π, we have $A_{\pi(i)} = \{j \in [n] : \pi(j) \leq \pi(i)\}$ for all $i \in [n]$.*

Now let $x \in \mathcal{K}$. There is a unique chain $A_0 \subset A_1 \subset \cdots A_p$ such that x can be expressed as a convex combination $x = \sum_{i=0}^{p} \mu_i \chi_{A_i}$ where $\mu_i > 0$ and $\sum_{i=0}^{p} \mu_i = 1$. A nice way to construct this combination is the following random process: choose a threshold $\tau \in [0,1]$ uniformly at random, and consider the level set $S_\tau = \{i : x_i > \tau\}$. The sets in the required chain are exactly the level sets which are obtained with positive probability, and for any such set A_i, $\mu_i = \mathbf{Pr}[S_\tau = A_i]$. In other words, we have $x = \mathbf{E}_\tau[\chi_{S_\tau}]$. This follows immediately by noting that for any i, we have $\mathbf{Pr}_\tau[i \in S_\tau] = x_i$. Of course, the chain and the weights μ_i can also be constructed deterministically simply by sorting the coordinates of x.

Now, we are ready to define[1] the Lovász extension $\hat{f}$:

Definition 5 *Let $x \in \mathcal{K}$. Let $A_0 \subset A_1 \subset \cdots A_p$ such that x can be expressed as a convex combination $x = \sum_{i=0}^{p} \mu_i \chi_{A_i}$ where $\mu_i > 0$ and $\sum_{i=0}^{p} \mu_i = 1$. Then the value of the Lovász extension $\hat{f}$ at x is defined to be*

$$\hat{f}(x) := \sum_{i=0}^{p} \mu_i f(A_i).$$

The preceding discussion gives an equivalent way of defining the Lovász extension: choose a threshold $\tau \in [0,1]$ uniformly at random, and consider the level set $S_\tau = \{i : x_i > \tau\}$. Then we have

$$\hat{f}(x) = \mathbf{E}_\tau[f(S_\tau)].$$

Note that the definition immediately implies that for all sets $S \subseteq [n]$, we have $\hat{f}(\chi_S) = f(S)$.

We will also need the notion of a maximal chain associated to a point $x \in \mathcal{K}$ in order to define subgradients of the Lovász extension:

Definition 6 *Let $x \in \mathcal{K}$, and let $A_0 \subset A_1 \subset \cdots A_p$ be the unique chain such that $x = \sum_{i=0}^{p} \mu_i \chi_{A_i}$ where $\mu_i > 0$ and $\sum_{i=0}^{p} \mu_i = 1$. A* **maximal chain associated with** *x is any maximal completion of the A_i chain, that is, a maximal chain $\emptyset = B_0 \subset B_1 \subset B_2 \subset \cdots B_n = [n]$ such that all sets A_i appear in the B_j chain.*

We have the following key properties of the Lovász extension. For proofs, refer to the book by Fujishige (2005, chapter IV).

Proposition 7 *For a submodular function f, the following properties of its Lovász extension $\hat{f} : \mathcal{K} \to \mathbb{R}$ hold:*

1. Note that this is not the standard definition of the Lovász extension, but an equivalent characterization.

1. *$\hat{f}$ is convex.*

2. *Let $x \in \mathcal{K}$. Let $\emptyset = B_0 \subset B_1 \subset B_2 \subset \cdots B_n = [n]$ be an arbitrary maximal chain associated with x, and let $\pi : [n] \to [n]$ be the corresponding permutation. Then, a subgradient g of $\hat{f}$ at x is given as follows:*

$$g_i \;=\; f(B_{\pi(i)}) - f(B_{\pi(i)-1}).$$

With the notation above, the following Lemma is from the paper by Jegelka and Bilmes (2011):

Lemma 8 (Lemma 1 from Jegelka and Bilmes, 2011) *The subgradients g of the Lovász extension $\hat{f} : \mathcal{K} \to [-M, M]$ of a submodular function are bounded by $\|g\|_2 \leq \|g\|_1 \leq 4M$.*

We provide a proof of this lemma in the appendix for completeness.

3. The Full Information Setting

In this section we give two algorithms for regret minimization in the full information setting. The first is a randomized combinatorial algorithm, based on the "follow the leader" approach of Hannan (1957) and Kalai and Vempala (2005) which attain the regret bound of $O(Mn\sqrt{T})$.

The second is an analytical algorithm based on (sub)gradient descent on the Lovász extension. It attains the regret bound of $O(M\sqrt{nT})$. We also prove a lower bound of $\Omega(M\sqrt{nT})$ on the regret of any algorithm for online submodular minimization, implying that the analytical algorithm is optimal up to constants.

Both algorithms have pros and cons: while the second algorithm is much simpler and more efficient, we do not know how to extend it to distributive lattices, for which the first algorithm readily applies.

3.1 A Combinatorial Algorithm

In this section we analyze a combinatorial, strongly polynomial, algorithm for minimizing regret in the full information Online Submodular Minimization setting:

Algorithm 1 Submodular Follow-The-Perturbed-Leader

1: Input: parameter $\eta > 0$.
2: Initialization: For every $i \in [n]$, choose a random number $r_i \in [-M/\eta, M/\eta]$ uniformly at random. Define $R : 2^{[n]} \to \mathbb{R}$ as $R(S) = \sum_{i \in S} r_i$.
3: **for** $t = 1$ to T **do**
4:    Use the set $S_t = \arg\min_{S \subseteq [n]} \sum_{\tau=1}^{t-1} f_\tau(S) + R(S)$, and obtain cost $f_t(S_t)$.
5: **end for**

Define $\Phi_t : 2^{[n]} \to \mathbb{R}$ as $\Phi_t(S) = \sum_{\tau=1}^{t-1} f_\tau(S) + R(S)$. Note that R is a submodular function, and Φ_t, being the sum of submodular functions, is itself submodular. Furthermore, it is easy to construct a value oracle for Φ_t simply by using the value oracles for the f_τ. Thus, the optimization in step 3 is poly-time solvable given oracle access to Φ_t.

