

---

# Projection-Free Online Optimization with Stochastic Gradient: From Convexity to Submodularity

---

Lin Chen<sup>1,2</sup> Christopher Harshaw<sup>1,3</sup> Hamed Hassani<sup>4</sup> Amin Karbasi<sup>1,2</sup>

## Abstract

Online optimization has been a successful framework for solving large-scale problems under computational constraints and partial information. Current methods for online convex optimization require either a projection or exact gradient computation at each step, both of which can be prohibitively expensive for large-scale applications. At the same time, there is a growing trend of non-convex optimization in machine learning community and a need for online methods. Continuous DR-submodular functions, which exhibit a natural diminishing returns condition, have recently been proposed as a broad class of non-convex functions which may be efficiently optimized. Although online methods have been introduced, they suffer from similar problems. In this work, we propose **Meta-Frank-Wolfe**, the first online projection-free algorithm that uses stochastic gradient estimates. The algorithm relies on a careful sampling of gradients in each round and achieves the optimal  $O(\sqrt{T})$  adversarial regret bounds for convex and continuous submodular optimization. We also propose **One-Shot Frank-Wolfe**, a simpler algorithm which requires only a single stochastic gradient estimate in each round and achieves an  $O(T^{2/3})$  stochastic regret bound for convex and continuous submodular optimization. We apply our methods to develop a novel “lifting” framework for the online discrete submodular maximization and also see that they outperform current state-of-the-art techniques on various experiments.

## 1. Introduction

As the amount of collected data becomes massive in both size and complexity, algorithm designers are faced with unprecedented challenges in statistics, machine learning, and control. In the past decade, online optimization has provided a successful computational framework for tackling a wide variety of challenging problems, ranging from non-parametric regression to portfolio management (Candriello et al., 2017; Agarwal et al., 2006). In online optimization, a large or complex optimization problem is broken down into a sequence of smaller optimization problems, each of which must be solved with limited information. This framework captures many real-world scenarios in which standard optimization theory does not apply. For instance, a machine learning application cannot feasibly process terabytes of data at a single time; rather, subsets of data may be handled in a sequential fashion. Another example is when the true objective function is the expectation of an unknown distribution of functions, and may only be accessible via samples, as is the case for problems in online learning and control theory (Xiao, 2010; Wang & Boyd, 2008).

Online convex optimization, a branch of online optimization that considers sequentially minimizing convex functions, has proved particularly useful for statistical and machine learning applications. Online convex optimization has enjoyed much success in these areas because most offline machine learning techniques utilize the existing theory of convex optimization. As in the offline setting, gradient methods are a popular class of algorithms for online convex optimization due to their simplicity; however, they require projections onto the constraint set, which involve solving a quadratic program in the general case. These projections are infeasible for large scale applications with complicated constraints such as matrix completion, network routing problems, and maximum matchings. Online projection-free methods have been proposed and are much more efficient, replacing a projection onto the constraint set with a linear optimization over the constraint set at each iteration (Hazan & Kale, 2012; Garber & Hazan, 2013). However, these projection-free methods require exact gradient computations, which may be prohibitively expensive for even moderately sized data sets and intractable when a

---

<sup>1</sup>Yale Institute for Network Science, Yale University, New Haven, CT, USA <sup>2</sup>Department of Electrical Engineering, Yale University <sup>3</sup>Department of Computer Science, Yale University <sup>4</sup>Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA, USA. Correspondence to: Lin Chen <lin.chen@yale.edu>.

closed form does not exist. Thus, there is a huge need for online convex optimization routines that are projection-free and also robust to stochastic gradient estimates.

While convex programs may be efficiently solved (at least in theory), there is a growing number of non-convex problems arising in machine learning and statistics. Notable examples include nonnegative principle component analysis, low-rank matrix recovery, sigmoid loss functions for binary classification, and the training of deep neural networks, to name a few. Understanding which types of non-convex functions may be efficiently optimized and developing techniques for doing so is a pressing research question for both theory and practice. Recently, continuous DR-submodular functions have been proposed as a broad class of non-convex functions which admit efficient approximate maximization routines, even though exact maximization is NP-Hard (Bian et al., 2017). These functions capture many real-life applications, such as optimal experiment design, non-definite quadratic programming, coverage and diversity functions, and continuous relaxation of discrete submodular functions. Recent works (Chen et al., 2018) have proposed methods for online continuous DR-submodular optimization; however, these too require either expensive projections or exact gradient computations.

**Our contributions** In this paper, we present a suite of projection-free algorithms for online optimization that use stochastic estimates of the gradient and leverage the averaging technique (Mokhtari et al., 2018a;b) to reduce their variance. This includes

- **Meta-Frank-Wolfe**, the first projection-free algorithm for adversarial online optimization which requires only stochastic gradient estimates. The algorithm relies on a careful sampling of gradients in each round and achieves optimal  $O(\sqrt{T})$  regret and  $(1 - 1/e)$ -regret bounds for convex and submodular optimization, respectively.
- **One-Shot Frank-Wolfe**, a simpler projection-free algorithm for stochastic online optimization which requires only a single stochastic gradient estimate in each round. This simpler algorithm achieves  $O(T^{2/3})$  regret and  $(1 - 1/e)$ -regret bounds for the convex and submodular case, respectively.
- A novel class of algorithms for online discrete submodular optimization which are based on lifting discrete functions to the continuous domain, applying our methods with an extremely efficient sampling technique, and using rounding schemes to produce a discrete solution.

Finally, to demonstrate the effectiveness of our algorithms,

we tested their performance on an extensive set of experiments and measured against common baselines.

## 2. Related Work

The Frank-Wolfe algorithm, also known as the conditional gradient descent, was originally proposed for the offline setting in (Frank & Wolfe, 1956). The framework of online convex optimization was introduced by Zinkevich (2003), in which the online projected gradient descent was proposed and proved to achieve an  $O(\sqrt{T})$  regret bound. However, the projections required for such an algorithm are too expensive for many large-scale online problems. The online conditional gradient descent was the first projection-free online algorithm, originally proposed in (Hazan & Kale, 2012). An improved conditional gradient algorithm was later designed for smooth and strongly convex optimization which achieves the optimal  $O(\sqrt{T})$  adversarial regret bound (Garber & Hazan, 2013). However, both of these algorithms can perform arbitrarily poorly if supplied with stochastic gradient estimates. Lafond et al. (2015) proposed an online Frank-Wolfe variant for the any-time stochastic online setting that converges to a stationary point for non-convex expected functions. While convergence is an important property of the any-time methods, arbitrary stationary points do not yield approximation guarantees for general non-convex functions.

