Projection-free Online Learning in Dynamic Environments

Yuanyu Wan, Bo Xue, Lijun Zhang*
National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China
{wanyy, xueb, zhanglj}@lamda.nju.edu.cn

Abstract

To efficiently solve high-dimensional problems with complicated constraints, projection-free online learning has received ever-increasing research interest. However, previous studies either focused on static regret that is not suitable for dynamic environments, or only established the dynamic regret bound under the smoothness of losses. In this paper, without the condition of the smoothness, we propose a novel projection-free online algorithm, and achieve an $O(\max\{T^{2/3}V_T^{1/3},\sqrt{T}\})$ dynamic regret bound for convex functions and an $O(\max\{\sqrt{TV_T\log T},\log T\})$ dynamic regret bound for strongly convex functions, where T is the time horizon and V_T denotes the variation of loss functions. Specifically, we first improve an existing projection-free algorithm called online conditional gradient (OCG) to enjoy small dynamic regret bounds with the prior knowledge of V_T. To work with unknowable V_T, we maintain multiple instances of the improved OCG that can handle different functional variations, and combine them with a meta-algorithm that can track the best one. Experimental results validate the efficiency and effectiveness of our algorithm.

Introduction

Online convex optimization (OCO) is a powerful framework for online learning, which enjoys both computational efficiency and theoretical guarantees (Shalev-Shwartz 2011). According to the protocol of OCO, it is a repeated game between a learner and an adversary. In each round t, the learner first chooses a decision $\mathbf{x}_t\in\mathcal{K}$, where $\mathcal{K}$ is a convex decision set. Then, the adversary reveals a convex loss function $f_t(\mathbf{x}):\mathcal{K}\to\mathbb{R}$, and the learner suffers a loss $f_t(\mathbf{x}_t)$. To measure the performance of the learner, the static regret with respect to the best fixed decision is commonly used, which is defined as

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

where T is the total number of rounds.

Over the past decades, various online algorithms have been put forward to minimize the static regret under different scenarios, which can be divided into projection-based algorithms (Zinkevich 2003; Hazan, Agarwal, and Kale 2007; Shalev-Shwartz and Singer 2007) and projection-free algorithms (Hazan and Kale 2012; Hazan 2016; Garber and Hazan 2016). Specifically, projection-based algorithms such as online gradient descent (OGD) (Zinkevich 2003) and regularized follow the leader (RFTL) (Shalev-Shwartz and Singer 2007; Hazan 2016) perform one projection step in each round, which could be computationally expensive for high-dimensional problems with complicated constraints. In contrast, projection-free algorithms such as online conditional gradient (OCG) (Hazan and Kale 2012; Hazan 2016) and its variants replace the projection step with one linear optimization step, which can be carried out efficiently and have received ever-increasing attention (Zhang et al. 2017b; Chen et al. 2018; Chen, Zhang, and Karbasi 2019; Wan, Tu, and Zhang 2020; Wan and Zhang 2021).

Although the static regret has been extensively studied for projection-based and projection-free algorithms, it is not suitable to measure the performance of the learner in dynamic environments, where the best decision may frequently change. To address this limitation, recent advances in OCO focused on dynamic regret which measures the performance of the learner against a sequence of local minimizers

$$R_D=R(\mathbf{x}_1^*,\cdots,\mathbf{x}_T^*)=\sum_{t=1}^{T}f_t(\mathbf{x}_t)-\sum_{t=1}^{T}f_t(\mathbf{x}_t^*)\quad(1)$$

where $\mathbf{x}_t^*\in\operatorname{argmin}_{\mathbf{x}\in\mathcal{K}}f_t(\mathbf{x})$ is a local minimizer, and proposed many projection-based algorithms for minimizing the dynamic regret (Besbes, Gur, and Zeevi 2015; Jadbabaie et al. 2015; Mokhtari et al. 2016; Yang et al. 2016; Zhang et al. 2018; Zhang, Lu, and Yang 2020; Zhang 2020).

Due to the arbitrary fluctuation in the loss functions, it is impossible to achieve a sub-linear dynamic regret bound unless introducing some conditions on the comparator sequence or the function sequence (Jadbabaie et al. 2015). A common condition introduced by previous studies (Besbes, Gur, and Zeevi 2015; Zhang et al. 2018) is that the functional variation defined as

$$V_T=\sum_{t=2}^{T}\max_{\mathbf{x}\in\mathcal{K}}|f_t(\mathbf{x})-f_{t-1}(\mathbf{x})|$$

is sub-linear in T. If the value of V_T is not small than 1 and given, Besbes, Gur, and Zeevi (2015) achieved an

*Lijun Zhang is the corresponding author.

