

Efficient Projection-Free Online Methods with Stochastic Recursive Gradient

Jiahao Xie,¹ Zebang Shen,^{*2} Chao Zhang,^{1,3} Boyu Wang,⁴ Hui Qian¹

¹College of Computer Science and Technology, Zhejiang University

²University of Pennsylvania ³Tencent AI Lab ⁴University of Western Ontario
 {xiejh, qianhui, zczju}@zju.edu.cn, zebang@seas.upenn.edu, bwang@csd.uwo.ca

Abstract

This paper focuses on projection-free methods for solving smooth Online Convex Optimization (OCO) problems. Existing projection-free methods either achieve suboptimal regret bounds or have high per-round computational costs. To fill this gap, two efficient projection-free online methods called ORGFW and MORGFW are proposed for solving stochastic and adversarial OCO problems, respectively. By employing a recursive gradient estimator, our methods achieve optimal regret bounds (up to a logarithmic factor) while possessing low per-round computational costs. Experimental results demonstrate the efficiency of the proposed methods compared to state-of-the-arts.

1 Introduction

We consider the following smooth Online Convex Optimization (OCO) problem. In each round $t = 1, \dots, T$, a learner chooses a decision \mathbf{x}_t from a compact convex set $\mathcal{C} \subseteq \mathbb{R}^d$. Then a smooth convex loss function f_t is revealed and the learner suffers the loss $f_t(\mathbf{x}_t)$. We consider both adversarial and stochastic settings. In the *adversarial setting*, the sequence of functions $\{f_t\}_{t=1}^T$ can be arbitrary (possibly adversarial), while in the *stochastic setting*, the loss functions are sampled i.i.d. from some fixed distribution $f_t \sim \mathcal{D}$. The target of the learner is to produce a sequence of decisions $\{\mathbf{x}_t\}$ that minimizes the regret, which is the cumulative loss suffered by the learner compared to that of the best fixed decision in hindsight, i.e.,

$$\sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}^*),$$

where $\mathbf{x}^* \in \operatorname{argmin}_{\mathbf{x} \in \mathcal{C}} \sum_{t=1}^T f_t(\mathbf{x})$ in the adversarial setting and $\mathbf{x}^* \in \operatorname{argmin}_{\mathbf{x} \in \mathcal{C}} \mathbb{E}_{f_t \sim \mathcal{D}} [f_t(\mathbf{x})]$ in the stochastic setting, respectively.

This model captures a wide range of real-world applications in which data points arrive sequentially, e.g., online recommendation systems, online spam email filtering, online prediction in financial markets, online portfolio selection, to name a few (Hazan 2016; Hoi et al. 2018; Cesa-Bianchi and Lugosi 2006; Agarwal et al. 2006).

Existing methods for solving OCO problems can be divided into two categories: (i) projection-based methods (Zinkevich 2003; Shalev-Shwartz and Singer 2007; Xiao 2010; Duchi, Hazan, and Singer 2011; Cutkosky and Boahen 2017), and (ii) projection-free methods (Hazan and Kale 2012; Hazan 2016; Lafond, Wai, and Moulines 2015; Chen et al. 2018). For high-dimensional applications with complicated constraints (e.g., low rank matrix completion (Chandrasekaran et al. 2009), network routing (Hazan 2016), and structural SVMs (Lacoste-Julien et al. 2013)), the projection operation can be computationally expensive or even intractable, rendering projection-based methods impractical. In contrast, projection-free methods such as Frank-Wolfe-type methods (Hazan and Kale 2012; Hazan 2016) only require to solve linear optimization problems over the constraint set, which is usually simpler than projection. Thus projection-free methods have attracted considerable attention in recent years.

However, existing projection-free online methods suffer from a trade-off between regret and computational complexity. The seminal work, the Online Frank-Wolfe (OFW) method (Hazan and Kale 2012), achieves an $\mathcal{O}(T^{3/4})$ regret in the adversarial setting. Besides, in the stochastic setting, OFW achieves a nearly optimal $\tilde{\mathcal{O}}(\sqrt{T})$ regret¹ with high probability. For both settings, the per-round computational cost of OFW ($\mathcal{O}(T)$ on average) is considerably high. The Regularized OFW method (Hazan 2016) improves the per-round computational cost to $\mathcal{O}(1)$ in the adversarial setting while remaining the same $\mathcal{O}(T^{3/4})$ regret bound as OFW. Additionally, OFW and Regularized OFW require to access exact gradients of f_t 's, which can be computationally prohibitive in online applications with large-scale streaming data where a large batch of data arrives in each round (Dekel et al. 2012). To tackle this problem, Chen et al. (2018) propose two methods called Meta-Frank-Wolfe and OSFW, which use stochastic gradient estimates, for the adversarial and stochastic settings, respectively. Meta-Frank-Wolfe requires $\mathcal{O}(T^{3/2})$ stochastic gradient evalua-

^{*}Corresponding author.
 Copyright © 2020, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

¹It is known that the optimal regret bound for general OCO problems is $\mathcal{O}(\sqrt{T})$ (see, e.g., (Hazan 2016)). We call $\tilde{\mathcal{O}}(\sqrt{T})$ nearly optimal, where $\tilde{\mathcal{O}}()$ suppresses a poly-logarithmic factor.

Table 1: Comparison of projection-free online methods. The fourth column shows the per-round computational cost in average. The fifth column indicates whether a method uses stochastic gradients or exact (full) gradients of f_t 's.

Algorithm	Setting	Regret	Per-round cost	Stochastic gradients	Guarantee
OFW	adversarial, cvx.	$\mathcal{O}(T^{3/4})$	$\mathcal{O}(T)$	No	deterministic
Regularized OFW	adversarial, cvx., smooth	$\mathcal{O}(T^{3/4})$	$\mathcal{O}(1)$	No	deterministic
Meta-Frank-Wolfe	adversarial, cvx., smooth	$\mathcal{O}(\sqrt{T})$	$\mathcal{O}(T^{3/2})$	Yes	in expectation
MORGFW (this work)	adversarial, cvx., smooth	$\tilde{\mathcal{O}}(\sqrt{T})$	$\mathcal{O}(T)$	Yes	w.h.p.
OFW	stoch., cvx., smooth	$\tilde{\mathcal{O}}(\sqrt{T})$	$\mathcal{O}(T)$	No	w.h.p.
OSFW	stoch., cvx., smooth	$\mathcal{O}(T^{2/3})$	$\mathcal{O}(1)$	Yes	in expectation
ORGFW (this work)	stoch., cvx., smooth	$\tilde{\mathcal{O}}(\sqrt{T})$	$\mathcal{O}(1)$	Yes	w.h.p.

OFW (Hazan and Kale 2012), Regularized OFW (Hazan 2016), OSFW and Meta-Frank-Wolfe (Chen et al. 2018)

tions in each round, although it achieves the optimal regret bound ($\mathcal{O}(\sqrt{T})$ in expectation) in the adversarial setting. OSFW achieves a suboptimal $\mathcal{O}(T^{2/3})$ regret bound (in expectation) in the stochastic setting. To the best of our knowledge, none of existing projection-free online methods has both the optimal (or nearly optimal) regret bound and a low computational cost at the same time.