Johnson & Zhang (2013) introduced the variance reduction technique for accelerating stochastic gradient descent. It was independently discovered by Mahdavi et al. (2013). Allen-Zhu & Hazan (2016) applied this technique to non-convex optimization. Hazan & Luo (2016) devised a projection-free stochastic convex optimization algorithm based on this technique. Mokhtari et al. (2018a;b) proposed the first sample-efficient variance reduction technique for projection-free algorithms that does not require increasing batch sizes. Their method achieves the tight  $(1 - 1/e)$  approximation guarantee for monotone and continuous DR-submodular functions. Although these variance reduction techniques have enjoyed success in the offline setting, they have yet to be as extensively applied in the online setting that we consider in this paper.

In the discrete domain, Streeter & Golovin (2009) studied the online maximization problem of monotone submodular set functions subject to a knapsack constraint and introduced the meta-action technique. In a celebrated work, Calinescu et al. (2011) proposed an (offline) method for maximizing monotone submodular set functions subject to a matroid constraint by working in the continuous domain via the multilinear extension, then rounding the fractional solution. By combining the meta-action and lifting techniques, Golovin et al. (2014) presented an algorithm whose  $(1 - 1/e)$ -regret is bounded by  $O(\sqrt{T})$ . The lifting method therein relies

on an expensive sampling procedure that does not scale favorably to large applications.

Bach (2015) demonstrated connections between continuous submodular functions and convex functions in the context of minimization. Building upon the continuous greedy algorithm of (Calinescu et al., 2011), Bian et al. (2017) proposed an algorithm that achieves a  $(1 - 1/e)$ -approximation guarantee for maximizing monotone continuous DR-submodular functions subject to down-closed convex constraints. Projected gradient methods were investigated in (Hassani et al., 2017) and were shown to attain a  $1/2$ -approximation ratio for monotone continuous DR-submodular functions. Very recently, Chen et al. (2018) borrowed the idea of meta-action (Streeter & Golovin, 2009) and proposed several online algorithms for maximizing monotone continuous DR-submodular functions. However, each of these methods either requires an expensive projection step at each iteration or cannot handle stochastic gradient estimates.

### 3. Preliminaries

In this work, we are interested in optimizing two classes of functions, namely convex and continuous DR-submodular. To begin defining continuous submodular functions, we first recall the definition of a submodular set function. A real-valued set function  $f : 2^\Omega \rightarrow \mathbb{R}_+$  is *submodular* if

$$f(A) + f(B) \geq f(A \cup B) + f(A \cap B)$$

for all  $A, B \subset \Omega$ . The notion of submodularity has been extended to continuous domains (Wolsey, 1982; Vondrák, 2007; Bach, 2015). Consider a function  $f : \mathcal{X} \rightarrow \mathbb{R}_+$  where the domain is of the form  $\mathcal{X} = \prod_{i=1}^n \mathcal{X}_i$  and each  $\mathcal{X}_i$  is a compact subset of  $\mathbb{R}_+$ . We say that  $f$  is *continuous submodular* if  $f$  is continuous and for all  $\mathbf{x}, \mathbf{y} \in \mathcal{X}$ , we have

$$f(\mathbf{x}) + f(\mathbf{y}) \geq f(\mathbf{x} \vee \mathbf{y}) + f(\mathbf{x} \wedge \mathbf{y})$$

where  $\mathbf{x} \vee \mathbf{y}$  and  $\mathbf{x} \wedge \mathbf{y}$  are component-wise maximum and minimum, respectively. Note that we have defined both discrete and continuous functions to be nonnegative on their respective domains. For efficient maximization, we also require that these functions satisfy a diminishing returns condition (Bian et al., 2017). We say that  $f$  is *continuous DR-submodular* if  $f$  is differentiable and

$$\nabla f(\mathbf{x}) \geq \nabla f(\mathbf{y})$$

for all  $\mathbf{x} \leq \mathbf{y}$ . The main attraction of continuous DR-submodular functions is that they are concave in positive directions; that is, for all  $\mathbf{x} \leq \mathbf{y}$ ,

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle$$

(Calinescu et al., 2011; Bian et al., 2017). A function  $f$  is *monotone* if  $f(\mathbf{x}) \leq f(\mathbf{y})$  for all  $\mathbf{x} \leq \mathbf{y}$ . A function  $f$  is *L-smooth* if  $\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\| \leq L\|\mathbf{x} - \mathbf{y}\|$  for all  $\mathbf{x}, \mathbf{y}$ .

We now provide a brief introduction to online optimization, referring the interested reader to the excellent survey of (Hazan et al., 2016). In the online setting, a player seeks to iteratively optimize a sequence of functions  $f_1, \dots, f_T$  over  $T$  rounds. In each round, a player must first choose a point  $\mathbf{x}_t$  from the constraint set  $\mathcal{K}$ . After playing  $\mathbf{x}_t$ , the value of  $f_t(\mathbf{x}_t)$  is revealed to the player, along with access to the gradient  $\nabla f$ . Although the player does not know the function  $f_t$  while choosing  $\mathbf{x}_t$ , they may use information of previously seen functions to guide their choice. The situation where an arbitrary sequence of functions  $f_1, \dots, f_T$  is presented is known as the *adversarial online setting*. In the adversarial setting, the goal of the player is to minimize *adversarial regret*, which is defined as

$$\mathcal{R}_T \triangleq \sum_{t=1}^T f_t(\mathbf{x}_t) - \inf_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T f_t(\mathbf{x})$$

for minimization problems and analogously defined for maximization problems. Intuitively, a player's regret is low if the accumulated value of their actions over the  $T$  rounds is close to that of the single best action in hindsight. Indeed, this is a natural framework for data-intensive applications where the entire data may not fit onto a single disk and thus needs to be processed in  $T$  batches. The algorithm designer would like to devise a scheme to process the  $T$  batches separately in a way that is competitive with the best single disk solution.

A slightly different formulation known as *stochastic online setting* is when the functions are chosen *i.i.d.* from some unknown distribution  $f_t \sim \mathcal{D}$ . In this case, the player seeks to minimize *stochastic regret*, which is defined as

$$\mathcal{SR}_T \triangleq \sum_{t=1}^T f(\mathbf{x}_t) - T \cdot \inf_{\mathbf{x} \in \mathcal{K}} f(\mathbf{x})$$

where  $f(\mathbf{x}) = \mathbb{E}_{f_t \sim \mathcal{D}}[f_t(\mathbf{x})]$  denotes the expected function. This is a natural framework for many statistical and machine learning applications, such as empirical risk minimization, where the true objective is unknown but pairs of data points and labels are sampled. While the stochastic setting appears "easier" than the adversarial setting (in the sense that any strategy for the adversarial settings applies to stochastic settings and obtains a potentially lower regret), the strategies designed for the stochastic setting may be much simpler and more computationally efficient. For both adversarial and stochastic settings, a strategy that achieves a regret that is sublinear in  $T$  is considered good and  $O(\sqrt{T})$  regret bounds are optimal for convex functions in both settings. Although convex programs can be efficiently solved to high accuracy, general non-convex programs cannot be efficiently exactly optimized, thus necessitating another definition of regret.