$O(T^{2/3}V_T^{1/3})$ dynamic regret bound for convex functions and an $O(\log T\sqrt{TV_T})$ dynamic regret bound for strongly convex functions by properly restarting OGD. Moreover, Zhang et al. (2018) showed that maintaining multiple OGD and combining them carefully are able to achieve almost the same dynamic regret bounds for convex and strongly convex functions without any prior knowledge of V_T.

However, the projection step could limit their applications. While recent studies (Kalhan et al. 2019, 2020) proposed projection-free algorithms for dynamic environments, they only established the dynamic regret bound for smooth functions. To tackle this limitation, we propose a novel projection-free algorithm named as Multi-OCG+ to minimize the dynamic regret without the smoothness. Specifically, we first develop an improved variant of OCG named as OCG+, which achieves an $O(\max\{T^{2/3}V_T^{1/3}, \sqrt{T}\})$ dynamic regret bound for convex functions and an $O(\max\{\sqrt{TV_T\log T}, \log T\})$ dynamic regret bound for strongly convex functions with the prior knowledge of V_T. Compared with the original OCG, there exist three critical changes.

- Besides the surrogate loss function defined in OCG, our OCG+ further introduces a new surrogate loss function from Garber and Hazan (2016) to utilize the strong convexity.
- The restarting strategy (Besbes, Gur, and Zeevi 2015) has been utilized in our OCG+ to handle a specific functional variation.
- Different from OCG that only performs 1 linear optimization step in each round, our OCG+ performs multiple linear optimization steps.

Note that the third change is inspired by previous studies (Chen et al. 2018; Xie et al. 2020; Hazan and Minasyan 2020), which have utilized multiple linear optimization steps to improve the static regret of projection-free algorithms. Furthermore, to handle the unknown V_T, our Multi-OCG+ maintains multiple instances of OCG+ with different restarting frequencies, and combines them with a meta-algorithm that can track the best one. We prove that the dynamic regret bounds of Multi-OCG+ are on the same order as those bounds of OCG+.

Related Work

In this section, we briefly review the related work on the static and dynamic regret in the context of OCO.

Static Regret

Since the pioneering work of Zinkevich (2003), the static regret has been extensively studied under different scenarios (Hazan, Agarwal, and Kale 2007; Shalev-Shwartz 2011; Hazan 2016). For convex functions, there are several algorithms such as the classical OGD (Zinkevich 2003) and RFTL (Hazan 2016) that achieved the $O(\sqrt{T})$ static regret bound. For strongly convex functions, Hazan, Agarwal, and Kale (2007) established the $O(\log T)$ static regret bound for OGD. The $O(\sqrt{T})$ rate for convex functions and the $O(\log T)$ rate for strongly convex functions are known to be minimax optimal (Abernethy et al. 2008). However, in each round, these algorithms need to perform one projection step, which could be computationally expensive for high-dimensional problems with complicated constraints (Hazan and Kale 2012). For example, OGD needs to perform the following projection step

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \|\mathbf{x} - (\mathbf{x}_t - \eta_t \nabla f_t(\mathbf{x}_t))\|_2^2$$

where η_t is a parameter.

To tackle this computational bottleneck, OCG (Hazan and Kale 2012; Hazan 2016) was put forward, which is the first projection-free online algorithm for OCO. The key idea of OCG is to replace the projection step with one linear optimization step, as follows

$$\mathbf{v}_t = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \left\{\nabla F_t(\mathbf{x}_t)^\top \mathbf{x}\right\},\ \mathbf{x}_{t+1} = \mathbf{x}_t + \sigma_t(\mathbf{v}_t - \mathbf{x}_t)$$