To bridge this gap, we propose two novel projection-free methods, Online stochastic Recursive Gradient-based Frank-Wolfe (ORGFW) and Meta-ORGFW (MORGFW), for OCO problems in the stochastic and adversarial settings, respectively. Both methods achieve nearly optimal regret bounds with high probability while having low computational costs. To achieve this goal, we utilize a recursive variance reduction technique to reduce noise in stochastic gradients without bringing much extra computation. Then, we develop a new analysis technique based on martingale concentration inequalities to bound the gradient approximation error to a desired accuracy, which allows us to derive the optimal regret bound. Note that a similar variance reduction technique has been adopted by (Cutkosky and Orabona 2019) for solving unconstrained nonconvex stochastic optimization. While they focus on finding an approximate stationary point, we aim at producing a sequence of decisions that has low regret. Our contributions are listed as follows.

- We show that ORGFW achieves a nearly optimal $\tilde{\mathcal{O}}(\sqrt{T})$ regret bound for smooth OCO problems in the stochastic setting. To the best of our knowledge, this is the first projection-free online method that has both a nearly optimal regret bound and an $\mathcal{O}(1)$ per-round computational cost in such setting.
- For smooth OCO problems in the adversarial setting, MORGFW achieves a nearly optimal $\tilde{\mathcal{O}}(\sqrt{T})$ regret bound. This method only requires $\mathcal{O}(T)$ stochastic gradient evaluations in each round, improving upon the $\mathcal{O}(T^{3/2})$ cost of Meta-Frank-Wolfe (Chen et al. 2018).
- Compared to the regret bounds in (Chen et al. 2018), which hold in expectation, our results hold with high probability and therefore rule out the possibility that the regret has high variance. To establish high-probability regret bounds, we propose a new analysis technique by

utilizing a martingale concentration inequality to bound the gradient approximation error in high probability. This technique can be of independent interests for establishing high-probability bounds for other online methods.

A summary of our results and previous ones is provided in Table 1. In addition to regret bounds for online learning, we also prove the convergence of ORGFW for solving both convex and nonconvex stochastic optimization problems. Our experimental results demonstrate the advantages of the proposed methods over existing projection-free methods.

2 Related Work

Online projection-free methods. The classical Frank-Wolfe (FW) method (a.k.a. conditional gradient descent) is introduced by (Frank and Wolfe 1956) for solving offline optimization problems. Starting with (Hazan 2008), Frank-Wolfe has regained a lot of popularity because it has the advantages of projection-free, norm-free, and sparse iterates (Bubeck and others 2015). Hazan and Kale (2012) propose the first online Frank-Wolfe method called OFW, which requires to evaluate the gradient of the cumulative loss function $\sum_{\tau=1}^t f_t$ at the t -th iteration and thus has a high computational cost in general. Lafond, Wai, and Moulines (2015) propose Online Away-step Frank-Wolfe (OAW), which incorporates the away step technique (Guélat and Marcotte 1986) into OFW. They show that both OFW and OAW achieve logarithmic regrets for OCO problems in the stochastic setting if the loss functions are strongly convex and smooth and the constraint set satisfies additional assumptions. Besides, they also prove that these two methods find a stationary point of a nonconvex stochastic optimization problem. Hazan (2016) proposes a method called Regularized OFW, which leverages a regularization technique and only requires to evaluate one gradient of f_t at the t -th iteration. Zhang et al. (2017) extends Regularized OFW to distributed online learning for solving OCO problems with large-scale streaming data. Another direction to solve large-scale OCO problems is to reduce the computational cost by using stochastic gradient estimates instead of exact gradients of f_t 's, which is studied in (Chen et al. 2018).

Variance reduction. Variance Reduction (VR) techniques are originally proposed to reduce variance in gra-

gradient estimation for stochastic gradient methods (Johnson and Zhang 2013; Defazio, Bach, and Lacoste-Julien 2014; Nguyen et al. 2017; Fang et al. 2018; Zhou, Xu, and Gu 2018; Nguyen, Scheinberg, and Takáč 2018). Several stochastic projection-free VR methods have been proposed for solving offline optimization problems (Hazan and Luo 2016; Reddi et al. 2016; Mokhtari, Hassani, and Karbasi 2018; Shen et al. 2019; Yurtsever, Sra, and Cevher 2019). These VR methods cannot directly apply to OCO problems since OCO problems are fundamentally different from offline ones. Recently, Chen et al. (2018) propose the first projection-free VR method for OCO.

3 Notation and Preliminaries

Notation. We use bold lowercase symbols (e.g., \mathbf{x}) to denote vectors and bold uppercase symbols (e.g., \mathbf{A}) to denote matrices. The entry in the i -th row and j -th column of a matrix \mathbf{A} is denoted by $[\mathbf{A}]_{ij}$. Throughout this paper, we use $\|\mathbf{x}\|$ to denote the standard Euclidean norm of a vector \mathbf{x} .

We consider both the *adversarial setting* and the *stochastic setting* of online convex optimization problems. For these two settings, the definitions of regret are slightly different. In the adversarial setting, the regret is defined as

$$\mathcal{R}_T := \sum_{t=1}^T f_t(\mathbf{x}_t) - \min_{\mathbf{x} \in \mathcal{C}} \sum_{t=1}^T f_t(\mathbf{x}). \quad (1)$$

In the stochastic setting, the regret is defined as

$$S\mathcal{R}_T := \sum_{t=1}^T (f_t(\mathbf{x}_t) - f_t(\mathbf{x}^*)), \quad (2)$$

where $\mathbf{x}^* \in \operatorname{argmin}_{\mathbf{x} \in \mathcal{C}} \bar{f}(\mathbf{x}) := \mathbb{E}_{f_t \sim \mathcal{D}}[f_t(\mathbf{x})]$. We note that the OCO problem in the stochastic setting is closely related to but different from the *stochastic optimization* problem (Birge and Louveaux 1997). In OCO, the goal is to produce a sequence of decision variables that has low regret and the learner must properly respond to the environment (i.e., updating the decision variable) as soon as new data arrive (Dekel et al. 2012). In stochastic optimization, however, we aim to find an approximate minimizer of the loss function and the performance of a method is measured by the convergence rate. Compared to the OCO problem, stochastic optimization focuses on the quality of the final output of a method instead of the sequence of iterates produced over the course of optimization.

4 Online Stochastic Recursive Gradient-Based Frank-Wolfe

In this section, we present our projection-free methods for solving OCO problems. We first introduce the Online stochastic Recursive Gradient-based Frank-Wolfe (ORGFW) method, which uses a stochastic recursive gradient estimator, for the stochastic setting. Based on ORGFW, we introduce the Meta-ORGFW (MORGFW) method for the more challenging adversarial setting.