The  $\alpha$ -regret is defined as

$$\alpha\text{-}\mathcal{R}_T \triangleq \alpha \sup_{\mathbf{x} \in \mathcal{K}} \sum_{t=1}^T f_t(\mathbf{x}) - \sum_{t=1}^T f_t(\mathbf{x}_t)$$

for adversarial maximization problems, and may be analogously extended to other scenarios. Intuitively,  $\alpha$ -regret compares a player's actions with the best  $\alpha$ -approximation to the optimal solution in hindsight. This is appropriate when the objective functions do not admit efficient optimization routines, but do admit constant-factor approximations, as is the case with continuous DR-submodular functions.

Nearly all optimization methods for both offline and online settings use first order information of the objective function; however, exact gradient computations can be costly, especially when the objective function is only readily expressed as a large sum of individual functions or is itself an expectation over an unknown distribution. In this case, stochastic estimates are usually much more computationally efficient to obtain via sampling or simulation. In this work, we assume that once a function  $f_t$  is revealed, the player gains oracle access to unbiased stochastic estimates of the gradient, rather than the exact gradient. More precisely, the player may query the oracle to obtain a random linear function  $\tilde{\nabla} f(x)$  such that  $\mathbb{E}[\nabla f(x) - \tilde{\nabla} f(x)] = 0$  for all  $x$ . This computational model captures commonly used mini-batch methods for estimating gradients, among other examples. In this work, we make a few main assumptions that allow our algorithms to be analyzed.

**Assumption 1.** The constraint set  $\mathcal{K}$  is convex and compact, with diameter  $D = \sup_{\mathbf{x}, \mathbf{y} \in \mathcal{K}} \|\mathbf{x} - \mathbf{y}\|$  and radius  $R = \sup_{\mathbf{x} \in \mathcal{K}} \|\mathbf{x}\|$ .

**Assumption 2.** In the adversarial setting, each function  $f_t$  is  $L$ -smooth and in the stochastic setting, the expected function  $f$  is  $L$ -smooth.

**Assumption 3.** In the adversarial setting, the gradient oracle is unbiased  $\mathbb{E}[\nabla f_t(\mathbf{x}) - \tilde{\nabla} f_t(\mathbf{x})] = 0$  and has a bounded variance  $\mathbb{E}[\|\nabla f_t(\mathbf{x}) - \tilde{\nabla} f_t(\mathbf{x})\|^2] \leq \sigma^2$  for all points  $\mathbf{x}$  and functions  $f_t$ . In the stochastic setting, the gradient oracle is unbiased  $\mathbb{E}[\nabla f(\mathbf{x}) - \tilde{\nabla} f_t(\mathbf{x})] = 0$  and has a bounded variance  $\mathbb{E}[\|\nabla f(\mathbf{x}) - \tilde{\nabla} f_t(\mathbf{x})\|^2] \leq \sigma^2$  for all points  $\mathbf{x}$  and functions  $f_t$ .

We remark that in the stochastic setting and under mild regularity conditions, unbiasedness of the gradients  $\mathbb{E}[\nabla f_t(\mathbf{x}) - \tilde{\nabla} f_t(\mathbf{x})] = 0$  implies unbiasedness  $\mathbb{E}[\nabla f(\mathbf{x}) - \tilde{\nabla} f_t(\mathbf{x})] = 0$  in Assumption 3 because  $f(\mathbf{x}) = \mathbb{E}_{f_t \sim \mathcal{D}}[f_t(\mathbf{x})]$ . Moreover, upper bounds on the variance terms  $\mathbb{E}[\|\nabla f(\mathbf{x}) - \nabla f_t(\mathbf{x})\|^2] \leq \sigma_a^2$  and  $\mathbb{E}[\|\nabla f_t(\mathbf{x}) - \tilde{\nabla} f_t(\mathbf{x})\|^2] \leq \sigma_b^2$  yield a variance bound of  $\mathbb{E}[\|\nabla f(\mathbf{x}) - \tilde{\nabla} f_t(\mathbf{x})\|^2] \leq \sigma_a^2 + \sigma_b^2$ , by the triangle inequality.

## 4. Main Results

We now present two algorithms for online optimization of convex and continuous DR-submodular functions in the adversarial and stochastic settings. Unlike previous work, these methods are projection-free and require only *stochastic estimates* of the gradients, rather than exact gradient computations. In both algorithms, the main computational primitive is linear optimization over a compact convex set. In addition, we remark that both algorithms can be converted into an anytime algorithm that does not require the knowledge of the horizon  $T$  via the doubling trick; see Section 2.3.1 of (Shalev-Shwartz et al., 2012).

### 4.1. Adversarial Online Setting

Algorithm 1 combines the recent variance reduction technique of (Mokhtari et al., 2018a) along with the use of online linear optimization oracles to minimize the regret in each round. An online linear optimization oracle is an instance of an online linear optimization (minimization/maximization in the convex/DR-submodular setting, respectively) algorithm that optimizes linear objectives in a sequential manner. Both the variance reduction in the stochastic gradient estimates and the online linear oracles are crucial in the algorithm, as just one technique is not enough to get sub-linear regret bounds in the adversarial setting. At a high level, our algorithm produces iterates  $x_t$  by running  $K$  steps of a Frank-Wolfe procedure, using an average of previous gradient estimates and linear online optimization oracles in place of exact optimization of the true gradient. After a point  $x_t$  is played in round  $t$ , our algorithm queries the gradient oracle  $\tilde{\nabla} f_t$  at  $K$  points. Then, the gradient estimates are averaged with those from previous rounds and fed as objective functions into  $K$  linear online optimization oracles. The  $K$  points chosen by the oracles are used as iterates in a full  $K$ -step Frank-Wolfe subroutine to obtain the next point  $x_{t+1}$ . A formal description is provided in Algorithm 1.

There are only a few differences in Algorithm 1 for convex and submodular optimization. First, the online oracles should be minimizing in the case of convex optimization and maximizing in the case of submodular optimization. Second, the initial point  $\mathbf{x}_1$  may be any point in  $\mathcal{K}$  for convex problems but should be set to 0 for submodular problems (even if  $\mathcal{K}$  is not down-closed). Finally, the update rule is

$$\mathbf{x}_t^{(k+1)} \leftarrow (1 - \eta_k) \mathbf{x}_t^{(k)} + \eta_k \mathbf{v}_t^{(k)}$$

for convex problems and

$$\mathbf{x}_t^{(k+1)} \leftarrow \mathbf{x}_t^{(k)} + \eta_k \mathbf{v}_t^{(k)}$$

for submodular problems. We now provide a formal regret bound.