where $F_t(\mathbf{x}) = \eta\sum_{i=1}^{t-1}\nabla f_i(\mathbf{x}_i)^\top \mathbf{x} + \|\mathbf{x}-\mathbf{x}_1\|_2^2$ is a surrogate loss function, η and σ_t are parameters, which can be carried out efficiently. However, the static regret bound of OCG is $O(T^{3/4})$, which is worse than the optimal $O(\sqrt{T})$ bound for convex functions. Recently, projection-free algorithms with optimal static regret for both convex and strongly convex functions have been proposed for special decision sets such as polytope (Garber and Hazan 2016) and smooth set (Levy and Krause 2019). Furthermore, if $O(T^{3/2})$ linear optimization steps are allowed in each round, Chen et al. (2018) have achieved the optimal static regret bound $O(\sqrt{T})$ for convex and smooth functions. In the same setting, Xie et al. (2020) reduced the number of linear optimization steps to $O(T)$ while achieved the static regret bound of $O(\sqrt{T}\log T)$. Hazan and Minasyan (2020) proposed a projection-free algorithm by estimating the expected decision of follow the perturbed leader (FPL) algorithm (Hazan 2016) with enough samples, each of which is computed by one linear optimization step. If T linear optimization steps are allowed in each round, their algorithm attains a regret bound of $O(\sqrt{dT})$ for convex functions, where d is the dimensionality and the dependence on d is caused by the randomized regularization in FPL.

Dynamic Regret

In the pioneering work of Zinkevich (2003), a more general definition of dynamic regret is proposed to measure the performance of the learner against any sequence of comparators

$$R(\mathbf{u}_1,\cdots,\mathbf{u}_T) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t) \tag{2}$$

where $\mathbf{u}_1,\cdots,\mathbf{u}_T \in \mathcal{K}$. To bound the general dynamic regret, Zinkevich (2003) introduced the path-length defined as $P(\mathbf{u}_1,\cdots,\mathbf{u}_T) = \sum_{t=2}^{T}\|\mathbf{u}_t - \mathbf{u}_{t-1}\|_2$ and proved that OGD enjoys a general dynamic regret bound of $O(\sqrt{T}(1+P(\mathbf{u}_1,\cdots,\mathbf{u}_T)))$. Similarly, using a dynamic model $\Phi_t(\cdot)$ to predict a reference point for the t-th round, Hall and Willett (2013) introduced a variant of the path-length defined as $P'(\mathbf{u}_1,\cdots,\mathbf{u}_T) =$

$\sum_{t=2}^{T} \|\mathbf{u}_t - \Phi_t(\mathbf{u}_{t-1})\|_2$ and proposed a novel algorithm to achieve a general dynamic regret bound of $O\sqrt{T}\,(1 + P'(\mathbf{u}_1, \cdots, \mathbf{u}_T)))$.

Furthermore, Zhang, Lu, and Zhou (2018) proposed a serial of novel algorithms that reduce the general dynamic regret bounds to $O(\sqrt{T(1 + P(\mathbf{u}_1, \cdots, \mathbf{u}_T))})$ and $O(\sqrt{T\,(1 + P'(\mathbf{u}_1, \cdots, \mathbf{u}_T))})$, respectively. Because of $\mathbf{x}_t^* \in \operatorname{argmin}_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x})$, it is easy to verify that

$$R(\mathbf{u}_1, \cdots, \mathbf{u}_T) \leq R(\mathbf{x}_1^*, \cdots, \mathbf{x}_T^*)$$

which implies that the dynamic regret defined in (1) can be treated as the worst case of the general one defined in (2). Additionally, there also exist many studies that directly investigate the worst case under different scenarios (Besbes, Gur, and Zeevi 2015; Jadbabaie et al. 2015; Mokhtari et al. 2016; Yang et al. 2016; Zhang et al. 2017a, 2018).

However, the above algorithms are based on the projection step, which could limit their applications. Recently, Kalhan et al. (2019, 2020) proposed two projection-free algorithms termed OFW and Meta-Frank Wolfe for dynamic environments. For smooth functions, they proved that OFW and Meta-Frank Wolfe with $O(T^a)$ linear optimization steps per round have the dynamic regret bound of $O(\sqrt{T}(1 + V_T + \sqrt{D_T}))$ and $O(1 + V_T + T^{1-a} + R_D^{\mathcal{A}})$ respectively, where $D_T = \sum_{t=2}^{T} \|\nabla f_t(\mathbf{x}_t) - \nabla f_{t-1}(\mathbf{x}_{t-1})\|_2^2$ and $R_D^{\mathcal{A}}$ denotes the dynamic regret of an online linear optimization oracle. Compared with Kalhan et al. (2019, 2020), our work has significant differences. First, we do not require the smoothness of functions, which is not satisfied by some common losses such as the hinge loss and absolute loss. Second, we further utilize the strongly convexity to achieve a better regret bound, which was not not studied by Kalhan et al. (2019, 2020). Third, as long as V_T is sublinear in T, our regret of $O(\max\{T^{2/3}V_T^{1/3}, \sqrt{T}\})$ for convex functions is sublinear in T. By contrast, the regret of OFW could be $O(T)$ if $V_T = O(\sqrt{T})$ or $D_T = O(T)$. Moreover, since $R_D^{\mathcal{A}}$ could be $O(T)$ in the worst case, it is not comparable between the regret of Meta-Frank Wolfe and our regret.