Algorithm 1: ORGFW

Input : parameters $\{\rho_t\}_{t=1}^T$, $\{\eta_t\}_{t=1}^T$, and initial point $\mathbf{x}_0 = \mathbf{x}_1 \in \mathcal{C}$

- 1 **for** $t = 1, 2, \dots, T$ **do**
- 2 Play \mathbf{x}_t , then receive $f_t(\mathbf{x}_t)$ and stochastic gradients $\nabla F_t(\mathbf{x}_{t-1}, \xi_t)$ and $\nabla F_t(\mathbf{x}_t, \xi_t)$;
- 3 **if** $t = 1$ **then**
- 4 $\mathbf{d}_t \leftarrow \nabla F_t(\mathbf{x}_t, \xi_t)$;
- 5 **else**
- 6 $\mathbf{d}_t \leftarrow \nabla F_t(\mathbf{x}_t, \xi_t) + (1 - \rho_t)(\mathbf{d}_{t-1} - \nabla F_t(\mathbf{x}_{t-1}, \xi_t))$;
- 7 **end**
- 8 $\mathbf{v}_t \leftarrow \operatorname{argmin}_{\mathbf{v} \in \mathcal{C}} \langle \mathbf{d}_t, \mathbf{v} \rangle$;
- 9 $\mathbf{x}_{t+1} \leftarrow \mathbf{x}_t + \eta_t(\mathbf{v}_t - \mathbf{x}_t)$;
- 10 **end**

4.1 Algorithm in the Stochastic Setting

Now we present ORGFW, which is detailed in Algorithm 1. In each round $t = 1, \dots, T$, ORGFW plays \mathbf{x}_t and receives the loss $f_t(\mathbf{x}_t)$ as well as stochastic gradients $\nabla F_t(\mathbf{x}_{t-1}, \xi_t)$ and $\nabla F_t(\mathbf{x}_t, \xi_t)$, where ξ_t is a random variable following some distribution \mathcal{P}_t such that $\mathbb{E}_{\xi_t \sim \mathcal{P}_t}[\nabla F_t(\mathbf{x}, \xi_t)] = \nabla f_t(\mathbf{x})$. For example, if f_t has a finite-sum structure of the form $f_t(\mathbf{x}) = \frac{1}{n_t} \sum_{i=1}^{n_t} f_{t,i}(\mathbf{x})$, which occurs in online problems with large-scale streaming data, one can let \mathcal{P}_t be the uniform distribution over $\{1, \dots, n_t\}$ and $\nabla F_t(\cdot, \xi_t) = \nabla f_{t,\xi_t}(\cdot)$. In line 6 of ORGFW, we estimate the gradient of \bar{f} using a stochastic recursive estimator

$$\mathbf{d}_t = \nabla F_t(\mathbf{x}_t, \xi_t) + (1 - \rho_t)(\mathbf{d}_{t-1} - \nabla F_t(\mathbf{x}_{t-1}, \xi_t)) \quad (3)$$

where $\mathbf{d}_1 = \nabla F_1(\mathbf{x}_1, \xi_1)$ and ρ_t is a parameter to be determined later. If the exact gradient $\nabla f_t(\mathbf{x})$ can be efficiently computed, one can directly replace $\nabla F_t(\mathbf{x}, \xi_t)$ with $\nabla f_t(\mathbf{x})$. After updating \mathbf{d}_t , ORGFW finds a solution \mathbf{v}_t to the linear optimization problem $\operatorname{argmin}_{\mathbf{v} \in \mathcal{C}} \langle \mathbf{d}_t, \mathbf{v} \rangle$ and updates \mathbf{x}_{t+1} along the direction $\mathbf{v}_t - \mathbf{x}_t$, where the step size η_t will be determined later.

The recursive estimator (3) is inspired by (Cutkosky and Orabona 2019) in which a similar estimator is devised for solving unconstrained nonconvex stochastic optimization problems. One difference between (3) and the estimator in (Cutkosky and Orabona 2019) is that in (3), the stochastic gradients $\nabla F_t(\cdot, \xi_t)$ in different rounds are sampled from different distributions, while in their estimator, all the stochasticity comes from the same distribution \mathcal{P} . More importantly, our analysis is fundamentally different from theirs. In our analysis, we explicitly show that the gradient approximation error converges to zero at a sublinear rate w.h.p., which is critical to analyzing regret bounds in high probability. In contrast, Cutkosky and Orabona (2019) do not explicitly analyze the convergence property of the approximation error but instead construct a Lyapunov function to derive convergence analysis of their method.

Algorithm 2: MORGFW

Input : Parameters $T, K, \{\eta_k\}_{k=1}^K, \{\rho_k\}_{k=1}^K$, base algorithms $\mathcal{E}^{(1)}, \dots, \mathcal{E}^{(K)}$, initial point $\mathbf{x}_1 \in \mathcal{C}$

- 1 **for** $t = 1, 2, \dots, T$ **do**
- 2 Initialize $\mathbf{x}_t^{(1)} = \mathbf{x}_1$;
- 3 **for** $k = 1, 2, \dots, K$ **do**
- 4 $\mathbf{v}_t^{(k)} \leftarrow$ output of $\mathcal{E}^{(k)}$ in round $t - 1$;
- 5 $\mathbf{x}_t^{(k+1)} \leftarrow (1 - \eta_k)\mathbf{x}_t^{(k)} + \eta_k\mathbf{v}_t^{(k)}$;
- 6 **end**
- 7 Play $\mathbf{x}_t = \mathbf{x}_t^{(K+1)}$ and receives $f_t(\mathbf{x}_t)$ and stochastic gradient oracle $\nabla F_t(\cdot, \cdot)$;
- 8 **for** $k = 1, 2, \dots, K$ **do**
- 9 Sample $\xi_t^{(k)} \sim \mathcal{P}_t$;
- 10 **if** $k = 1$ **then**
- 11 $\mathbf{d}_t^{(k)} \leftarrow \nabla F_t(\mathbf{x}_t^{(k)}, \xi_t^{(k)})$;
- 12 **else**
- 13 $\mathbf{d}_t^{(k)} \leftarrow \nabla F_t(\mathbf{x}_t^{(k)}, \xi_t^{(k)}) + (1 - \rho^k)(\mathbf{d}_t^{(k-1)} - \nabla F_t(\mathbf{x}_t^{(k-1)}, \xi_t^{(k)}))$;
- 14 **end**
- 15 Feedback $\langle \mathbf{v}_t^{(k)}, \mathbf{d}_t^{(k)} \rangle$ to $\mathcal{E}^{(k)}$;
- 16 **end**
- 17 **end**