---

**Algorithm 1** Meta-Frank-Wolfe
 

---

**Input:** convex set  $\mathcal{K}$ , time horizon  $T$ , linear optimization oracles  $\mathcal{E}^{(1)} \dots \mathcal{E}^{(K)}$ , step sizes  $\rho_k \in (0, 1)$  and  $\eta_k \in (0, 1)$ , and initial point  $\mathbf{x}_1$

**Output:**  $\{\mathbf{x}_t : 1 \leq t \leq T\}$

- 1: Initialize online linear optimization oracles  $\mathcal{E}^{(1)} \dots \mathcal{E}^{(K)}$
  - 2: Initialize  $\mathbf{d}_t^{(0)} = 0$  and  $\mathbf{x}_t^{(1)} = \mathbf{x}_1$
  - 3: **for**  $t \leftarrow 1, 2, 3, \dots, T$  **do**
  - 4:  $\mathbf{v}_t^{(k)} \leftarrow$  output of oracle  $\mathcal{E}^{(k)}$  in round  $t - 1$
  - 5:  $\mathbf{x}_t^{(k+1)} \leftarrow \text{update}(\mathbf{x}_t^{(k)}, \mathbf{v}_t^{(k)}, \eta_k)$  for  $k = 1 \dots K$
  - 6: Play  $\mathbf{x}_t = \mathbf{x}_t^{(K+1)}$ , then obtain value  $f_t(\mathbf{x}_t)$  and unbiased oracle access to  $\nabla f_t$
  - 7:  $\mathbf{d}_t^{(k)} \leftarrow (1 - \rho_k)\mathbf{d}_t^{(k-1)} + \rho_k \tilde{\nabla} f_t(\mathbf{x}_t^{(k-1)})$  for  $k = 1 \dots K$
  - 8: Feedback  $\langle \mathbf{v}_t^{(k)}, \mathbf{d}_t^{(k)} \rangle$  to  $\mathcal{E}^{(k)}$  for  $k = 1 \dots K$
  - 9: **end for**
- 

**Theorem 1** (Proof in Appendices B and C). *Suppose Assumptions 1 - 3 hold, the online linear optimization oracles have regret at most  $\mathcal{R}_T^\mathcal{E}$ , and the averaging parameters are chosen as  $\rho_k = \frac{2}{(k+3)^{2/3}}$ . Then for convex functions  $f_1, \dots, f_T$  and step sizes  $\eta_k = \frac{1}{k+3}$ , the adversarial regret of Algorithm 1 is at most*

$$\frac{4TDQ^{1/2}}{K^{1/3}} + \frac{4T}{K} \left( M + \frac{LD^2}{3} \log(K+1) \right) + \frac{4}{3} \mathcal{R}_T^\mathcal{E}$$

in expectation, where  $M = \max_{1 \leq t \leq T} [f_t(\mathbf{x}_1) - f_t(\mathbf{x}^*)]$  and  $Q \triangleq \max\{4^{2/3} \max_{1 \leq t \leq T} \|\nabla f_t(\mathbf{x}_1)\|^2, 4\sigma^2 + 3(LD)^2/2\}$ . For monotone continuous DR-submodular functions  $f_1, \dots, f_T$  and step sizes  $\eta_k = \frac{1}{k}$ , the adversarial  $(1 - 1/e)$ -regret of Algorithm 1 is at most

$$\frac{3TDQ^{1/2}}{2K^{1/3}} + \frac{LD^2T}{2K} + \mathcal{R}_T^\mathcal{E}$$

in expectation, where  $Q \triangleq \max\{\max_{1 \leq t \leq T} \|\nabla f_t(\mathbf{x}_1)\|^2 4^{2/3}, 4\sigma^2 + 6L^2R^2\}$ .

From Theorem 1, we observe that by setting  $K = T^{3/2}$  and choosing a projection-free online linear optimization oracle with  $\mathcal{R}_T^\mathcal{E} = O(\sqrt{T})$ , such as Follow the Perturbed Leader (Cohen & Hazan, 2015), both regrets are bounded above by  $O(\sqrt{T})$ . We remark that the expectation in Theorem 1 is with respect to the stochastic gradient estimates.

## 4.2. Stochastic Online Setting

In the stochastic online setting, where functions are sampled i.i.d.  $f_t \sim \mathcal{D}$ , we can develop much simpler algorithms that still achieve sublinear regret. Algorithm 2 works without

instantiating any online linear optimization oracles and requires only a single stochastic estimate of the gradient at each round. Indeed, because the functions are not arbitrarily chosen, variance reduction along with one Frank-Wolfe step suffices to achieve a sublinear regret bound.

---

**Algorithm 2** One-Shot Frank-Wolfe
 

---

**Input:** convex set  $\mathcal{K}$ , time horizon  $T$ , step sizes  $\rho_t \in (0, 1)$  and  $\eta_t \in (0, 1)$ , and initial point  $\mathbf{x}_1$

**Output:**  $\{\mathbf{x}_t : 1 \leq t \leq T\}$

- 1:  $\mathbf{d}_0 \leftarrow 0$
  - 2: **for**  $t \leftarrow 1, 2, 3, \dots, T$  **do**
  - 3: Play  $\mathbf{x}_t$ , then obtain value  $f_t(\mathbf{x}_t)$  and unbiased oracle access to  $\nabla f_t$
  - 4:  $\mathbf{d}_t \leftarrow (1 - \rho_t)\mathbf{d}_{t-1} + \rho_t \tilde{\nabla} f_t(\mathbf{x}_t)$
  - 5:  $\mathbf{v}_t \leftarrow \arg \max_{\mathbf{v} \in \mathcal{K}} \langle \mathbf{d}_t, \mathbf{v} \rangle$
  - 6:  $\mathbf{x}_{t+1} \leftarrow \text{update}(\mathbf{x}_t, \mathbf{v}_t, \eta_t)$
  - 7: **end for**
- 

The differences in Algorithm 2 for convex and submodular optimization are similar to those in Algorithm 1. Namely, the update rules are the same and the initial point  $\mathbf{x}_1$  may be arbitrarily chosen from  $\mathcal{K}$  for convex optimization, and set to 0 for submodular optimization.