While the algorithm itself is a simple extension of Hannan (1957) follow-the-perturbed-leader algorithm, previous analysis (such as the one given by Kalai and Vempala, 2005), which rely on linearity of the cost functions, cannot be made to work here. Instead, we introduce a new analysis technique: we divide the decision space using n different cuts so that any two decisions are separated by at least one cut, and then we give an upper bound on the probability that the chosen decision switches sides over each such cut. This new technique may have applications to other problems as well. We now prove the regret bound of Theorem 1:

Theorem 9 *Algorithm 1 run with parameter* $\eta = \frac{1}{\sqrt{T}}$ *achieves the following regret bound:*

$$\mathbf{E}[\mathit{Regret}_T] \leq 6Mn\sqrt{T}.$$

Proof We note that the algorithm is essentially running a "follow-the-leader" algorithm on the cost functions $f_0, f_1, \ldots, f_{t-1}$, where $f_0 = R$ is a fictitious "period 0" cost function used for regularization. The first step to analyzing this algorithm is to use a stability lemma, essentially proved by Kalai and Vempala (2005) and reproved in the appendix as Lemma 21 for completeness, which bounds the regret as follows:

$$\text{Regret}_T \leq \sum_{t=1}^{T} [f_t(S_t) - f_t(S_{t+1})] + R(S^*) - R(S_1).$$

Here, $S^* = \arg\min_{S \subseteq [n]} \sum_{t=1}^{T} f_t(S)$.

To bound the expected regret, by linearity of expectation, it suffices to bound $\mathbf{E}[f(S_t) - f(S_{t+1})]$, where for the purpose of analysis, we assume that we re-randomize in every round (that is, choose a fresh random function $R : 2^{[n]} \to \mathbb{R}$). Naturally, the expectation $\mathbf{E}[f(S_t) - f(S_{t+1})]$ is the same regardless of when R is chosen.

To bound this, we need the following lemma:

Lemma 10

$$\mathbf{Pr}[S_t \neq S_{t+1}] \leq 2n\eta.$$

Proof First, we note the following simple union bound:

$$\mathbf{Pr}[S_t \neq S_{t+1}] \leq \sum_{i \in [n]} \mathbf{Pr}[i \in S_t \text{ and } i \notin S_{t+1}] + \mathbf{Pr}[i \notin S_t \text{ and } i \in S_{t+1}]. \tag{1}$$

Now, fix any i, and we aim to bound $\mathbf{Pr}[i \in S_t \text{ and } i \notin S_{t+1}]$. For this, we condition on the randomness in choosing r_j for all $j \neq i$. Define $R' : 2^{[n]} \to \mathbb{R}$ as $R'(S) = \sum_{j \in S, j \neq i} r_j$, and $\Phi'_t : 2^{[n]} \to \mathbb{R}$ as $\Phi'_t(S) = \sum_{\tau=1}^{t-1} f_\tau(S) + R'(S)$. Note that if $i \notin S$, then $R'(S) = R(S)$ and $\Phi'_t(S) = \Phi_t(S)$. Let

$$A = \arg\min_{S \subseteq [n] : i \in S} \Phi'(S) \quad \text{and} \quad B = \arg\min_{S \subseteq [n] : i \notin S} \Phi'(S).$$

Now, we note that the event $i \in S_t$ happens only if $\Phi'_t(A) + r_i < \Phi'_t(B)$, and $S_t = A$. But if $\Phi'_t(A) + r_i < \Phi'_t(B) - 2M$, then we must have $i \in S_{t+1}$, since for any C such that $i \notin C$,

$$\Phi_{t+1}(A) = \Phi'_t(A) + r_i + f_t(A) < \Phi'_t(B) - M < \Phi'_t(C) + f_t(C) = \Phi_t(C).$$

The inequalities above use the fact that $f_t(S) \in [-M, M]$ for all $S \subseteq [n]$. Thus, if $v := \Phi'_t(B) - \Phi'_t(A)$, we have

$$\mathbf{Pr}[i \in S_t \text{ and } i \notin S_{t+1} \mid r_j, j \neq i] \leq \mathbf{Pr}[r_i \in [v - 2M, v] \mid r_j, j \neq i] \leq \eta,$$

since r_i is chosen uniformly from $[-M/\eta, M/\eta]$. We can now remove the conditioning on r_j for $j \neq i$, and conclude that

$$\mathbf{Pr}[i \in S_t \text{ and } i \notin S_{t+1}] \leq \eta.$$

Similarly, we can bound $\mathbf{Pr}[i \notin S_t \text{ and } i \in S_{t+1}] \le \eta$. Finally, the union bound (1) over all choices of i yields the required bound on $\mathbf{Pr}[S_t \neq S_{t+1}]$. ■

Continuing the proof, we have (since $|f(S)| \le M$)

$$\begin{aligned}\mathbf{E}[f(S_t) - f(S_{t+1})] &= \mathbf{E}[f(S_t) - f(S_{t+1}) \mid S_t \neq S_{t+1}] \cdot \mathbf{Pr}[S_t \neq S_{t+1}] \\ &\le 2M \cdot \mathbf{Pr}[S_t \neq S_{t+1}] \\ &\le 4Mn\eta.\end{aligned}$$

The last inequality follows from Lemma 10. Now, we have $R(S^*) - R(S_1) \le 2Mn/\eta$, and so

$$\begin{aligned}\mathbf{E}[\text{Regret}_T] &\le \sum_{t=1}^{T} \mathbf{E}[f(S_t) - f(S_{t+1})] + \mathbf{E}[R(S^*) - R(S_1)] \\ &\le 4Mn\eta T + \frac{2Mn}{\eta} \\ &\le 6Mn\sqrt{T},\end{aligned}$$

since $\eta = \frac{1}{\sqrt{T}}$. ■

3.2 An Analytical Algorithm

In this section, we give a different algorithm based on the Online Gradient Descent method of Zinkevich (2003). We apply this technique to the Lovász extension of the cost function coupled with a simple randomized construction of the subgradient, as given in definition 5. This algorithm requires the concept of a *Euclidean projection* of a point in $\mathbb{R}^n$ on to the set $\mathcal{K}$, which is a function $\Pi_{\mathcal{K}} : \mathbb{R}^n \to \mathcal{K}$ defined by

$$\Pi_{\mathcal{K}}(y) := \arg\min_{x \in \mathcal{K}} \|x - y\|.$$

Since $\mathcal{K} = [0,1]^n$, it is easy to implement this projection: indeed, for a point $y \in \mathbb{R}^n$, the projection $x = \Pi_{\mathcal{K}}(y)$ is defined by

$$x_i = \begin{cases} y_i & \text{if } y_i \in [0,1] \\ 0 & \text{if } y_i < 0 \\ 1 & \text{if } y_i > 1. \end{cases}$$