Main Results

In this section, we first introduce necessary preliminaries. Then, we present our OCG+ that is an improved variant of OCG, and establish small dynamic regret bounds with the prior knowledge of V_T. Finally, we present our Multi-OCG+, which enjoy similar dynamic regret bounds without any prior knowledge of V_T.

Preliminaries

Following previous studies on OCO (Hazan and Kale 2012; Zhang, Lu, and Zhou 2018), we introduce two common assumptions, which will be used to bound the dynamic regret.

Assumption 1 *The diameter of the convex decision set $\mathcal{K}$ is bounded by D, i.e., $\|\mathbf{x} - \mathbf{y}\|_2 \leq D$ for any $\mathbf{x} \in \mathcal{K}, \mathbf{y} \in \mathcal{K}$.*

Assumption 2 *At each round t, the loss function $f_t(\mathbf{x})$ is G-Lipschitz over $\mathcal{K}$, i.e., $|f_t(\mathbf{x}) - f_t(\mathbf{y})| \leq G\|\mathbf{x} - \mathbf{y}\|_2$ for any $\mathbf{x} \in \mathcal{K}, \mathbf{y} \in \mathcal{K}$.*

Algorithm 1 CG

1: **Input**: feasible set $\mathcal{K}$, K, $F(\mathbf{x})$, $\mathbf{x}_{\text{in}}$
2: $\mathbf{z}_0 = \mathbf{x}_{\text{in}}$
3: **for** $k = 0, 1, \cdots, K-1$ **do**
4: $\quad \mathbf{v}_k \in \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \{\nabla F(\mathbf{z}_k)^\top \mathbf{x}\}$
5: $\quad \sigma_k = \operatorname*{argmin}_{\sigma\in[0,1]} \{F(\mathbf{z}_k + \sigma(\mathbf{v}_k - \mathbf{z}_k))\}$
6: $\quad \mathbf{z}_{k+1} = \mathbf{z}_k + \sigma_k(\mathbf{v}_k - \mathbf{z}_k)$
7: **end for**
8: **return** $\mathbf{x}_{\text{out}} = \mathbf{z}_K$

Then, we recall the standard definitions for smooth and strongly convex functions (Boyd and Vandenberghe 2004).

Definition 1 *Let $f(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$ be a function over $\mathcal{K}$. It is called β-smooth over $\mathcal{K}$ if for all $\mathbf{x} \in \mathcal{K}, \mathbf{y} \in \mathcal{K}$*

$$f(\mathbf{y}) \leq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x}) + \frac{\beta}{2}\|\mathbf{y} - \mathbf{x}\|_2^2.$$

Definition 2 *Let $f(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$ be a function over $\mathcal{K}$. It is called α-strongly convex over $\mathcal{K}$ if for all $\mathbf{x} \in \mathcal{K}, \mathbf{y} \in \mathcal{K}$*

$$f(\mathbf{y}) \geq f(\mathbf{x}) + \nabla f(\mathbf{x})^\top(\mathbf{y} - \mathbf{x}) + \frac{\alpha}{2}\|\mathbf{y} - \mathbf{x}\|_2^2.$$

Furthermore, we introduce conditional gradient (CG) (Frank and Wolfe 1956; Jaggi 2013), which will be utilized as a subroutine of our proposed algorithms. Given a function $F(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$ and an initial point $\mathbf{z}_0 = \mathbf{x}_{\text{in}} \in \mathcal{K}$, the idea of CG is to iteratively perform linear optimization step K times

$$\mathbf{v}_k = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \{\nabla F(\mathbf{z}_k)^\top \mathbf{x}\},\ \mathbf{z}_{k+1} = \mathbf{z}_k + \sigma_k(\mathbf{v}_k - \mathbf{z}_k)$$

where $\sigma_k = \operatorname{argmin}_{\sigma\in[0,1]} \{F(\mathbf{z}_k + \sigma(\mathbf{v}_k - \mathbf{z}_k))\}$ is selected by line search. The detailed procedures of CG are summarized in Algorithm 1. Compared with performing linear optimization step once, CG can output a point $\mathbf{x}_{\text{out}}$ such that $F(\mathbf{x}_{\text{out}})$ is small enough.