4.2 Algorithm in the Adversarial Setting

Inspired by the Meta-Frank-Wolfe method (Chen et al. 2018), we use the recursive estimator (3) to develop a meta algorithm called MORGFW for OCO problems in the adversarial setting. Note that MORGFW is a general framework that relies on the outputs of base Online Linear Optimization (OLO)² algorithms. The MORGFW method is detailed in Algorithm 2. In each round $t = 1, \dots, T$, it simulates a K -step Frank-Wolfe subroutine using stochastic gradients of f_t and OLO algorithms $\mathcal{E}^{(1)}, \dots, \mathcal{E}^{(K)}$. We refer to $\mathcal{E}^{(k)}$ for $k \in \{1, \dots, K\}$ as the base algorithms. Typical algorithms for OLO include Follow the Perturbed Leader (Kalai and Vempala 2005), Online Gradient Descent (Zinkevich 2003), Regularized-Follow-The-Leader (Shalev-Shwartz and Singer 2007), etc. From line 2 to line 6 in MORGFW, we sequentially take K Frank-Wolfe-type update steps in which the update direction $\mathbf{v}_t^{(k)}$ is produced by the base algorithm $\mathcal{E}^{(k)}$. We then take the final iterate $\mathbf{x}_t^{(K+1)}$ as the prediction in the t -th round and receive the loss function f_t as well as the stochastic gradient oracle. From line 8 to line 16 in MORGFW, we sequentially compute $\mathbf{d}_t^{(k)}$ using the recursive estimator to approximate $\nabla f_t(\mathbf{x}_t^{(k)})$ for $k = 1, \dots, K$ and feedback the linear loss $\langle \mathbf{d}_t^{(k)}, \mathbf{v}_t^{(k)} \rangle$ to $\mathcal{E}^{(k)}$. If the exact gradient ∇f_t can be efficiently computed, one can directly replace \mathbf{d}_t with $\nabla f_t(\mathbf{x}_t)$.

²Online linear optimization is a special case of online convex optimization in which the loss functions are linear.

5 Regret Analysis

In this section, we analyze the regret bounds of the proposed methods. As a byproduct, we also derive convergence guarantee of ORGFW for convex and nonconvex stochastic optimization problems, respectively. All missing proofs are left to the long version of this paper due to the limit of space. To begin with, we make the following two common assumptions on the constraint set \mathcal{C} and stochastic gradients of f_t 's, respectively.

Assumption 1. *The compact convex set $\mathcal{C} \subseteq \mathbb{R}^d$ has diameter D , i.e., $\forall \mathbf{x}, \mathbf{y} \in \mathcal{C}$,*

$$\|\mathbf{x} - \mathbf{y}\| \leq D.$$

Assumption 2. *The stochastic gradient $\nabla F_t(\mathbf{x}, \xi_t)$ is unbiased (i.e., $\mathbb{E}_{\xi_t}[\nabla F_t(\mathbf{x}, \xi_t)] = \nabla f_t(\mathbf{x})$) and is L -Lipschitz continuous over the constraint set \mathcal{C} , i.e.,*

$$\|\nabla F_t(\mathbf{x}, \xi_t) - \nabla F_t(\mathbf{y}, \xi_t)\| \leq L\|\mathbf{x} - \mathbf{y}\|, \forall \mathbf{x}, \mathbf{y} \in \mathcal{C}.$$

Assumption 2 immediately implies that f_t is differentiable and has L -Lipschitz-continuous gradients.

5.1 Analysis of ORGFW

In the stochastic online setting, we denote the expected loss function as $\bar{f} = \mathbb{E}_{f_t \sim \mathcal{D}}[f_t]$. In order to obtain high probabilistic results, the following common assumption is required.

Assumption 3. *We assume the following*

a The distance between the stochastic gradient $\nabla F_t(\mathbf{x}, \xi_t)$ and the exact gradient is bounded over the constraint set \mathcal{C} , i.e., for any $\mathbf{x} \in \mathcal{C}$, $t \in \{1, \dots, T\}$, there exists $\sigma^2 < \infty$ such that with probability 1,

$$\|\nabla F_t(\mathbf{x}, \xi_t) - \nabla \bar{f}(\mathbf{x})\|^2 \leq \sigma^2.$$

b The difference of $f_t(\mathbf{x})$ and $\bar{f}(\mathbf{x})$ is bounded over the constraint set \mathcal{C} , i.e., $\forall \mathbf{x} \in \mathcal{C}$, $t \in \{1, \dots, T\}$, there exists $M^2 < \infty$ such that with probability 1,

$$|f_t(\mathbf{x}) - \bar{f}(\mathbf{x})|^2 \leq M^2.$$

In our proofs, we develop a new analysis technique to show that the norm of the gradient estimation error $\epsilon_t := \mathbf{d}_t - \nabla \bar{f}(\mathbf{x}_t)$ converges to zero rapidly w.h.p. The main idea of our analysis technique is summarized in the following. First, we reformulate ϵ_t as the sum of a martingale difference sequence $\{\zeta_{t,\tau}\}_{\tau=1}^t$ w.r.t. a filtration $\{\mathcal{F}_\tau\}_{\tau=0}^t$, i.e., $\epsilon_t = \sum_{\tau=1}^t \zeta_{t,\tau}$, where $\mathbb{E}[\zeta_{t,\tau} | \mathcal{F}_{\tau-1}] = \mathbf{0}$ and $\mathcal{F}_{\tau-1}$ is the σ -field generated by $\{f_1, \xi_1, \dots, f_{\tau-1}, \xi_{\tau-1}\}$. By showing that $\|\zeta_{t,\tau}\| \leq c_{t,\tau}$ for some constant $c_{t,\tau}$, one can relate the error $\|\epsilon_t\|$ to the quantity $q_t := \sum_{\tau=1}^t c_{t,\tau}^2$ via an Azuma-Hoeffding-type concentration inequality (Pinelis 1994, Theorem 3.5). With carefully chosen $\{\rho_t\}_{t=1}^T$ and $\{\eta_t\}_{t=1}^T$, the quantity q_t can be shown to converge to zero at a sublinear rate by induction. As a result, $\|\epsilon_t\|$ converges to zero at a sublinear rate w.h.p. as stated in the following lemma.

Lemma 1. *Consider ORGFW with $\eta_t = \rho_t = 1/(t+1)^\alpha$ for some $\alpha \in (0, 1]$. If Assumptions 1, 2, and 3.a are satisfied, for any $t \geq 1$ and $\delta_0 \in (0, 1)$, we have w.p. at least $1 - \delta_0$,*

$$\|\epsilon_t\| \leq 2(2LD + \frac{3^\alpha \sigma}{3^\alpha - 1})(t+1)^{-\alpha/2} \sqrt{2 \log(4/\delta_0)}.$$

Lemma 1 shows that the gradient approximation error $\|\epsilon_t\|$ converges to zero at a fast sublinear rate $\tilde{O}(1/t^{\alpha/2})$ w.h.p. if $\eta_t = \rho_t = 1/(t+1)^\alpha$ for any $\alpha \in (0, 1]$. This result is critical to the regret analysis of our methods.

Now we are ready to present the first main theorem.

Theorem 1. (Regret Bound w.h.p. in the Stochastic Setting) Consider ORGFW with $\eta_t = \rho_t = 1/(t+1)$. If \bar{f} is convex and Assumptions 1-3 are satisfied, then w.p. at least $1 - \delta$ for any $\delta \in (0, 1)$,

$$\begin{aligned} \mathcal{SR}_T \leq & (\log T + 1)(\bar{f}(\mathbf{x}_1) - \bar{f}(\mathbf{x}^*)) + \frac{LD^2(\log T + 1)^2}{2} \\ & + (16LD^2 + 16\sigma D + 4M)\sqrt{2T\log(8T/\delta)}, \end{aligned}$$

where \mathcal{SR}_T is defined in (2).