Algorithm 2 Submodular Subgradient Descent

Input: parameter $\eta > 0$. Let $x_1 \in \mathcal{K}$ be an arbitrary initial point.
for $t = 1$ to T **do**
  Choose a threshold $\tau \in [0,1]$ uniformly at random, and use the set $S_t = \{i : x_t(i) > \tau\}$ and obtain cost $f_t(S_t)$.
  Find a maximal chain associated with x_t, $\emptyset = B_0 \subset B_1 \subset B_2 \subset \cdots B_n = [n]$, and use $f_t(B_0), f_t(B_1), \ldots, f_t(B_n)$ to compute a subgradient g_t of $\hat{f}_t$ at x_t as in part 2 of Proposition 7.
  Update: set $x_{t+1} = \Pi_{\mathcal{K}}(x_t - \eta g_t)$.
end for

In the analysis of the algorithm, we need the following regret bound. It is a simple extension of Zinkevich's analysis of Online Gradient Descent to vector-valued random variables whose expectation is the subgradient of the cost function (the generality to random variables is not required for this section, but it will be useful in the next section):

Lemma 11 *Let $\hat{f}_1, \hat{f}_2, \ldots, \hat{f}_T : \mathcal{K} \to \mathbb{R}$ be a sequence of convex cost functions over the cube $\mathcal{K}$. Let $x_1, x_2, \ldots, x_T \in \mathcal{K}$ be defined by $x_1 = 0$ and $x_{t+1} = \Pi_{\mathcal{K}}(x_t - \eta \hat{g}_t)$, where $\hat{g}_1, \hat{g}_2, \ldots, \hat{g}_T$ are vector-valued random variables such that $\mathbf{E}[\hat{g}_t | x_t] = g_t$, where g_t is a subgradient of $\hat{f}_t$ at x_t. Then the expected regret of playing $x_1, x_2, \ldots, x_T$ is bounded by*

$$\sum_{t=1}^{T} \mathbf{E}[\hat{f}_t(x_t)] - \min_{x \in \mathcal{K}} \sum_{t=1}^{T} \hat{f}_T(x) \ \le\ \frac{n}{2\eta} + 2\eta \sum_t \mathbf{E}[\|\hat{g}_t\|^2].$$

Proof Let $y_{t+1} = x_t - \eta \hat{g}_t$, so that $x_{t+1} = \Pi_{\mathcal{K}}(y_{t+1})$. Note that

$$\|y_{t+1} - x^*\|^2 \ =\ \|x_t - x^*\|^2 - 2\eta \hat{g}_t^\top (x_t - x^*) + \eta^2 \|\hat{g}_t\|^2.$$

Rearranging,

$$\begin{aligned} \hat{g}_t^\top (x_t - x^*) &= \frac{1}{2\eta}[\|x_t - x^*\|^2 - \|y_{t+1} - x^*\|^2] + \frac{\eta}{2}\|\hat{g}_t\|^2 \\ &\le \frac{1}{2\eta}[\|x_t - x^*\|^2 - \|x_{t+1} - x^*\|^2] + \frac{\eta}{2}\|\hat{g}_t\|^2, \end{aligned}$$

since $\|x_{t+1} - x^*\| \le \|y_{t+1} - x^*\|$ by the properties of Euclidean projections onto convex sets. Hence, we have

$$\begin{aligned} \sum_{t=1}^{T} \hat{g}_t^\top (x_t - x^*) &\le \sum_{t=1}^{T} \frac{\|x_t - x^*\|^2 - \|x_{t+1} - x^*\|^2}{2\eta} + \frac{\eta}{2}\|\hat{g}_t\|^2 \\ &\le \frac{n}{2\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \|\hat{g}_t\|^2, \end{aligned}$$

since $\|x_1 - x^*\|^2 \le n$, both x_1 and x^* being in the cube $\mathcal{K}$. Next, since $\mathbf{E}[\hat{g}_t | x_t] = g_t$, a subgradient of $\hat{f}_t$ at x_t, we have

$$\mathbf{E}[\hat{g}_t^\top (x_t - x^*) | x_t] \ =\ g_t^\top (x_t - x^*) \ \ge\ \hat{f}_t(x_t) - \hat{f}_t(x^*),$$

since $\hat{f}_t$ is a convex function. Taking expectation over the choice of x_t, we have

$$\mathbf{E}[\hat{g}_t^\top (x_t - x^*)] \ \ge\ \mathbf{E}[\hat{f}_t(x_t)] - \hat{f}_t(x^*).$$

Thus, we can bound the expected regret as follows:

$$\sum_{t=1}^{T} \mathbf{E}[\hat{f}_t(x_t)] - \hat{f}_t(x^*) \ \le\ \mathbf{E}\left[\sum_{t=1}^{T} \hat{g}_t^\top (x_t - x^*)\right] \ \le\ \frac{n}{2\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \mathbf{E}[\|\hat{g}_t\|^2].$$

■

We can now prove the following regret bound:

Theorem 12 *Algorithm 2, run with parameter* $\eta = \sqrt{\frac{n}{16MT}}$*, achieves the following regret bound:*

$$\mathbf{E}[Regret_T] \ \leq \ 4M\sqrt{nT}.$$

Furthermore, with probability at least $1-\varepsilon$*, we have*

$$Regret_T \ \leq \ 4M\sqrt{nT} + M\sqrt{2T\log(1/\varepsilon)}.$$

Proof Note that be Definition 5, we have that $\mathbf{E}[f_t(S_t)] = \hat{f}_t(x_t)$. Since the algorithm runs Online Gradient Descent (from Lemma 11) with $\hat{g}_t = g_t$ (that is, no randomness), we get the following bound on the regret. Here, we use the bound of Lemma 8 $\|\hat{g}_t\|^2 = \|g_t\|^2 \leq 16M^2$.

$$\begin{aligned}
\mathbf{E}[\mathrm{Regret}_T] \ &= \ \sum_{t=1}^{T} \mathbf{E}[f_t(S_t)] - \min_{S \subseteq [n]} \sum_{t=1}^{T} f(S) \\
&\leq \ \sum_{t=1}^{T} \hat{f}_t(x_t) - \min_{x \in \mathcal{K}} \sum_{t=1}^{T} \hat{f}_T(x) \\
&\leq \ \frac{n}{2\eta} + \frac{16}{2}\eta M^2 T \\
&\leq \ 4M\sqrt{nT},
\end{aligned}$$

where the last inequality is due to the choice of η as in the theorem statement.