Note that CG is originally an algorithm for offline optimization, and OCG proposed by Hazan and Kale (2012) is its extension for online learning. However, OCG only performs 1 linear optimization in each round, and its static regret bound $O(T^{3/4})$ is worse that the optimal bound $O(\sqrt{T})$. As a result, directly applying the restarting strategy (Besbes, Gur, and Zeevi 2015) to OCG could not achieve the optimal dynamic regret. Although previous studies improved the static regret of OCG by performing multiple linear optimization steps in each round, there exist some limitations. Chen et al. (2018) and Xie et al. (2020) used stochastic gradients to update decision, and their results required the smoothness of loss functions. The regret bound of Hazan and Minasyan (2020) has additional dependence on the dimensionality d, which is caused by the randomized regularization. Different from these studies that employed some randomized methods, we utilize CG that is a deterministic method to improve the static regret of OCG, and further propose a projection-free algorithm with small dynamic regret bounds for convex and strongly convex functions.

Algorithm 2 OCG+

1: **Input**: feasible set $\mathcal{K}$, the modulus of strong convexity λ, K_γ, γ
2: Set $\eta_\gamma = \frac{D}{G\sqrt{\gamma}}$
3: **for** $t = 1, \cdots, T$ **do**
4:   **if** $t \bmod \gamma = 1$ **then**
5:     $s_\gamma = t$ and choose $\mathbf{x}_t^\gamma \in \mathcal{K}$
6:   **end if**
7:   **if** $\lambda = 0$ **then**
8:     $F_{t+1}^\gamma(\mathbf{x}) = \eta_\gamma \sum_{i=s_\gamma}^{t} \nabla f_i(\mathbf{x}_i^\gamma)^\top \mathbf{x} + \|\mathbf{x} - \mathbf{x}_{s_\gamma}^\gamma\|_2^2$
9:   **else**
10:     $F_{t+1}^\gamma(\mathbf{x}) = \sum_{i=s_\gamma}^{t} \left(\nabla f_i(\mathbf{x}_i^\gamma)^\top \mathbf{x} + \frac{\lambda}{2}\|\mathbf{x} - \mathbf{x}_i^\gamma\|_2^2\right) + \frac{\lambda}{2}\|\mathbf{x} - \mathbf{x}_{s_\gamma}^\gamma\|_2^2$
11:   **end if**
12:   $\mathbf{x}_{t+1}^\gamma = \mathrm{CG}(\mathcal{K}, K_\gamma, F_{t+1}^\gamma(\mathbf{x}), \mathbf{x}_t^\gamma)$
13: **end for**

OCG+ for a Knowable Function Variation

To cope with dynamic environments, we investigate the dynamic regret defined in (1), which measures the performance of the learner against a sequence of local minimizers. Following previous studies (Besbes, Gur, and Zeevi 2015; Zhang et al. 2018), the goal of this paper is to upper bound the dynamic regret by the functional variation V_T.

By utilizing CG as a subroutine, the detailed procedures of our algorithm are presented in Algorithm 2, where λ is the modulus of strong convexity of loss functions, γ and K_γ are parameters, which is named as improved online conditional gradient (OCG+). Compared with the original OCG, we have made three critical changes to achieve small dynamic regret bounds. First, besides the surrogate loss function for convex functions in line 8 of OCG+, a new surrogate loss function $F_{t+1}^\gamma(\mathbf{x})$ in line 10 is introduced from Garber and Hazan (2016), which can utilize the strong convexity. Note that Garber and Hazan (2016) utilized this surrogate loss function to propose a projection-free online algorithm over polyhedral sets, which only performs one linear optimization step in each round and attains the optimal static regret. By contrast, our algorithm are designed to minimize dynamic regret over any convex decision set, which requires multiple linear optimization steps in each round. Second, we utilize the restarting strategy (Besbes, Gur, and Zeevi 2015) in our OCG+ to handle a specific functional variation. The restarting frequency of our OCG+ is determined by the parameter γ. Let $r = \lceil T/\gamma \rceil$, $q_i = (i-1)\gamma + 1$ for $i = 1, \cdots, r$ and $q_{r+1} = T + 1$. It is easy to verify that OCG+ essentially performs the same steps on time intervals

$$[q_1, q_2 - 1], [q_2, q_3 - 1], \cdots, [q_r, q_{r+1} - 1] \tag{3}$$

successively. Third, during the j-th time interval in (3), our OCG+ invokes CG shown in Algorithm 1 as

$$\mathbf{x}_{t+1}^\gamma = \mathrm{CG}(\mathcal{K}, K_\gamma, F_{t+1}^\gamma(\mathbf{x}), \mathbf{x}_t^\gamma)$$

to choose the decision $\mathbf{x}_{t+1}^\gamma$.