Theorem 1 shows that ORGFW achieves a nearly optimal $\tilde{O}(\sqrt{T})$ regret bound w.h.p. for OCO problems in the stochastic setting under mild assumptions. As a byproduct, we provide convergence guarantee of ORGFW for convex stochastic optimization in the following corollary.

Corollary 1. (Convergence rate for Convex Stochastic Optimization) Assume that \bar{f} is convex and Assumptions 1, 2, and 3.a are satisfied. If we run ORGFW with $\rho_t = \eta_t = 1/(t+1)$ and let $\hat{\mathbf{x}} = \frac{1}{T} \sum_{t=1}^T \mathbf{x}_t$, we have, with probability at least $1 - \delta$ for any $\delta \in (0, 1)$,

$$\begin{aligned} \bar{f}(\hat{\mathbf{x}}) - \bar{f}(\mathbf{x}^*) \leq & \frac{\log T + 1}{T} (\bar{f}(\mathbf{x}_1) - \bar{f}(\mathbf{x}^*)) \\ & + \frac{LD^2(\log T + 1)^2}{2T} + 16(LD^2 + \sigma D) \frac{\sqrt{2\log(4T/\delta)}}{\sqrt{T}}. \end{aligned}$$

Corollary 1 shows that ORGFW achieves a convergence rate of $\tilde{O}(1/\sqrt{T})$ w.h.p. for convex stochastic optimization problems. In other words, ORGFW needs $\tilde{O}(1/\epsilon^2)$ stochastic gradient evaluations to find a solution $\hat{\mathbf{x}}$ such that $\bar{f}(\hat{\mathbf{x}}) - \bar{f}(\mathbf{x}^*) \leq \epsilon$, which matches the state-of-the-art result (Lan and Zhou 2016; Yurtsever, Sra, and Cevher 2019).

Similarly, one can also prove that ORGFW finds an approximate stationary point of a nonconvex stochastic optimization problem. A point $\mathbf{x} \in \mathcal{C}$ is called an ϵ -approximate stationary point if it satisfies the condition

$$\mathcal{G}(\mathbf{x}) := \max_{\mathbf{u} \in \mathcal{C}} \langle \nabla \bar{f}(\mathbf{x}), \mathbf{x} - \mathbf{u} \rangle \leq \epsilon, \quad (4)$$

where the non-negative quantity $\mathcal{G}(\mathbf{x})$ is known as the Frank-Wolfe gap. The following proposition establishes the convergence rate of ORGFW to a stationary point.

Proposition 1. (Convergence rate for Nonconvex Stochastic Optimization) Assume that Assumptions 1, 2, and 3.a are satisfied. If we run ORGFW with $\rho_t = \eta_t = 1/(t+1)^{2/3}$, we have w.p. at least $1 - \delta$,

$$\begin{aligned} \min_{1 \leq t \leq T} \mathcal{G}(\mathbf{x}_t) \leq & \frac{2(\bar{f}(\mathbf{x}_1) - \bar{f}(\mathbf{x}^*))}{T^{1/3}} + \frac{4\log(T+1)}{T^{1/3}} \left(LD^2 \right. \\ & \left. + (2LD^2 + 3\sigma^2)\sqrt{2\log(4T/\delta)} \right). \end{aligned}$$

Proposition 1 shows that ORGFW converges to a stationary point at a rate of $\tilde{O}(1/T^{1/3})$. In other words, ORGFW finds an ϵ -approximate stationary point in $\tilde{O}(1/\epsilon^3)$ stochastic gradient evaluations. This result outperforms the $\tilde{O}(1/\epsilon^4)$ bound of OFW and OAW (Lafond, Wai, and Moulines 2015) and matches the state-of-art result (Shen et al. 2019; Yurtsever, Sra, and Cevher 2019).

5.2 Analysis of MORGFW

In the adversarial online setting, we make the following assumption which is analogous to Assumption 3.a.

Assumption 4. The distance between the stochastic gradient $\nabla F_t(\mathbf{x}, \xi_t)$ and the exact gradient $f_t(\mathbf{x})$ is bounded over the constraint set \mathcal{C} (with probability 1), i.e., $\forall \mathbf{x} \in \mathcal{C}$, $t \in \{1, \dots, T\}$, there exists $\hat{\sigma}^2 < \infty$ such that

$$\|\nabla F_t(\mathbf{x}, \xi_t) - \nabla f_t(\mathbf{x})\|^2 \leq \hat{\sigma}^2.$$

In the following theorem, we establish the regret bound of MORGFW in the adversarial setting.

Theorem 2. (Regret Bound w.h.p. in the Adversarial Setting) Consider MORGFW with $K = T$, $\rho_k = \eta_k = 1/(k+1)$. Suppose that each of the base algorithms $\mathcal{E}^{(1)}, \dots, \mathcal{E}^{(K)}$ has a regret $\mathcal{R}_T^\mathcal{E}$. If each f_t is convex and Assumptions 1, 2, and 4 are satisfied, then w.p. at least $1 - \delta$,

$$\begin{aligned} \mathcal{R}_T \leq & 16(LD^2 + \hat{\sigma}D)\sqrt{2T\log(4T^2/\delta)} \\ & + 2LD^2\log(T+1) + Q + \mathcal{R}_T^\mathcal{E}, \end{aligned}$$

where \mathcal{R}_T is defined in (1), $Q = \max_{1 \leq t \leq T} \{f_t(\mathbf{x}_1) - f_t(\mathbf{x}^*)\}$, and $\mathbf{x}^* \in \operatorname{argmin}_{\mathbf{x} \in \mathcal{C}} \sum_{t=1}^T f_t(\mathbf{x})$.

Theorem 2 shows that the regret of MORGFW is bounded from above by $\tilde{O}(\sqrt{T}) + \mathcal{R}_T^\mathcal{E}$ w.h.p., where $\mathcal{R}_T^\mathcal{E}$ is the regret of the base algorithms $\mathcal{E}^{(1)}, \dots, \mathcal{E}^{(K)}$. It remains to choose proper base algorithms for OLO. A suitable choice is Follow the Perturbed Leader which is a projection-free method with $\mathcal{R}_T^\mathcal{E} = \mathcal{O}(\sqrt{T})$ for OLO (Kalai and Vempala 2005; Cohen and Hazan 2015). Thus, by choosing Follow the Perturbed Leader as the base algorithm $\mathcal{E}^{(k)}$ in MORGFW, we achieve a nearly optimal $\tilde{O}(\sqrt{T})$ regret.

We remark that in Theorem 2, the parameter K is dependent on the time horizon T . Thus, MORGFW requires prior knowledge of the time horizon. Nevertheless, this issue can be easily solved by the doubling trick (Shalev-Shwartz 2012, Section 2.3.1). Indicated by (Shalev-Shwartz 2012), the regret bound only increases by a constant multiplicative factor if we adopt this trick.