We proceed to give a high probability bound. The following Theorem is by Hoeffding (see the book by Cesa-Bianchi and Lugosi 2006, Appendix A):

Theorem 13 (Hoeffding) *Let* $X_1, \ldots, X_T$ *be independent random variables such that* $|X_t| \leq M$*. Then, for* $\varepsilon > 0$*, we have*

$$\Pr\left[\sum_{t=1}^{T} X_t - \mathbf{E}\left[\sum_{t=1}^{T} X_t\right] > M\sqrt{2T\log(1/\varepsilon)}\right] \ \leq \ \varepsilon.$$

Note that the sequence of points $x_1, x_2, \ldots, x_T$ is deterministic since it is obtained by deterministic gradient descent. The sets S_t are obtained by independent randomized rounding on the x_t's, and so the random variables $X_t = f_t(S_t)$ are independent. Note that $|X_t| \leq M$. Applying the Hoeffding bound above we get that with probability at least $1-\varepsilon$,

$$\sum_{t=1}^{T} f_t(S_t) \ \leq \ \sum_{t=1}^{T} \mathbf{E}[f_t(S_t)] + M\sqrt{2T\log(1/\varepsilon)},$$

which implies the high probability regret bound. ■

3.3 Lower Bound on Regret

We give a simple lower bound (which is reminiscent of the lower bounds for the setting of prediction from expert advice as in the book by Cesa-Bianchi and Lugosi, 2006), that in the full-information setting any algorithm for online submodular minimization can be made to have regret $\Omega(M\sqrt{nT})$. This show that the upper bound of Theorem 12 is optimal up to constants.

Theorem 14 *In the full-information setting, for any algorithm for online submodular minimization, there is a sequence of submodular cost functions* $f_1, f_2, \ldots, f_T : 2^{[n]} \to [-M,M]$ *such that the regret of the algorithm is at least* $\Omega(M\sqrt{nT})$*.*

Proof Consider the following randomized sequence of cost functions. In round t, choose the element $i(t) = (t \bmod n) + 1 \in [n]$, and a Rademacher random variable $\sigma_t \in \{-1,1\}$ chosen independently of all other random variables. Then, define $f_t : 2^{[n]} \to [-M,M]$ as:

$$\forall S \subseteq [n]: \; f_t(S) \;=\; \begin{cases} -\sigma_t M & \text{if } i(t) \notin S \\ \sigma_t M & \text{if } i(t) \in S. \end{cases}$$

It is easy to check that f_t is submodular (in fact, it is modular). Note that for any set S_t played by the algorithm in round t, we have $\mathbf{E}[f_t(S_t)] = 0$, where the expectation is taken over the choice of σ_t. Thus, in expectation, the cost of the algorithm is 0. But now consider the set $S \subseteq [n]$ defined as follows. For all $i \in [n]$, let $X_i = \sum_{t:i(t)=i} \sigma_t$. Then let $S = \{i: \; X_i \le 0\}$. Observe that by construction,

$$\sum_t f_t(S) \;=\; \sum_i -M|X_i|,$$

and hence

$$\mathbf{E}\left[\sum_t f_t(S)\right] = \mathbf{E}\left[\sum_i -M|X_i|\right] = n \cdot -M \cdot \Omega\left(\sqrt{\frac{T}{n}}\right) = -\Omega(M\sqrt{nT}).$$

Here, we used the fact that each X_i is a sum of at least $\lfloor \frac{T}{n} \rfloor$ independent Rademacher random variables, and Khintchine's inequality (see Cesa-Bianchi and Lugosi, 2006, Appendix A) implies that if Y is a sum of m independent Rademacher random variables, then $\mathbf{E}[|Y|] \ge \sqrt{m/2}$. Hence, the expected regret of the algorithm is $\Omega(M\sqrt{nT})$. In particular, there is a specific choice of the Rademacher random variables σ_t such that the algorithm incurs regret at least $\Omega(M\sqrt{nT})$. ■

4. The Bandit Setting

We now present an algorithm for the Bandit Online Submodular Minimization problem. The algorithm is based on the Online Gradient Descent algorithm of Zinkevich (2003). The main idea is to use just one sample for both exploration (to construct an unbiased estimator for the subgradient) and exploitation (to construct an unbiased estimator for the point chosen by the Online Gradient Descent algorithm).

Algorithm 3 Bandit Submodular Subgradient Descent

Input: parameters $\eta, \delta > 0$. Let $x_1 \in \mathcal{K}$ be arbitrary.
for $t = 1$ to T **do**
  Find a maximal chain associated with x_t, $\emptyset = B_0 \subset B_1 \subset B_2 \subset \cdots B_n = [n]$, and let π be the associated permutation as in part 2 of Proposition 7. Then x_t can be written as $x_t = \sum_{i=0}^{n} \mu_i \chi_{B_i}$, where $\mu_i = 0$ for the extra sets B_i that were added to complete the maximal chain for x_t.
  Choose the set S_t as follows:

$$S_t = B_i \quad \text{with probability} \quad \rho_i = (1-\delta)\mu_i + \frac{\delta}{n+1}.$$

  Use the set S_t and obtain cost $f_t(S_t)$.
  If $S_t = B_0$, then set $\hat{g}_t = -\frac{1}{\rho_0} f_t(S_t) e_{\pi(1)}$, and if $S_t = B_n$ then set $\hat{g}_t = \frac{1}{\rho_n} f_t(S_t) e_{\pi(n)}$. Otherwise, $S_t = B_i$ for some $1 \le i \le n-1$. Choose $\varepsilon_t \in \{+1, -1\}$ uniformly at random, and set:

$$\hat{g}_t = \begin{cases} \frac{2}{\rho_i} f_t(S_t) e_{\pi(i)} & \text{if } \varepsilon_t = 1 \\ -\frac{2}{\rho_i} f_t(S_t) e_{\pi(i+1)} & \text{if } \varepsilon_t = -1. \end{cases}$$

  Update: set $x_{t+1} = \Pi_{\mathcal{K}}(x_t - \eta \hat{g}_t)$.
end for

Before launching into the analysis, we define some convenient notation first. Define the filtration $\mathcal{F} = (\mathcal{F}_{t \le T})$, where $\mathcal{F}_t$ is the smallest σ-field with respect to which the random coin tosses of the algorithm in rounds $1, 2, \ldots, t$ are measurable, and let $\mathbf{E}_t[\cdot] = \mathbf{E}[\cdot | \mathcal{F}_{t-1}]$, and $\mathrm{VAR}_t[\cdot] = \mathrm{VAR}[\cdot | \mathcal{F}_{t-1}]$.