The following two theorems present the dynamic regret bounds of Algorithm 2 for convex and strongly convex functions, respectively.

Theorem 1 *Under Assumptions 1 and 2, for convex losses, Algorithm 2 with $\gamma \le T$ and $K_\gamma = \gamma$ ensures*

$$R_D \le \frac{8TGD}{\sqrt{\gamma}} + 2\gamma V_T.$$

Theorem 2 *Under Assumptions 1 and 2, for λ-strongly convex losses, Algorithm 2 with $\gamma \le T$ and $K_\gamma = \gamma^2$ ensures*

$$R_D \le \frac{2T(c_1 + c_2 \ln(\gamma + 1))}{\gamma} + 2\gamma V_T$$

where $c_1 = \frac{\lambda D^2}{2} + 2(G + \lambda D)D$ and $c_2 = \frac{2(G+\lambda D)^2}{\lambda}$.

Based on Theorems 1 and 2, we derive the specific dynamic regret bounds of Algorithm 2.

Corollary 1 *Assume that $f_t(\mathbf{x})$ is convex for any $t \in [T]$. If $V_T \ge \sqrt{1/T}$, under Assumptions 1 and 2, Algorithm 2 with $\gamma = \lfloor (T/V_T)^{2/3} \rfloor$ and $K_\gamma = \gamma$ ensures*

$$R_D \le (8\sqrt{2}GD + 2)T^{2/3}V_T^{1/3}.$$

Otherwise, under Assumptions 1 and 2, Algorithm 2 with $\gamma = T$ and $K_\gamma = \gamma$ achieves

$$R_D \le 8GD\sqrt{T} + 2\sqrt{T}.$$

Corollary 2 *Let $c_1 = \frac{\lambda D^2}{2} + 2(G + \lambda D)D$ and $c_2 = \frac{2(G+\lambda D)^2}{\lambda}$. Assume that $f_t(\mathbf{x})$ is λ-strongly convex for any $t \in [T]$. If $V_T \ge \ln(T+1)/T$, under Assumptions 1 and 2, Algorithm 2 with $\gamma = \left\lfloor \sqrt{T\ln(T+1)/V_T} \right\rfloor$ and $K_\gamma = \gamma^2$ ensures*

$$R_D \le (4c_1 + 4c_2 + 2)\sqrt{TV_T \ln(T+1)}.$$

Otherwise, under Assumptions 1 and 2, Algorithm 2 with $\gamma = T$ and $K_\gamma = \gamma^2$ achieves

$$R_D \le 2c_1 + (2c_2 + 2)\ln(T+1).$$

Remark From Corollaries 1 and 2, if the value of V_T is available, our OCG+ achieves an $O(\max\{T^{2/3}V_T^{1/3}, \sqrt{T}\})$ dynamic regret bound for convex functions and an $O(\max\{\sqrt{TV_T \log T}, \log T\})$ dynamic regret bound for strongly convex functions. If $V_T \ge 1$, these bounds reduce to $O(T^{2/3}V_T^{1/3})$ for convex functions which matches the minimax rate proved by Besbes, Gur, and Zeevi (2015), and $O(\sqrt{TV_T \log T})$ for strongly convex losses which nearly matches the minimax rate of $O(\sqrt{TV_T})$ proved by Besbes, Gur, and Zeevi (2015) up to a polylogarithmic factor.

Multi-OCG+ for Unknown Function Variations

However, in practice, the value of V_T could be unknown, which limits the applications of OCG+. To tackle this limitation, an efficient strategy to search the optimal γ for OCG+ is required. Note that similar problems also exist in previous studies on OCO (van Erven and Koolen 2016; Zhang, Lu, and Zhou 2018; Wang, Lu, and Zhang 2019; Wang et al. 2020), where it is necessary to search the optimal parameter for other algorithms. The main idea of their solutions is to run multiple instances of their algorithms with different parameters, and combine them with a meta-algorithm based on the exponentially weighted average forecaster (Cesa-Bianchi and Lugosi 2006), which is able to track the best instance.