**Theorem 2** (Proof in Appendices D and E). *Suppose Assumptions 1 - 3 hold and the averaging parameters are chosen as  $\rho_t = \frac{2}{(t+3)^{2/3}}$ . Then for a convex expected function  $f$  and step sizes  $\eta_t = \frac{1}{t+3}$ , the stochastic regret of Algorithm 1 is at most*

$$4M \log(T+1) + 6Q^{1/2}DT^{2/3} + \frac{4}{3}LD^2 \log^2(T+3)$$

in expectation, where  $M = f(\mathbf{x}_1) - f(\mathbf{x}^*)$  and  $Q \triangleq \max\{4^{2/3} \|\nabla F(\mathbf{x}_1)\|^2, 4\sigma^2 + 3(LD)^2/2\}$ . For expected functions  $f$  which are monotone continuous DR-submodular and step sizes  $\eta_k = \frac{1}{k}$ , the stochastic  $(1 - 1/e)$ -regret of Algorithm 2 is at most

$$(1 - 1/e)M + \frac{3DQ^{1/2}}{10}(3T^{2/3} + 2T^{-1}) + \frac{LD^2}{2}.$$

in expectation, where  $M = f(\mathbf{x}^*) - f(0)$  and  $Q \triangleq \max\{\|\nabla f(0)\|^2 4^{2/3}, 4\sigma^2 + 6L^2R^2\}$

## 4.3. Lifting Methods for Discrete Online Optimization

One exciting application of our online continuous DR-submodular optimization algorithms is a new approach for online discrete submodular optimization. While previous methods could only handle knapsack constraints (Streeter & Golovin, 2009) or required expensive sampling procedures (Golovin et al., 2014), our continuous methods can

be applied to the discrete setting to handle general matroid constraints and computationally cheap sampling procedures.

Suppose  $f_1, \dots, f_T$  are nonnegative monotone submodular set functions on a ground set  $\Omega$  with matroid constraint  $\mathcal{I}$  and  $\bar{f}_1, \dots, \bar{f}_T$  are corresponding multi-linear extensions with matroid polytope  $\mathcal{K} \subset [0, 1]^n$ . A discrete procedure that uses our continuous algorithm is as follows: at each round  $t$ , the online continuous algorithm produces a fractional solution  $\mathbf{x}_t \in \mathcal{K}$ , which is then rounded to a set  $X_t \in \mathcal{I}$  and played as the discrete solution. The value  $f_t(X_t)$  is revealed and the player is granted access to the discrete function  $f_t$ . Then, the player supplies the continuous algorithm with a stochastic gradient estimate  $\tilde{\nabla} \bar{f}_t$  obtained by a single function evaluation, as

$$\frac{\partial f_t(\mathbf{x})}{\partial x_i} = \mathbb{E}[f(R \cup \{i\}) - f(R)], \quad \forall i \in [n], \quad (1)$$

where  $R$  is random subset of  $[n] \setminus \{i\}$  such that for every  $j \in [n] \setminus \{i\}$ , the event  $j \in R$  happens with an independent probability of  $x_j$ . Because a lossless rounding scheme is used, the discrete player enjoys a regret that is no worse than that of the continuous solution. Provably lossless rounding schemes include the pipage rounding (Ageev & Sviridenko, 2004; Calinescu et al., 2011) and contention resolution (Vondrák et al., 2011).

Most discrete submodular maximization algorithms that go through the multi-linear extension require a gradient estimate with high accuracy. In order to do this, they appeal to a concentration bound, which requires  $O(n^2)$  evaluations of the discrete function for independently chosen samples. In stark contrast, our algorithms can handle stochastic gradient estimates and thus require only a single function evaluation, finally making continuous methods a reality for large-scale online discrete optimization problems. The framework of the one-sampling lifting method is illustrated in Fig. 1.

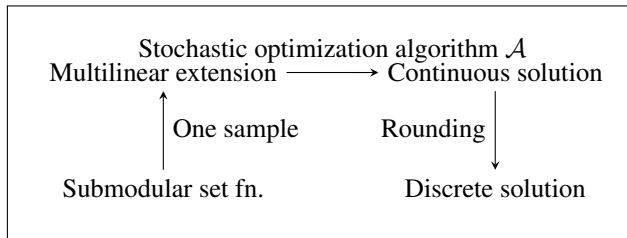


Figure 1. Diagram of the one-sample lifting method

As an example, we present in Algorithm 3 how to use **Meta-Frank-Wolfe** as an online maximization algorithm of submodular set functions. According to Theorem 1, the  $(1 - 1/e)$ -regret of Algorithm 3 is bounded by  $\frac{3TDQ^{1/2}}{K^{1/3}} + \frac{LD^2T}{2K} + \mathcal{R}_T^\mathcal{E}$ , where  $\mathcal{R}_T^\mathcal{E}$  is the regret of  $\mathcal{E}^{(k)}$  up to horizon  $T$ . If one sets  $\mathcal{E}^{(k)}$  to an online linear maximization algo-

rithm with regret bound  $O(\sqrt{T})$  and sets  $K = T^{3/2}$ , the  $(1 - 1/e)$ -regret is at most  $O(\sqrt{T})$ .

---

### Algorithm 3 Meta-Frank-Wolfe for online discrete submodular maximization

---

**Input:** matroid constraint  $\mathcal{I}$ , time horizon  $T$ , linear optimization oracles  $\mathcal{E}^{(1)} \dots \mathcal{E}^{(K)}$ , step sizes  $\rho_k \in (0, 1)$  and  $\eta_k \in (0, 1)$ , and initial point  $\mathbf{x}_1$

**Output:**  $\{X_t : 1 \leq t \leq T\}$

- 1: Initialize online linear optimization oracles  $\mathcal{E}^{(1)} \dots \mathcal{E}^{(K)}$ , setting the constraint set to the matroid polytope of  $\mathcal{I}$
  - 2: Initialize  $\mathbf{d}_t^{(0)} = 0$  and  $\mathbf{x}_t^{(1)} = \mathbf{x}_1$
  - 3: **for**  $t \leftarrow 1, 2, 3, \dots, T$  **do**
  - 4:    $\mathbf{v}_t^{(k)} \leftarrow$  output of oracle  $\mathcal{E}^{(k)}$  in round  $t - 1$
  - 5:    $\mathbf{x}_t^{(k+1)} \leftarrow$  update( $\mathbf{x}_t^{(k)}, \mathbf{v}_t^{(k)}, \eta_k$ ) for  $k = 1 \dots K$
  - 6:    $\mathbf{x}_t \leftarrow \mathbf{x}_t^{(K+1)}$
  - 7:   play  $X_t \leftarrow$  round( $\mathbf{x}_t$ ), obtain value  $f_t(X_t)$  and observe the function  $f_t$
  - 8:   Sample  $\tilde{\nabla} \bar{f}_t(\mathbf{x}_t^{(k)})$  for  $k = 0, \dots, K - 1$
  - 9:    $\mathbf{d}_t^{(k)} \leftarrow (1 - \rho_k)\mathbf{d}_t^{(k-1)} + \rho_k \tilde{\nabla} f_t(\mathbf{x}_t^{(k-1)})$  for  $k = 1 \dots K$
  - 10:   Feedback  $\langle \mathbf{v}_t^{(k)}, \mathbf{d}_t^{(k)} \rangle$  to  $\mathcal{E}^{(k)}$  for  $k = 1 \dots K$
  - 11: **end for**
- 

## 5. Experiment

In this section, we test our online algorithms for monotone continuous DR-submodular and convex optimization on both real-world and synthetic data sets. We find that our algorithms outperform most baselines, including projected gradient descent, when supplied with stochastic gradient estimates. All code was written in the Julia programming language and tested on a Macintosh desktop with an Intel Processor i7 with 16 GB of RAM. No parts of the code were optimized past basic Julia usage. A list of all algorithms to be compared in this section is presented below.