A first observation is that in expectation, the regret of the algorithm above is almost the same as if it had played x_t all along and the loss functions were replaced by the Lovász extensions of the actual loss functions.

Lemma 15 *For all t, we have $\mathbf{E}[f_t(S_t)] \le \mathbf{E}[\hat{f}_t(x_t)] + 2\delta M$.*

Proof From Definition 5 we have that $\hat{f}_t(x_t) = \sum_i \mu_i f_t(B_i)$. On the other hand, $\mathbf{E}_t[f_t(S_t)] = \sum_i \rho_i f_t(B_i)$, and hence:

$$\mathbf{E}_t[f_t(S_t)] - \hat{f}_t(x_t) \;=\; \sum_{i=0}^{n} (\rho_i - \mu_i) f_t(B_i) \;\le\; \delta \sum_{i=0}^{n} \left[\frac{1}{n+1} + \mu_i\right] |f_t(B_i)| \;\le\; 2\delta M. \tag{2}$$

The lemma now follows by taking expectations on both sides with respect to the randomness up to round $t-1$. ■

Next, by Proposition 7, the subgradient of the Lovász extension of f_t at point x_t corresponding to the maximal chain $B_0 \subset B_1 \subset \cdots \subset B_n$ is given by $g_t(i) = f(B_{\pi(i)}) - f(B_{\pi(i)-1})$. Using this fact, it is easy to check that the random vector $\hat{g}_t$ is constructed in such a way that $\mathbf{E}[\hat{g}_t | x_t] = \mathbf{E}_t[\hat{g}_t] = g_t$. Furthermore, we can bound the norm of this estimator as follows:

$$\mathbf{E}_t[\|\hat{g}_t\|^2] \;\le\; \sum_{i=0}^{n} \frac{4}{\rho_i^2} f_t(B_i)^2 \cdot \rho_i \;\le\; \frac{4M^2(n+1)^2}{\delta} \;\le\; \frac{16M^2 n^2}{\delta}. \tag{3}$$

We can now remove the conditioning, and conclude that $\mathbf{E}[\|\hat{g}_t\|^2] \leq \frac{16M^2n^2}{\delta}$.

Theorem 16 *Algorithm 3, run with parameters* $\delta = \frac{n}{T^{1/3}}$, $\eta = \frac{1}{4MT^{2/3}}$, *achieves the following regret bound:*

$$\mathbf{E}[Regret_T] \ \leq \ 6MnT^{2/3}.$$

Proof We bound the expected regret as follows: using Lemma 15), we have

$$\begin{aligned}
\sum_{t=1}^{T} \mathbf{E}[f_t(S_t)] - \min_{S\subseteq[n]} \sum_{t=1}^{T} f_t(S) \ &\leq \ 2\delta MT + \sum_{t=1}^{T} \mathbf{E}[\hat{f}_t(x_t)] - \min_{x\in\mathcal{K}} \sum_{t=1}^{T} \hat{f}_t(x) \\
&\leq \ 2\delta MT + \frac{n}{2\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \mathbf{E}[\|\hat{g}_t\|^2] && \text{(By Lemma 11)} \\
&\leq \ 2\delta MT + \frac{n}{2\eta} + \frac{8n^2M^2\eta T}{\delta}. && \text{(By (3))}
\end{aligned}$$

The bound is now obtained using the stated values for η, δ. ■

4.1 High Probability Bounds on the Regret

The theorem of the previous section gave a bound on the expected regret. However, a much stronger claim can be made that essentially the same regret bound holds with very high probability (exponential tail). The following gives high probability bounds against an adaptive adversary.

Theorem 17 *With probability at least* $1-4\varepsilon$, *Algorithm 3, run with parameters* $\delta = \frac{n}{T^{1/3}}$, $\eta = \frac{1}{4MT^{2/3}}$, *achieves the following regret bound:*

$$Regret_T \ \leq \ 38MnT^{2/3} + 44M\sqrt{n}T^{2/3}\sqrt{\log(1/\varepsilon)}.$$

To prove the high probability regret bound, we require the following concentration lemma which can be found in the book by Cesa-Bianchi and Lugosi (2006, Appendix A):

Lemma 18 (Bernstein inequality for martingales) *Let* $X_1, \ldots, X_T$ *be a sequence of bounded random variables adapted to a filtration* $\mathcal{F} = (\mathcal{F}_t)_{t\leq T}$. *Let* $\mathbf{E}_t[\cdot] := \mathbf{E}[\cdot|\mathcal{F}_{t-1}]$. *Suppose that* $|X_t| \leq b$ *and let* $\mathbf{E}_t[X_t^2] \leq V$ *for all* $t \leq T$. *Then, for* $\varepsilon > 0$, *we have*

$$\Pr\left[\left|\sum_{t=1}^{T} X_t - \mathbf{E}_t[X_t]\right| > \sqrt{2TV\log(1/\varepsilon)} + b\log(1/\varepsilon)\right] \ \leq \ \varepsilon.$$

The following simple corollary will be useful in the analysis:

Corollary 19 *In the setup of Lemma 18, assume that the parameters* T, V, b *and* ε *satisfy* $\sqrt{TV} > b\sqrt{\log(1/\varepsilon)}$. *Then*

$$\Pr\left[\left|\sum_{t=1}^{T} X_t - \mathbf{E}_t[X_t]\right| > 4\sqrt{TV\log(1/\varepsilon)}\right] \ \leq \ \varepsilon.$$

Proof [Theorem 17] If $T \leq 2\log^{3/2}(1/\varepsilon)$, then the regret can be trivially bounded by

$$2MT \;\leq\; 4MT^{2/3}\sqrt{\log(1/\varepsilon)} \leq\; 38MnT^{2/3} + 44M\sqrt{n}T^{2/3}\sqrt{\log(1/\varepsilon)}.$$