Inspired by this idea, we define a set $\mathcal{H} = \{\gamma_1, \cdots, \gamma_N\}$ of N values for parameter γ, and activate a set of experts $\{E^\gamma|\gamma \in \mathcal{H}\}$ by invoking OCG+ shown in Algorithm 2 as

$$E^\gamma = \text{OCG+}(\mathcal{K}, \lambda, K_\gamma, \gamma).$$

In each round, we simultaneously run these experts $\{E^\gamma|\gamma \in \mathcal{H}\}$ to generate a set of decisions $\{\mathbf{x}_t^\gamma|\gamma \in \mathcal{H}\}$. Then, we adopt the exponentially weighted average forecaster as follows to generate a weight w_t^γ for each expert E^γ, and combine $\{\mathbf{x}_t^\gamma|\gamma \in \mathcal{H}\}$ as $\mathbf{x}_t = \sum_{\gamma\in\mathcal{H}} w_t^\gamma \mathbf{x}_t^\gamma$. According to Cesa-Bianchi and Lugosi (2006), the initial weight of each expert E^{γ_i} is set to be $w_1^{\gamma_i} = \frac{C}{i(i+1)}$ where $C = 1 + \frac{1}{N}$ is used to normalize these weights. In each round t, after observing the loss function, the weights of experts are updated as

$$w_{t+1}^\gamma = \frac{w_t^\gamma \exp(-\tau f_t(\mathbf{x}_t^\gamma))}{\sum_{\gamma\in\mathcal{H}} w_t^\gamma \exp(-\tau f_t(\mathbf{x}_t^\gamma))}$$

where $\tau > 0$ is a constant. The detailed procedures of our algorithm are summarized in Algorithm 3, and it is named as Multi-OCG+. Compared with OCG+, our Multi-OCG+ does not require any prior knowledge of V_T and enjoys the following dynamic regret bounds.

Theorem 3 *Assume that $f_t(\mathbf{x})$ is convex for any $t \in [T]$. Let $\mathcal{H} = \left\{\gamma_i = 2^i | i = 0, \cdots, N\right\}$ where $N = \lfloor\log_2(T)\rfloor$, and $\tau = \sqrt{8/TG^2D^2}$. For each expert in $\{E^\gamma|\gamma \in \mathcal{H}\}$, let $K_\gamma = \gamma$. Under Assumptions 1 and 2, Algorithm 3 ensures*

$$\begin{aligned} R_D \leq& \max\left\{c_3\sqrt{T}, c_4 T^{2/3} V_T^{1/3}\right\} \\ &+ \sqrt{TG^2D^2/8}\,(1 + 2\ln N) \end{aligned}$$

where $c_3 = 8\sqrt{2}GD + 2$ and $c_4 = 16GD + 2$.

Theorem 4 *Assume that $f_t(\mathbf{x})$ is λ-strongly convex for any $t \in [T]$. Let $\mathcal{H} = \left\{\gamma_i = 2^i | i = 0, \cdots, N\right\}$ where $N = \lfloor\log_2(T)\rfloor$, and $\tau = \lambda/G^2$. For each expert in $\{E^\gamma|\gamma \in \mathcal{H}\}$, let $K_\gamma = \gamma^2$. Under Assumptions 1 and 2, Algorithm 3 ensures*

$$\begin{aligned} &R_D \\ &\leq \max\left\{\begin{aligned} 4c_1 + (4c_2 + 2)\ln(T+1) + \frac{2G^2}{\lambda}\ln N \\ (8c_1 + 8c_2 + 2)\sqrt{TV_T\ln(T+1)} + \frac{2G^2}{\lambda}\ln N \end{aligned}\right. \end{aligned}$$

where $c_1 = \frac{\lambda D^2}{2} + 2(G + \lambda D)D$ and $c_2 = \frac{2(G+\lambda D)^2}{\lambda}$.

Remark To compare with previous studies, we consider the case of $V_T \geq 1$. For convex functions, Theorem 3 shows that the dynamic regret bound of our Algorithm 3 is on the order of $O(T^{2/3}V_T^{1/3})$ where we treat the double logarithmic as a constant, which also matches the minimax rate proved by Besbes, Gur, and Zeevi (2015). For strongly-convex functions, Theorem 4 shows that the dynamic regret bound of our Algorithm 3 is on the order of $O(\sqrt{TV_T \log T})$, which nearly matches the minimax rate proved by Besbes, Gur, and Zeevi (2015) up to a polylogarithmic factor. Although Besbes, Gur, and Zeevi (2015) established similar dynamic regret bounds, their algorithm needs to know the upper bound of V_T and is not projection-free. Besides, our bound for convex functions is slightly better than the bound $O(T^{2/3}V_T^{1/3}\log^{1/3} T)$ achieved by another projection-based algorithm (Zhang et al. 2018) that does not require any prior knowledge of V_T.