6 Experiments

To validate the theoretical results in the previous section, we first conduct numerical experiments on an OCO problem, i.e., online multiclass logistic regression. To further show the efficiency of the proposed methods, we also conduct experiments on an offline nonconvex optimization problem – training a constrained one-hidden-layer neural network.³ We

³Source code: <https://github.com/xjjiajiahao/ORGFW>

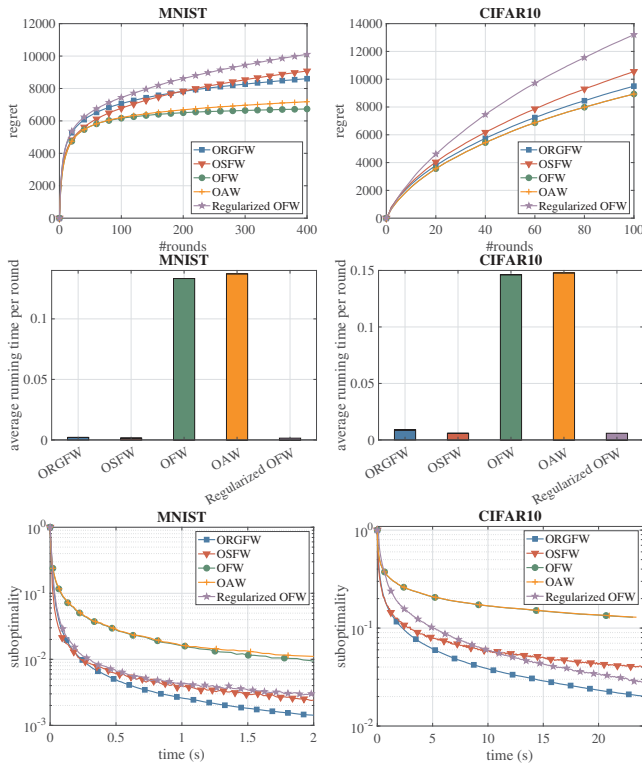


Figure 1: Results on the online multiclass logistic regression task in the stochastic setting (left: MNIST, right: CIFAR10). The left column shows the regret in the stochastic setting versus the number of rounds. The right column shows the suboptimality $(f(\mathbf{W}_t) - f(\mathbf{W}^*)) / (f(\mathbf{W}_1) - f(\mathbf{W}^*))$, where $\mathbf{W}^* \in \arg\min_{\mathbf{W} \in \mathcal{C}} f(\mathbf{W})$ and \mathbf{W}_1 is the initial point.

Table 2: Summary of the multiclass datasets.

Dataset	#features	#instances	#classes
MNIST	784	60,000	10
CIFAR10	3072	50,000	10

use two well-known multiclass datasets: MNIST⁴ and CIFAR10⁵. Detailed information of these datasets are listed in Table 2. For all compared methods, we choose hyperparameters via grid search and simply set the initial point to $\mathbf{0}$. Besides, we repeat the random methods for 6 trails and report the average result.

6.1 Online Multiclass Logistic Regression

In the first experiment, we consider an OCO problem – online multiclass logistic regression (Zhang et al. 2019). In each round $t = 1, \dots, T$, we receive a subset \mathcal{B}_t of data points with $|\mathcal{B}_t| = B$, where each data point $i \in \mathcal{B}_t$ is of the form $(\mathbf{a}_i, y_i) \in \mathbb{R}^d \times \{1, \dots, C\}$. Here \mathbf{a}_i is a feature vector and $y_i \in \{1, \dots, C\}$ is the corresponding label. We define

⁴<http://yann.lecun.com/exdb/mnist/>

⁵<https://www.cs.toronto.edu/~kriz/cifar.html>

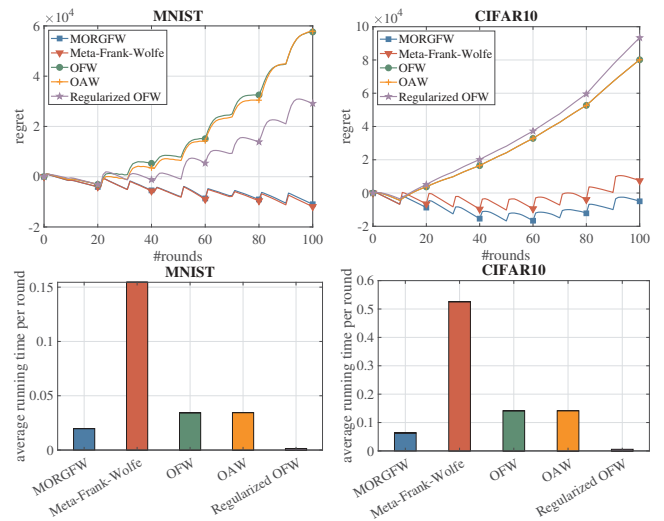


Figure 2: Results on the online multiclass logistic regression task in the adversarial setting (left: MNIST, right: CIFAR10). The left column shows the regret versus the number of rounds. The right column shows the average running time per round of each method.

f_t as the multiclass logistic loss function

$$f_t(\mathbf{W}) = - \sum_{i \in \mathcal{B}_t} \sum_{c=1}^C \mathbf{1}\{y_i = c\} \log \frac{\exp(\mathbf{W}_c^T \mathbf{a}_i)}{\sum_{j=1}^C \exp(\mathbf{W}_j^T \mathbf{a}_i)},$$

and set the constraint $\mathcal{C} = \{\mathbf{W} \in \mathbb{R}^{d \times C} : \|\mathbf{W}\|_1 \leq r\}$ for some constant $r \in \mathbb{R}_+$, where $\|\mathbf{W}\|_1$ denotes the matrix ℓ_1 norm, i.e., $\|\mathbf{W}\|_1 = \max_{1 \leq j \leq C} \sum_{i=1}^d |[\mathbf{W}]_{ij}|$. We note that the loss function f_t is convex and smooth. We consider both stochastic and adversarial online settings in this experiment. In the stochastic setting, each subset of data \mathcal{B}_t is sampled i.i.d. from the whole dataset. In the adversarial setting, we first sort data points by class label and then select $\{\mathcal{B}_1, \dots, \mathcal{B}_T\}$ sequentially from these datasets after sorting. For the MNIST dataset, we set $|\mathcal{B}_t| = 600$ and $r = 8$. For CIFAR10, we set $|\mathcal{B}_t| = 500$ and $r = 32$.

In the stochastic setting, we compare the performance of ORGFW with OFW, Regularized OFW, OAW, and OSFW. In the implementation of all these methods, we use the exact gradient ∇f_t in the t -th iteration since it can be computed efficiently. We report the regret and the average per-round running time in the first two rows of Figure 1. It can be seen from the top row that the regret of ORGFW is lower than OSFW and Regularized OFW, and slightly worse than OFW and OAW. Nevertheless, the second row shows that the per-round computational cost of ORGFW is significantly better than OFW and OAW. This implies that ORGFW has the advantages of low regret and low computational cost at the same time. We also include the suboptimality in the third row of Figure 1 to demonstrate the efficiency of ORGFW to solve the stochastic optimization problem $\min_{\mathbf{W} \in \mathcal{C}} f(\mathbf{W}) := \mathbb{E}_{\mathcal{B}_t} [f_t(\mathbf{W})]$. One can see that ORGFW outperforms all the other methods in terms of the suboptimality versus the running time.