So from now on we assume that $T > 2\log^{3/2}(1/\varepsilon)$. We need the following lemma:

Lemma 20 *If $T > 2\log^{3/2}(1/\varepsilon)$, then with probability at least $1-4\varepsilon$, all of the following inequalities hold:*

$$\sum_{t=1}^{T} \|\hat{g}_t\|^2 \;\leq\; \sum_{t=1}^{T} \mathbf{E}_t[\|\hat{g}_t\|^2] + 64M^2T^{4/3}\sqrt{\log(1/\varepsilon)}, \tag{4}$$

$$\sum_{t=1}^{T} g_t^\top x_t \;\leq\; \sum_{t=1}^{T} \hat{g}_t^\top x_t + 16M\sqrt{n}T^{2/3}\sqrt{\log(1/\varepsilon)}, \tag{5}$$

$$\forall S \subseteq [n], \quad \sum_{t=1}^{T} \hat{g}_t^\top \chi_S \;\leq\; \sum_{t=1}^{T} g_t^\top \chi_S + 16M\sqrt{n}T^{2/3}\sqrt{\log(2^n/\varepsilon)}, \tag{6}$$

$$\textit{and} \;\; \sum_{t=1}^{T} f_t(S_t) \;\leq\; \sum_{t=1}^{T} \mathbf{E}_t[f_t(S_t)] + 4M\sqrt{T\log(1/\varepsilon)}. \tag{7}$$

Proof We use Lemma 18 to bound the probability of each of the four events *not* happening by ε, and then we apply a union bound. In the following, we use the lower bound

$$\rho_i \;\geq\; \frac{\delta}{n+1} \;\geq\; \frac{1}{2T^{1/3}}. \tag{8}$$

Recall the filtration $\mathcal{F} = (\mathcal{F}_{t\leq T})$, where $\mathcal{F}_t$ is the smallest σ-field with respect to which the random coin tosses of the algorithm in rounds $1,2,\ldots,t$ are measurable. In the following we will consider sequences of random variables $X_1, X_2, \ldots, X_T$ adapted to $\mathcal{F}$.

Proof of (4). Consider the random variables $X_t := \|\hat{g}_t\|^2$ for $t \leq T$, that are adapted to $\mathcal{F}$. To apply Corollary 19, we estimate the parameters b, V. If B_i was sampled in step t, we have, using (3),

$$|X_t| \;=\; \|\hat{g}_t\|^2 \;\leq\; \frac{4}{\rho_i^2} f_t^2(B_i) \;\leq\; 16M^2T^{2/3},$$

using (8). Thus, we can choose $b = 16M^2T^{2/3}$. Next, we have

$$\mathbf{E}[\|\hat{g}_t\|^4|\mathcal{F}_{t-1}] \;\leq\; \sum_{i=0}^{n} \frac{16}{\rho_i^4} f_t(B_i)^4 \cdot \rho_i \;\leq\; (n+1)\cdot 16 \cdot 8T \cdot M^4 \;\leq\; 256M^4nT.$$

Thus, we can choose $V = 256M^4nT$. Now, $\sqrt{TV} = 16M^2\sqrt{n}T > b\sqrt{\log(1/\varepsilon)}$ for $T > \log^{3/2}(1/\varepsilon)$. The required bound follows from Corollary 19 using the overestimation $\sqrt{n} \leq T^{1/3}$.

Proof of (5). Consider the random variables $X_t := \hat{g}_t^\top x_t$ for $t \leq T$, that are adapted to $\mathcal{F}$. First note that $\mathbf{E}_t[\hat{g}_t^\top x_t] = g_t^\top x_t$. To apply Corollary 19, we estimate the parameters b, V. First, if B_i was sampled in step t, then we have, using Hölder's inequality,

$$|X_t| \;\leq\; \|\hat{g}_t\|_1\|x_t\|_\infty \;\leq\; \frac{2}{\rho_i}|f_t(B_i)| \;\leq\; 4MT^{1/3},$$

using (8). Thus, we choose $b = 4MT^{1/3}$. Next, again using Hölder's inequality we have

$$\mathbf{E}_t[(\hat{g}_t^\top x_t)^2] \leq \mathbf{E}_t[\|\hat{g}_t\|_1^2 \|x_t\|_\infty^2] \leq \sum_{i=0}^{n} \frac{4}{\rho_i^2} f_t^2(B_i) \cdot \rho_i \leq 16M^2 nT^{1/3},$$

using (8). Thus, we can choose $V = 16M^2 nT^{1/3}$. Note that $\sqrt{TV} = 4M\sqrt{n}T^{2/3} > b\sqrt{\log(1/\varepsilon))}$ for $T > \log^{3/2}(1/\varepsilon)$. The required bound follows from Corollary 19.

Proof of (6). This bound follows exactly as the previous one, except we use the random variables $X_t := \hat{g}_t^\top \chi_S$ for every fixed set $S \subseteq [n]$, with error parameter $\varepsilon/2^n$. With this value of the error parameter, the conditions of Corollary 19 are met for $T > 2\log^{3/2}(1/\varepsilon)$. We then take a union bound over the 2^n choices of S to obtain the required bound.

Proof of (7). Consider the random variables $X_t := f_t(S_t)$ for $t \leq T$, that are adapted to $\mathcal{F}$. To apply Corollary 19, we estimate the parameters b, V. We have

$$|X_t| = |f_t(S_t)| \leq M.$$

So we can use $b = M$. As for V, we use the trivial bound $V = b^2 = M^2$. Again, $\sqrt{TV} = \sqrt{T}b > b\sqrt{\log(1/\varepsilon)}$ for $T > \log(1/\varepsilon)$. The required bound follows from Corollary 19. ■

Finally, we can imagine the points $x_1, x_2, \ldots, x_T$ as being produced by running Online Gradient Descent with linear cost functions $\hat{g}_t^\top x$, thinking of $\hat{g}_t$ as deterministic vectors. Thus, by Lemma 11, we get that for any $S \subseteq [n]$, we have

$$\sum_{t=1}^{T} \hat{g}_t^\top (x_t - \chi_S) \leq \frac{n}{2\eta} + \frac{\eta}{2} \sum_{t=1}^{T} \|\hat{g}_t\|^2. \tag{9}$$