- **Meta-Frank-Wolfe** is Algorithm 1. We compare the variance-reduced meta-Frank-Wolfe algorithm and the analogue without variance reduction, denoted **Meta-FW w/ VR** and **Meta-FW w/o VR**, respectively.
- **One-shot Frank-Wolfe** is Algorithm 2. We compare the One-shot online Frank-Wolfe algorithm with and without variance reduction, denoted **OS-FW w/ VR** and **OS-FW w/o VR**, respectively.
- **Regularized online Frank-Wolfe** is referred to as the online conditional gradient algorithm in (Hazan et al., 2016). It has a regularizer term when computing the gradient. Thus we term it the regularized online Frank-Wolfe algorithm and denote it as **Regularized-OFW**.

- Online projected gradient ascent (OGA) follows the direction of the projected gradient. Its  $1/2$ -regret is at most  $O(\sqrt{T})$  for online monotone continuous DR-submodular maximization if the step size is set to  $\Theta(1/\sqrt{t})$  on the  $t$ -th iteration (Chen et al., 2018). Note that OGA is not a projection-free algorithm. In the setting of convex minimization, we use online projected gradient descent instead (denoted by OGD).
- When we perform experiments on discrete submodular maximization problems using our lifting method, we also compare the above algorithms with the Online Greedy algorithm (Streeter & Golovin, 2009).

### 5.1. Online DR-Submodular Maximization

In order to test the performance of algorithms for online maximization of monotone continuous DR-submodular functions with stochastic gradient estimates, we conducted three sets of experiments on real-world datasets. We approximate the  $(1 - 1/e)$ -regret by running an offline Frank Wolfe maximization to produce a solution that is a  $(1 - 1/e)$  approximation to the optimum.

**Joke Recommendations (Continuous)** The first set of experiments is to optimize a sequence of continuous facility location objectives on the Jester dataset (Goldberg et al., 2001). It contains ratings of 100 jokes from 73,421 users and the rating range is  $[-10, 10]$ . We re-scale the rating range into  $[0, 20]$  so that all ratings are non-negative. Let  $R_{uj}$  be user  $u$ 's rating of joke  $j$ . All users are splitted into disjoint batches  $B_1, B_2, \dots, B_T$ , each containing  $B$  users. The facility location objective is defined as  $f_t(X) = \sum_{u \in B_t} \max_{j \in X} R_{uj}$ ,  $\forall X \subseteq [J]$ , where  $J = 100$  is the total number of jokes and  $[J] = \{1, 2, 3, \dots, J\}$ . Its multilinear extension is given by  $\bar{f}_t(\mathbf{x}) = \sum_{u \in B_t} \sum_{l=1}^J R_{uj_u^l} \mathbf{x}_{j_u^l} \prod_{m=1}^{l-1} (1 - \mathbf{x}_{j_u^m})$ ,  $\forall \mathbf{x} \in [0, 1]^J$ , where  $j_u^1, j_u^2, \dots, j_u^J$  is a permutation of  $1, 2, \dots, J$  such that  $R_{uj_u^1} \geq R_{uj_u^2} \geq \dots \geq R_{uj_u^J}$  (Iyer et al., 2014). In this experiment, the sequence of objective functions to be optimized is  $\{\bar{f}_1, \bar{f}_2, \dots, \bar{f}_T\}$ . The stochastic gradient is obtained by the sampling method given in Eq. (1) with only one sample for each coordinate of the gradient. We set the constraint set to  $\{\mathbf{x} \in [0, 1]^J : \mathbf{1}^\top \mathbf{x} \leq 1\}$  and choose  $B = 5$ . We present the results in Fig. 2(a). Meta-FW w/ VR attains the smallest regret. The counterpart without variance reduction Meta-FW w/o VR is inferior to Meta-FW w/ VR in terms of the regret. OS-FW w/ VR outperforms OS-FW w/o NVR, which suggests that the variance reduction technique improves the performance of the algorithms.

**Joke Recommendations (Discrete)** In the second set experiments, we consider online maximization of discrete submodular functions. The problem set up is the same

as before, but instead of evaluating regret of the multilinear extensions, we round solutions using pipage rounding and evaluate the regret on the discrete submodular functions. We set the batch size  $B$  to 40 and we recommend 10 jokes for users. The results are illustrated in Fig. 2(b). We observe that Meta-FW w/ VR outperforms all other algorithms again. The projected algorithm OGA is second to Meta-FW w/ VR. Online Greedy appears only better than Regularized-OFW. The experiment result show that the continuous algorithms designed under the framework of the lifting method perform better than the discrete algorithms.

**Topic Summarization** We consider the problem of selecting news documents in order to maximize the probabilistic coverage of news topics (El-Arini et al., 2009; Yue & Guestrin, 2011). We applied the latent Dirichlet allocation to the corpus of Reuters-21578, Distribution 1.0, set the number of topics to 10, and extracted the topic distribution of each news document. We sample  $T$  batches of news documents from the corpus and denote them by  $B_1, B_2, \dots, B_T$ , where each batch contains 50 randomly sampled documents. For each batch  $B_i$ , we define the probabilistic coverage function as follows  $f_i(X) = \frac{1}{10} \sum_{j=1}^{10} [1 - \prod_{a \in X} (1 - p_a(j))]$ ,  $\forall X \subseteq B_i$ , where  $p_a(\cdot)$  is the topic distribution of news document  $a$ . Its multilinear extension is  $\bar{f}_i(\mathbf{x}) = \frac{1}{10} \sum_{j=1}^{10} [1 - \prod_{a \in X} (1 - p_a(j)x_a)]$ ,  $\forall \mathbf{x} \in [0, 1]^{50}$ , see (Iyer et al., 2014). The sequence of objective functions that the algorithms are expected to maximize is  $\bar{f}_1, \bar{f}_2, \dots, \bar{f}_T$ . As in the experiments on joke recommendations, the stochastic gradient is obtained by the sampling method given in Eq. (1) with only one sample for each coordinate of the gradient. The constraint set is  $\{\mathbf{x} \in [0, 1]^{50} : \mathbf{1}^\top \mathbf{x} \leq 45\}$ . We show the  $(1 - 1/e)$ -regret of the algorithms in Fig. 2(c). Again, Meta-FW w/ VR exhibits the lowest regret than any other algorithm. Its non-variance-reduced counterpart Meta-FW w/o VR is second to it. OS-FW w/ VR outperforms OS-FW w/o NVR, which confirms the improvement brought by the variance reduction technique.