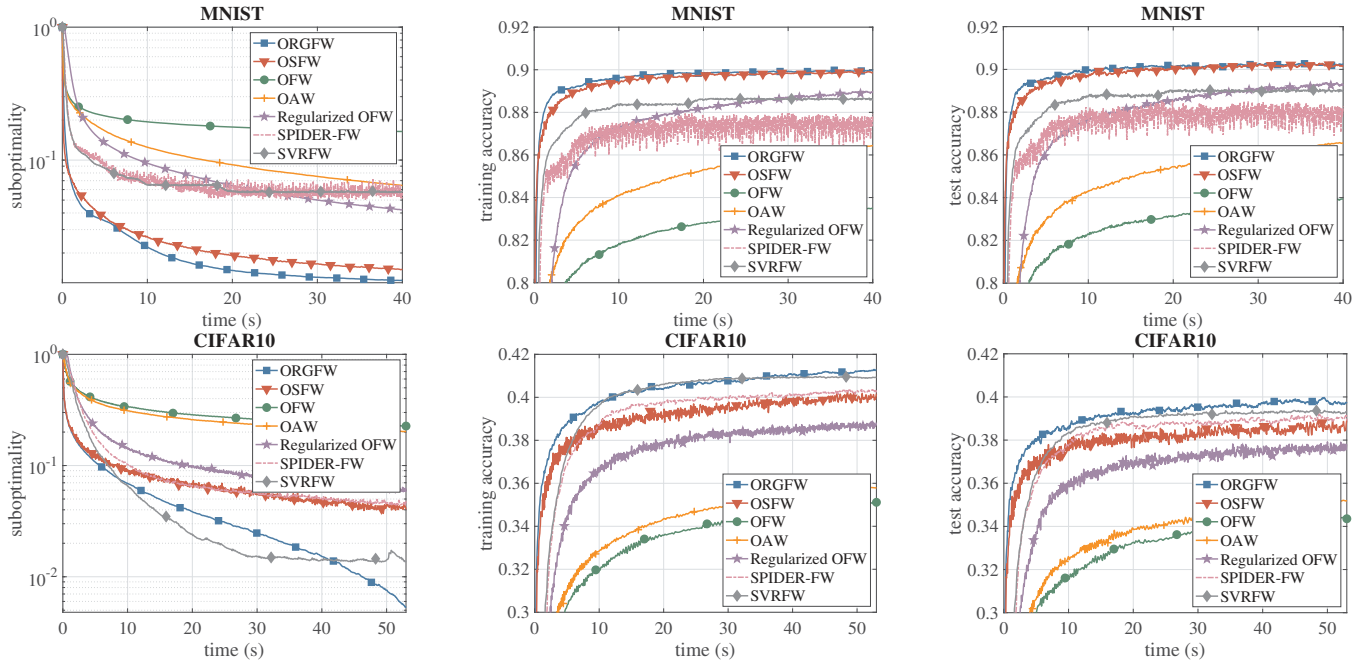


Figure 3: Results on the one-hidden-layer neural network optimization problem (top: MNIST, bottom: CIFAR10). The left column shows the suboptimality versus the running time. The middle (resp., right) column shows the training (resp., test) accuracy versus the running time.

In the adversarial setting, we compare MORGFW with OFW, Regularized OFW, OAW, and Meta-Frank-Wolfe. We set the number of rounds to $T = 100$ and set the parameter K in MORGFW and Meta-Frank-Wolfe to $K = T$ and $K = T^{3/2}$ as suggested by the theory, respectively. The results are shown in Figure 2. From the first row, we can see that the regret of MORGFW is comparable or lower than that of Meta-Frank-Wolfe and is significantly better than those of OFW, OAW, and Regularized OFW. We note that the zig-zag phenomenon is due to the adversarial nature of the loss function sequence. From the second row, we can see that the per-round computational cost of MORGFW is only worse than Regularized OFW. This confirms the advantages of MORGFW in achieving low regret and maintaining low computational cost simultaneously.

6.2 Training a One-hidden-layer Neural Network

In the second experiment, we focus on training a one-hidden-layer neural network with an additional ℓ_1 norm constraint (Zhang et al. 2019). Specifically, given a multiclass data set $\{(\mathbf{a}_i, y_i)\}_{i=1}^n$ with $(\mathbf{a}_i, y_i) \in \mathbb{R}^d \times \{1, \dots, C\}$, we consider the following problem

$$\min_{\substack{\mathbf{W}_1 \in \mathbb{R}^{d \times m}, \mathbf{b}_1 \in \mathbb{R}^m \\ \mathbf{W}_2 \in \mathbb{R}^{m \times C}, \mathbf{b}_2 \in \mathbb{R}^C}} \mathbb{E}_i \left[h(y_i, \phi(\mathbf{W}_2^T \sigma(\mathbf{W}_1^T \mathbf{a}_i + \mathbf{b}_1) + \mathbf{b}_2)) \right],$$

subject to $\|\mathbf{W}_j\|_1 \leq r_w$, $\|\mathbf{b}_j\|_1 \leq r_b$, $\forall j \in \{1, 2\}$. Here, i is a random variable sampled uniformly from $\{1, \dots, n\}$, ϕ is the softmax function, $\sigma(x) := 1/(1 + \exp(-x))$ is the sigmoid function, and $h(y, \mathbf{p}) := -\sum_{c=1}^C \mathbf{1}\{y = c\} \log(\mathbf{p}_c)$

for a probability vector $\mathbf{p} = (\mathbf{p}_1, \dots, \mathbf{p}_C)^T$. We note that training a neural network subject to an ℓ_1 constraint via FW-type methods exactly corresponds to a dropout regularization (Ravi et al. 2019).

We compare ORGFW with OSFW, Regularized OFW, OFW, and OAW. We also include two state-of-the-art offline projection-free methods: SVRF (Hazan and Luo 2016; Reddi et al. 2016) and SPIDER-FW (Shen et al. 2019). For the online methods, diminishing step sizes are used and a mini-batch of 16 data points are revealed to them in each round. For SVRF and SPIDER-FW, we use constant step sizes as suggested by (Shen et al. 2019). We compare the performance of these methods on the MNIST and CIFAR10 datasets. For both datasets, we set $m = 10$. In addition, we set the ℓ_1 ball radii $r_w = r_b = 10$. The experimental results are shown in Figure 3. One can see that ORGFW has the best performance in terms of the suboptimality, the training accuracy, and the test accuracy.