Thus, with probability $1-4\varepsilon$, for any $S \subseteq [n]$, we have

$$\begin{aligned}
&\sum_{t=1}^{T} f_t(S_t) - f_t(S) \\
&\le \sum_{t=1}^{T} \mathbf{E}_t[f_t(S_t)] - f_t(S) + 4M\sqrt{T\log(1/\varepsilon)} && \text{(By (7))} \\
&\le \sum_{t=1}^{T} \hat{f}_t(x_t) - \hat{f}_t(\chi_S) + 2nMT^{2/3} + 4M\sqrt{T\log(1/\varepsilon)} && \text{(By (2))} \\
&\le \sum_{t=1}^{T} g_t^{\top}(x_t - \chi_S) + 2MnT^{2/3} + 4M\sqrt{T\log(1/\varepsilon)} && \text{(by convexity of } \hat{f}_t) \\
&\le \sum_{t=1}^{T} \hat{g}_t^{\top}(x_t - \chi_S) + 2MnT^{2/3} + 4M\sqrt{T\log(1/\varepsilon)} \\
&\qquad + 32M\sqrt{n}T^{2/3}\sqrt{\log(2^n/\varepsilon)} && \text{(By (5), (6))} \\
&\le \sum_{t=1}^{T} \hat{g}_t^{\top}(x_t - \chi_S) + 34MnT^{2/3} + 36M\sqrt{n}T^{2/3}\sqrt{\log(1/\varepsilon)} \\
&\le \frac{n}{\eta} + \frac{\eta}{2}\sum_{t=1}^{T} \|\hat{g}_t\|^2 + 34MnT^{2/3} + 36M\sqrt{n}T^{2/3}\sqrt{\log(1/\varepsilon)} && \text{(By (9))} \\
&\le \frac{n}{2\eta} + \frac{\eta}{2}\left[\sum_{t=1}^{T} \mathbf{E}_t[\|\hat{g}_t\|^2] + 64M^2T^{4/3}\sqrt{\log(1/\varepsilon)}\right] \\
&\qquad + 34MnT^{2/3} + 36M\sqrt{n}T^{2/3}\sqrt{\log(1/\varepsilon)} && \text{(By (4))} \\
&\le 4MnT^{2/3} + 8MT^{2/3}\sqrt{\log(1/\varepsilon)} \\
&\qquad + 34MnT^{2/3} + 36M\sqrt{n}T^{2/3}\sqrt{\log(1/\varepsilon)} && \text{(C.f. proof of Thm 16)} \\
&\le 38MnT^{2/3} + 44M\sqrt{n}T^{2/3}\sqrt{\log(1/\varepsilon)}.
\end{aligned}$$

This gives the required bound. ■

5. Conclusions and Open Questions

We have described efficient regret minimization algorithms for submodular cost functions, in both the bandit and full information settings. This parallels the work of Streeter and Golovin (2008) who study two specific instances of online submodular *maximization* (for which the offline problem is NP-hard), and give (approximate) regret minimizing algorithms. We leave it as an open question whether there exists an efficient algorithm that attains $O(\sqrt{T})$ regret bounds for online submodular minimization in the bandit setting.

Acknowledgments

The first author would like to acknowledge support for this project from the Israeli Science Foundation (ISF grant 810/11).

Appendix A. Additional Lemmas

In this section we prove auxiliary lemmas that were used in the paper for completeness.

A.1 The FTL-BTL Lemma

The following stability lemma was essentially proved in Theorem 1.1 of Kalai and Vempala (2005). We reprove it here for completeness:

Lemma 21 *Let* $S_t = \arg\min_{S\subseteq[n]}\{\sum_{\tau=1}^{t-1} f_\tau(S) + R(S)\}$ *as in Algorithm 1. Then*

$$Regret_T \;\leq\; \sum_{t=1}^{T} [f_t(S_t) - f_t(S_{t+1})] + R(S^*) - R(S_1).$$

Where $S^* = \arg\min_{S\subseteq[n]} \sum_{t=1}^{T} f_t(S)$.

Proof For convenience, denote by $f_0 = R$, and assume we start the algorithm from $t = 0$ with an arbitrary S_0. The lemma is now proved by induction on T.
Induction base: Note that by definition, we have that $S_1 = \arg\min_S\{R(S)\}$, and thus $f_0(S_1) \leq f_0(S^*)$ for all S^*, thus $f_0(S_0) - f_0(S^*) \leq f_0(S_0) - f_0(S_1)$.
Induction step: Assume that that for T, we have

$$\sum_{t=0}^{T} f_t(S_t) - f_t(S^*) \quad\leq\quad \sum_{t=0}^{T} f_t(S_t) - f_t(S_{t+1})$$

and let us prove for $T+1$. Since $S_{T+2} = \arg\min_S\{\sum_{t=0}^{T+1} f_t(S)\}$ we have:

$$\begin{aligned}
\sum_{t=0}^{T+1} f_t(S_t) - \sum_{t=0}^{T+1} f_t(S^*) &\leq \sum_{t=0}^{T+1} f_t(S_t) - \sum_{t=0}^{T+1} f_t(S_{T+2}) \\
&= \sum_{t=0}^{T} (f_t(S_t) - f_t(S_{T+2})) + f_{T+1}(S_{t+1}) - f_{T+1}(S_{T+2}) \\
&\leq \sum_{t=0}^{T} (f_t(S_t) - f_t(S_{t+1})) + f_{T+1}(S_{t+1}) - f_{T+1}(S_{T+2}) \\
&= \sum_{t=0}^{T+1} f_t(S_t) - f_t(S_{t+1}).
\end{aligned}$$

Where in the third line we used the induction hypothesis for $S^* = S_{T+2}$. We conclude that

$$\begin{aligned}
\sum_{t=1}^{T} f_t(S_t) - f_t(S^*) &\leq \sum_{t=1}^{T} f_t(S_t) - f_t(S_{t+1}) + [-f_0(S_0) + f_0(S^*) + f_0(S_0) - f_0(S_1)] \\
&= \sum_{t=1}^{T} f_t(S_t) - f_t(S_{t+1}) + [R(S^*) - R(S_1)].
\end{aligned}$$

■

A.2 Proof of Lemma 8

Next, we give a proof of Lemma 8 from the paper of Jegelka and Bilmes (2011), for completeness:

Lemma 8 restatement: *The subgradients g of the Lovász extension $\hat{f} : \mathcal{K} \to [-M, M]$ of a submodular function are bounded by $\|g\|_2 \leq \|g\|_1 \leq 4M$.*