### 5.2. Online Convex Minimization

The next two sets of experiments test the performance of the algorithms for online minimization of convex functions with stochastic gradient estimates. For these experiments, the regret is computed by obtaining the offline solutions with a Frank-Wolfe solver.

**Stochastic Cost Network Flow** The fourth set of experiments is a minimum stochastic cost flow in a directed network. A directed graph  $G = (V, E)$  with source  $s \in V$ , sink  $v \in V$ , and edge capacities  $c : E \rightarrow \mathbb{R}_+$  is known to the player. A flow is a function  $\mathbf{x} : \mathbb{R}_+^{|E|} \rightarrow \mathbb{R}_+$  that satisfies the capacities on each edge  $0 \leq \mathbf{x}(e) \leq c(e)$  and obeys the

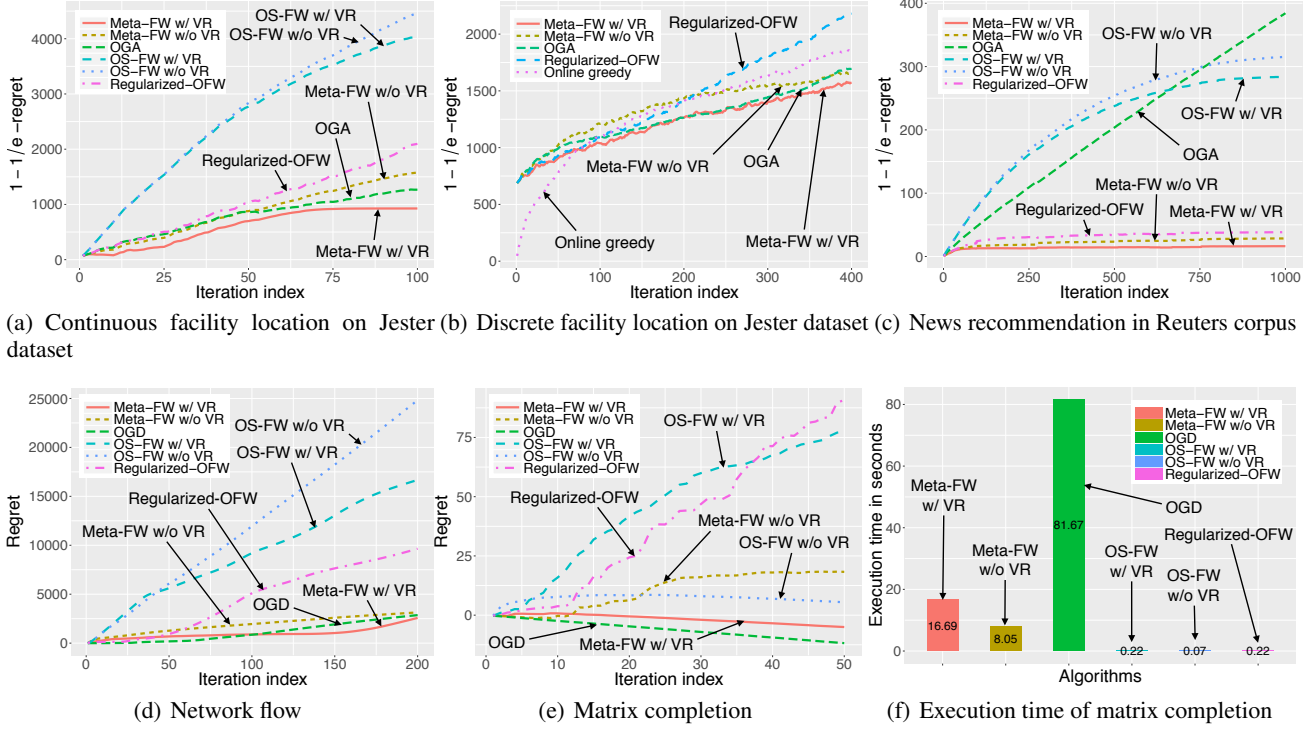


Figure 2. Figs. 2(a) to 2(c) shows the  $(1 - 1/e)$ -regret of online DR-submodular maximization algorithms. In Figs. 2(a) and 2(b), we show the regret for the continuous and discrete facility location objective functions on the Jester dataset, respectively. We show the results for the online news recommendation problem in the Reuters corpus in Fig. 2(c). The results for online convex minimization are illustrated in Figs. 2(d) to 2(f). In Fig. 2(d), we show the regret of the algorithms applied to the stochastic cost network flow problem. The results for the matrix completion problem are shown in Fig. 2(e) and the computational time is illustrated in Fig. 2(f).

conservation laws for all vertices  $z$ ,

$$\sum_{\{z,r\} \in E} \mathbf{x}(r) = \begin{cases} a & z = s \\ -a & z = v \\ 0 & \text{otherwise} \end{cases}$$

for some fixed  $a \geq 0$ . In each round  $t$ , a convex cost function on the flow  $f_t: \mathbb{R}^{|E|} \rightarrow \mathbb{R}_+$  is drawn from a distribution, unknown to the player. The goal is to minimize the stochastic regret of the flows chosen. Linear optimizations for this problem may be implemented as combinatorial network flow algorithms. We used the directed Zachary Karate network with 34 nodes and 78 arcs (Zachary, 1977). We set all edge capacities to 1 and cost functions are of the form  $f(\mathbf{x}) = \sum_{e \in E} w_e \mathbf{x}(e)^2$  where  $w_e \sim \text{Unif}[100, 120]$ . The results are presented in Fig. 2(d). Meta-FW w/ VR attains the lowest regret among all baselines. Again, the regret of Meta-FW w/o VR is larger than the variance-reduced Meta-FW w/ VR. Similarly, OS-FW w/ VR also outperforms OS-FW w/o NVR.