7 Conclusion

We proposed two efficient projection-free online methods, ORGFW and MORGFW, for solving online convex optimization problems in stochastic and adversarial settings, respectively. We provided novel regret analysis, which shows that the proposed methods achieve nearly optimal $\tilde{O}(\sqrt{T})$ regret bounds with low computational costs. In addition, we provided convergence analysis for ORGFW in stochastic optimization problems. Experimental results validate the advantages of the proposed methods.

Acknowledgements

This work is supported by Zhejiang Provincial Natural Science Foundation of China under Grant No. LZ18F020002, and National Natural Science Foundation of China (Grant No: 61672376, 61751209, 61472347).

References

- Agarwal, A.; Hazan, E.; Kale, S.; and Schapire, R. E. 2006. Algorithms for portfolio management based on the newton method. In *ICML*, 9–16.
- Birge, J. R., and Louveaux, F. 1997. Introduction to stochastic programming.
- Bubeck, S., et al. 2015. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning* 8(3-4):231–357.
- Cesa-Bianchi, N., and Lugosi, G. 2006. *Prediction, learning, and games*. Cambridge university press.
- Chandrasekaran, V.; Sanghavi, S.; Parrilo, P. A.; and Willsky, A. S. 2009. Sparse and low-rank matrix decompositions. *IFAC Proceedings Volumes* 42(10):1493–1498.
- Chen, L.; Harshaw, C.; Hassani, H.; and Karbasi, A. 2018. Projection-free online optimization with stochastic gradient: From convexity to submodularity. In *ICML*, 814–823.
- Cohen, A., and Hazan, T. 2015. Following the perturbed leader for online structured learning. In *ICML*, 1034–1042.
- Cutkosky, A., and Boahen, K. A. 2017. Stochastic and adversarial online learning without hyperparameters. In *NeurIPS*, 5059–5067.
- Cutkosky, A., and Orabona, F. 2019. Momentum-based variance reduction in non-convex sgd. *arXiv preprint arXiv:1905.10018*.
- Defazio, A.; Bach, F.; and Lacoste-Julien, S. 2014. Saga: A fast incremental gradient method with support for non-strongly convex composite objectives. In *NeurIPS*, 1646–1654.
- Dekel, O.; Gilad-Bachrach, R.; Shamir, O.; and Xiao, L. 2012. Optimal distributed online prediction using mini-batches. *Journal of Machine Learning Research* 13(Jan):165–202.
- Duchi, J.; Hazan, E.; and Singer, Y. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research* 12(Jul):2121–2159.
- Fang, C.; Li, C. J.; Lin, Z.; and Zhang, T. 2018. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. In *NeurIPS*, 689–699.
- Frank, M., and Wolfe, P. 1956. An algorithm for quadratic programming. *Naval research logistics quarterly* 3(1-2):95–110.
- Guélat, J., and Marcotte, P. 1986. Some comments on wolfe’s ‘away step’. *Mathematical Programming* 35(1):110–119.
- Hazan, E., and Kale, S. 2012. Projection-free online learning. In *ICML*, 1843–1850.
- Hazan, E., and Luo, H. 2016. Variance-reduced and projection-free stochastic optimization. In *ICML*, 1263–1271.
- Hazan, E. 2008. Sparse approximate solutions to semidefinite programs. In *Latin American symposium on theoretical informatics*, 306–316. Springer.
- Hazan, E. 2016. Introduction to online convex optimization. *Foundations and Trends® in Optimization* 2:157–325.
- Hoi, S. C.; Sahoo, D.; Lu, J.; and Zhao, P. 2018. Online learning: A comprehensive survey. *arXiv preprint arXiv:1802.02871*.
- Johnson, R., and Zhang, T. 2013. Accelerating stochastic gradient descent using predictive variance reduction. In *NeurIPS*, 315–323.
- Kalai, A., and Vempala, S. 2005. Efficient algorithms for on-line decision problems. *Journal of Computer and System Sciences* 71(3):291–307.
- Lacoste-Julien, S.; Jaggi, M.; Schmidt, M.; and Pletscher, P. 2013. Block-coordinate Frank-Wolfe optimization for structural SVMs. In *ICML*, 53–61.
- Lafond, J.; Wai, H.-T.; and Moulines, E. 2015. On the online frank-wolfe algorithms for convex and non-convex optimizations. *arXiv preprint arXiv:1510.01171*.
- Lan, G., and Zhou, Y. 2016. Conditional gradient sliding for convex optimization. *SIAM Journal on Optimization* 26(2):1379–1409.
- Mokhtari, A.; Hassani, H.; and Karbasi, A. 2018. Stochastic conditional gradient methods: From convex minimization to submodular maximization. *arXiv preprint arXiv:1804.09554*.
- Nguyen, L. M.; Liu, J.; Scheinberg, K.; and Takáč, M. 2017. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *LCML*, 2613–2621.
- Nguyen, L. M.; Scheinberg, K.; and Takáč, M. 2018. Inexact sarah algorithm for stochastic optimization. *arXiv preprint arXiv:1811.10105*.
- Pinelis, I. 1994. Optimum bounds for the distributions of martingales in banach spaces. *The Annals of Probability* 22(4):1679–1706.
- Ravi, S. N.; Dinh, T.; Lokhande, V. S.; and Singh, V. 2019. Explicitly imposing constraints in deep networks via conditional gradients gives improved generalization and faster convergence. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 4772–4779.
- Reddi, S. J.; Sra, S.; Póczos, B.; and Smola, A. 2016. Stochastic frank-wolfe methods for nonconvex optimization. In *2016 54th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 1244–1251.
- Shalev-Shwartz, S., and Singer, Y. 2007. A primal-dual perspective of online learning algorithms. *Machine Learning* 69(2-3):115–142.
- Shalev-Shwartz, S. 2012. Online learning and online convex optimization. *Foundations and Trends® in Machine Learning* 4(2):107–194.
- Shen, Z.; Fang, C.; Zhao, P.; Huang, J.; and Qian, H. 2019. Complexities in projection-free stochastic non-convex minimization. In *The 22nd International Conference on Artificial Intelligence and Statistics*, 2868–2876.
- Xiao, L. 2010. Dual averaging methods for regularized stochastic learning and online optimization. *Journal of Machine Learning Research* 11(Oct):2543–2596.
- Yurtsever, A.; Sra, S.; and Cevher, V. 2019. Conditional gradient methods via stochastic path-integrated differential estimator. In *ICML*, 7282–7291.
- Zhang, W.; Zhao, P.; Zhu, W.; Hoi, S. C.; and Zhang, T. 2017. Projection-free distributed online learning in networks. In *ICML*, 4054–4062.
- Zhang, M.; Chen, L.; Mokhtari, A.; Hassani, H.; and Karbasi, A. 2019. Quantized frank-wolfe: Faster optimization, lower communication, and projection free. *arXiv preprint arXiv:1902.06332*.
- Zhou, D.; Xu, P.; and Gu, Q. 2018. Stochastic nested variance reduced gradient descent for nonconvex optimization. In *NeurIPS*, 3921–3932.
- Zinkevich, M. 2003. Online convex programming and generalized infinitesimal gradient ascent. In *ICML*, 928–936.