Proof Recall the subgradient definition of proposition 7: Let $x \in \mathcal{K}$. Let $\emptyset = B_0 \subset B_1 \subset B_2 \subset \cdots B_n = [n]$ be an arbitrary maximal chain associated with x, and let $\pi : [n] \to [n]$ be the corresponding permutation. Note that $B_{\pi(i)} = \{j \in [n] : \pi(j) \leq \pi(i)\}$. Then, a subgradient g of $\hat{f}$ at x is given by:

$$g_i \;=\; f(B_{\pi(i)}) - f(B_{\pi(i)-1}).$$

Let $S^+ = \{i : \; g_i \geq 0\}$. First, we claim:

Proposition 22

$$\sum_{i \in S^+} g_i \;\leq\; M - f(\emptyset).$$

Proof Let $\sigma : S^+ \to \{1, 2, \ldots, |S^+|\}$ be the one-to-one mapping that orders the elements of S^+ according to π, that is, for $i, j \in S^+$, we have $\sigma(i) < \sigma(j)$ if and only if $\pi(i) < \pi(j)$. For $i \in [S^+]$, define $C_i = \{j \in S^+ : \; \sigma(j) \leq i\}$, and define $C_0 = \emptyset$. Since σ respects the ordering given by π, for all $i \in S^+$ we have

$$C_{\sigma(i)-1} \;=\; \{j \in S^+ : \; \sigma(j) \leq \sigma(i) - 1\} \;\subseteq\; \{j \in [n] : \; \pi(i) \leq \pi(j) - 1\} \;=\; B_{\pi(i)-1}.$$

Note that $C_{\sigma(i)} = C_{\sigma(i)-1} + i$ and $B_{\pi(i)} = B_{\pi(i)-1} + 1$. Thus by the submodularity of f, we have

$$g_i \;=\; f(B_{\pi(i)}) - f(B_{\pi(i)-1}) \;\leq\; f(C_{\sigma(i)}) - f(C_{\sigma(i)-1}).$$

Thus, we have

$$\begin{aligned}
\sum_{i \in S^+} g_i \;&\leq\; \sum_{i \in S^+} f(C_{\sigma(i)}) - f(C_{\sigma(i)-1}) \\
&=\; \sum_{i=1}^{|S^+|} f(C_i) - f(C_{i-1}) \\
&=\; f(S^+) - f(\emptyset) \\
&\leq\; M - f(\emptyset).
\end{aligned}$$

■

Now let $S^- := [n] \setminus S^+$ be the subset of indices of all negative entries of g. We have

$$\sum_{i \in S^-} g_i \;=\; \sum_{i \in [n]} g_i - \sum_{i \in S^+} g_i \;=\; f([n]) - f(\emptyset) - \sum_{i \in S^+} g_i \;\geq\; -2M.$$

The second equality above follows by the definition of g. Hence, we have

$$\|g\|_1 \;=\; \sum_{i\in S^+} g_i - \sum_{i\in S^-} g_i \;\leq\; 3M - f(\emptyset) \;\leq\; 4M.$$

■

References

Nicolò Cesa-Bianchi and Gábor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006.

Abraham D. Flaxman, Adam Tauman Kalai, and H. Brendan. McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *SODA*, pages 385–394, 2005. ISBN 0-89871-585-7.

Satoru Fujishige. *Submodular Functions and Optimization*. Elsevier, 2005.

Martin Grötschel, Laszlo Lovász, and Alexander Schrijver. *Geometric Algorithms and Combinatorial Optimization*. Springer Verlag, 1988.

Carlos Guestrin and Andreas Krause. Beyond convexity - submodularity in machine learning. In *Tutorial given in the 25rd International Conference on Machine Learning (ICML)*, 2008.

James Hannan. Approximation to Bayes risk in repeated play. *In M. Dresher, A. W. Tucker, and P. Wolfe, editors, Contributions to the Theory of Games, volume III*, pages 97–139, 1957.

Elad Hazan and Satyen Kale. Beyond convexity: Online submodular minimization. In *Twenty-Fourth Annual Conference on. Neural Information Processing Systems (NIPS)*, pages 700–708, 2009.

Satoru Iwata. A faster scaling algorithm for minimizing submodular functions. *SIAM J. Comput.*, 32(4):833–840, 2003.

Satoru Iwata and James B. Orlin. A simple combinatorial algorithm for submodular function minimization. In *SODA '09: Proceedings of the Nineteenth Annual ACM -SIAM Symposium on Discrete Algorithms*, pages 1230–1237, Philadelphia, PA, USA, 2009. Society for Industrial and Applied Mathematics.

Satoru Iwata, Lisa Fleischer, and Satoru Fujishige. A combinatorial strongly polynomial algorithm for minimizing submodular functions. *J. ACM*, 48:761–777, 2001.

Stefanie Jegelka and Jeff Bilmes. Online submodular minimization for combinatorial structures. In Lise Getoor and Tobias Scheffer, editors, *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, ICML '11, pages 345–352, New York, NY, USA, June 2011. ACM. ISBN 978-1-4503-0619-5.

Adam Kalai and Santosh Vempala. Efficient algorithms for online decision problems. *Journal of Computer and System Sciences*, 71(3):291–307, 2005.

Andreas Krause and Carlos Guestrin. Submodularity and its applications in optimized information gathering. *ACM Transactions on Intelligent Systems and Technology*, 2(4), July 2011.

Nick Littlestone and Manfred K. Warmuth. The weighted majority algorithm. In *FOCS*, pages 256–261. IEEE Computer Society, 1989.

S. Thomas McCormick. Submodular function minimization. In *Chapter 7 in the Handbook on Discrete Optimization*, pages 321–391. Elsevier, 2006.

James B. Orlin. A faster strongly polynomial time algorithm for submodular function minimization. *Math. Program.*, 118(2):237–251, 2009. ISSN 0025-5610. doi: http://dx.doi.org/10.1007/s10107-007-0189-2.

Alexander Schrijver. A combinatorial algorithm minimizing submodular functions in strongly polynomial time. *Journal of Combinatorial Theory, Series B*, 80(2):346–355, 2000.

Matthew J. Streeter and Daniel Golovin. An online algorithm for maximizing submodular functions. In *NIPS*, pages 1577–1584, 2008.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the Twentieth International Conference (ICML)*, pages 928–936, 2003.