**Matrix Completion** In the online convex matrix completion problem, one would like to construct a low rank matrix  $X \in \mathbb{R}^{m \times n}$  that well-approximates a given matrix

$M \in \mathbb{R}^{m \times n}$  on observed entries  $OB \subseteq [m] \times [n]$ . The convex relaxation is  $\min_{\text{Trace}(X) \leq k} \sum_{(i,j) \in OB} (X_{i,j} - M_{i,j})^2$ . In the online setting, observed entries of the matrix arrive in  $T$  batches,  $OB_1, OB_2, \dots, OB_T$ , each of size  $B$ . In each round, we construct a low-rank matrix to minimize the total regret over the  $T$  rounds. Although projection involves a full singular value decomposition, linear optimization here is simply a calculation of the largest singular vectors of  $(X - M)_{OB}$ , see Chapter 7 of (Hazan et al., 2016). In our experiment,  $M$  is a rank 10 matrix with  $m = n = 50$ , and  $B = 100$ . We illustrate the results in Fig. 2(e) and the computational time is shown in Fig. 2(f). Meta-FW w/ VR is only second to OGD. However, OGD is the slowest algorithm due to the computationally expensive projection operations and its computational time is five times that of Meta-FW w/ VR. The non-variance-reduced Meta-FW w/o VR is inferior to Meta-FW w/ VR in terms of regret.

## Acknowledgments

AK was supported by AFOSR YIP (FA9550-18-1-0160). CH was supported in part by NSF GRFP (DGE1122492) and by ONR Award N00014-16-1-2374.



## References

- Agarwal, A., Hazan, E., Kale, S., and Schapire, R. E. Algorithms for portfolio management based on the newton method. In *Proceedings of the 23rd International Conference on Machine Learning*, ICML '06, pp. 9–16, 2006.
- Ageev, A. A. and Sviridenko, M. I. Pipe rounding: A new method of constructing algorithms with proven performance guarantee. *Journal of Combinatorial Optimization*, 8(3):307–328, 2004.
- Allen-Zhu, Z. and Hazan, E. Variance reduction for faster non-convex optimization. In *International Conference on Machine Learning*, pp. 699–707, 2016.
- Bach, F. Submodular functions: from discrete to continuous domains. *arXiv preprint arXiv:1511.00394*, 2015.
- Bian, A., Mirzasoleiman, B., Buhmann, J. M., and Krause, A. Guaranteed non-convex optimization: Submodular maximization over continuous domains. In *AISTATS*, February 2017.
- Calandriello, D., Lazaric, A., and Valko, M. Second-order kernel online convex optimization with adaptive sketching. In *International Conference on Machine Learning*, ICML '17, 2017.
- Calinescu, G., Chekuri, C., Pál, M., and Vondrák, J. Maximizing a monotone submodular function subject to a matroid constraint. *SIAM Journal on Computing*, 40(6):1740–1766, 2011.
- Chen, L., Hassani, H., and Karbasi, A. Online continuous submodular maximization. In *AISTATS*, pp. to appear, 2018.
- Cohen, A. and Hazan, T. Following the perturbed leader for online structured learning. In *Proceedings of the 32Nd International Conference on International Conference on Machine Learning - Volume 37*, ICML'15, pp. 1034–1042, 2015.
- El-Arini, K., Veda, G., Shahaf, D., and Guestrin, C. Turning down the noise in the blogosphere. In *SIGKDD*, pp. 289–298. ACM, 2009.
- Frank, M. and Wolfe, P. An algorithm for quadratic programming. *Naval Research Logistics (NRL)*, 3(1-2):95–110, 1956.
- Garber, D. and Hazan, E. A linearly convergent conditional gradient algorithm with applications to online and stochastic optimization. *arXiv preprint arXiv:1301.4666*, 2013.
- Goldberg, K., Roeder, T., Gupta, D., and Perkins, C. Eigentaste: A constant time collaborative filtering algorithm. *information retrieval*, 4(2):133–151, 2001.
- Golovin, D., Krause, A., and Streeter, M. Online submodular maximization under a matroid constraint with application to learning assignments. Technical report, arXiv, 2014.
- Hassani, H., Soltanolkotabi, M., and Karbasi, A. Gradient methods for submodular maximization. *arXiv preprint arXiv:1708.03949*, 2017.
- Hazan, E. and Kale, S. Projection-free online learning. In *ICML*, pp. 1843–1850, 2012.
- Hazan, E. and Luo, H. Variance-reduced and projection-free stochastic optimization. In *ICML*, pp. 1263–1271, 2016.
- Hazan, E. et al. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.
- Iyer, R., Jegelka, S., and Bilmes, J. Monotone closure of relaxed constraints in submodular optimization: Connections between minimization and maximization. In *Uncertainty in Artificial Intelligence (UAI)*, Quebec City, Quebec Canada, July 2014. AUAI.
- Johnson, R. and Zhang, T. Accelerating stochastic gradient descent using predictive variance reduction. In *NIPS*, pp. 315–323, 2013.
- Lafond, J., Wai, H.-T., and Moulines, E. On the online Frank-Wolfe algorithms for convex and non-convex optimizations. *arXiv preprint arXiv:1510.01171*, 2015.
- Mahdavi, M., Zhang, L., and Jin, R. Mixed optimization for smooth functions. In *NIPS*, pp. 674–682, 2013.
- Mokhtari, A., Hassani, H., and Karbasi, A. Conditional gradient method for stochastic submodular maximization: Closing the gap. In *AISTATS*, pp. 1886–1895, 2018a.
- Mokhtari, A., Hassani, H., and Karbasi, A. Stochastic conditional gradient methods: From convex minimization to submodular maximization. *arXiv preprint arXiv:1804.09554*, 2018b.
- Shalev-Shwartz, S. et al. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning*, 4(2):107–194, 2012.
- Streeter, M. and Golovin, D. An online algorithm for maximizing submodular functions. In *NIPS*, pp. 1577–1584, 2009.
- Vondrák, J. *Submodularity in combinatorial optimization*. PhD thesis, Charles University, 2007.
- Vondrák, J., Chekuri, C., and Zenklusen, R. Submodular function maximization via the multilinear relaxation and contention resolution schemes. In *STOC*, pp. 783–792. ACM, 2011.

- Wang, Y. and Boyd, S. Fast model predictive control using online optimization. *IFAC Proceedings Volumes*, 41(2): 6974 – 6979, 2008. ISSN 1474-6670. 17th IFAC World Congress.
- Wolsey, L. A. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4): 385–393, 1982.
- Xiao, L. Dual averaging methods for regularized stochastic learning and online optimization. *J. Mach. Learn. Res.*, 11:2543–2596, December 2010. ISSN 1532-4435.
- Yue, Y. and Guestrin, C. Linear submodular bandits and their application to diversified retrieval. In *NIPS*, pp. 2483–2491, 2011.
- Zachary, W. W. An information flow model for conflict and fission in small groups. *Journal of anthropological research*, 33(4):452–473, 1977.
- Zinkevich, M. Online convex programming and generalized infinitesimal gradient ascent. In *ICML*, pp. 928–936, 2003.