Algorithm 3 Multi-OCG+

1: **Input**: feasible set $\mathcal{K}$, strong convexity parameter λ, τ, $\mathcal{H} = \{\gamma_1, \cdots, \gamma_N\}$ and $K_\gamma, \forall\gamma \in \mathcal{H}$
2: Activate a set of experts $\{E^\gamma|\gamma \in \mathcal{H}\}$ by invoking Algorithm 2 as $E^\gamma = \text{OCG+}(\mathcal{K}, \lambda, K_\gamma, \gamma)$
3: Set $w_1^{\gamma_i} = \frac{C}{i(i+1)}$ for $i \in [N]$, where $C = 1 + \frac{1}{N}$
4: **for** $t = 1, \cdots, T$ **do**
5:     Receive $\mathbf{x}_t^\gamma$ from each expert E^γ and set $\mathbf{x}_t = \sum_{\gamma\in\mathcal{H}} w_t^\gamma \mathbf{x}_t^\gamma$
6:     Set $w_{t+1}^\gamma = \frac{w_t^\gamma \exp(-\tau f_t(\mathbf{x}_t^\gamma))}{\sum_{\gamma\in\mathcal{H}} w_t^\gamma \exp(-\tau f_t(\mathbf{x}_t^\gamma))}, \forall\gamma \in \mathcal{H}$
7: **end for**

Theoretical Analysis

In this section, we only provide the proofs of Theorems 1 and 2. Due to the limitation of space, we postpone the omitted proofs to the supplementary material.

Proof of Theorem 1

Let $r = \lceil T/\gamma\rceil$, $q_i = (i-1)\gamma + 1$ for $i = 1, \cdots, r$ and $q_{r+1} = T + 1$. First, we have

$$\begin{aligned} &\sum_{t=1}^{T} f_t(\mathbf{x}_t^\gamma) - \sum_{t=1}^{T} \min_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x}) \\ =&\sum_{i=1}^{r}\left(\sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}_t^\gamma) - \sum_{t=q_i}^{q_{i+1}-1}\min_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x})\right) \\ =&\sum_{i=1}^{r}\left(\underbrace{\sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}_t^\gamma) - \min_{\mathbf{x}\in\mathcal{K}}\sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x})}_{:=a_i}\right. \\ &\left.+\underbrace{\min_{\mathbf{x}\in\mathcal{K}}\sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}) - \sum_{t=q_i}^{q_{i+1}-1}\min_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x})}_{:b_i}\right). \end{aligned} \tag{4}$$

To bound a_i, we introduce the following lemma, which gives the static regret bound of Algorithm 2 over each interval in (3).

Lemma 1 *Assume that $f_t(\mathbf{x})$ is convex for any $t \in [T]$. Let $r = \lceil T/\gamma\rceil$, $q_i = (i-1)\gamma + 1$ for $i = 1, \cdots, r$ and $q_{r+1} =$*

$T+1$. *Under Assumptions 1 and 2, for any* $\mathbf{x}^* \in \mathcal{K}$ *and* $j \in [r]$, *Algorithm 2 with* $K_\gamma = \gamma$ *ensures*

$$\sum_{t=q_j}^{q_{j+1}-1} f_t(\mathbf{x}_t^\gamma) - \sum_{t=q_j}^{q_{j+1}-1} f_t(\mathbf{x}^*) \leq 4GD\sqrt{\gamma}.$$

It is not hard to bound a_i using Lemma 1 as below

$$a_i = \sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}_t^\gamma) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}) \leq 4GD\sqrt{\gamma}. \quad (5)$$

To upper bound b_i, we follow the proof of Theorem 3 in Zhang et al. (2018) as below

$$\begin{aligned} b_i =& \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}) - \sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}_t^*) \\ \leq& \sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}_{q_i}^*) - \sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}_t^*) \\ \leq& \gamma \max_{t\in[q_i,q_{i+1}-1]} \left(f_t(\mathbf{x}_{q_i}^*) - f_t(\mathbf{x}_t^*)\right). \end{aligned} \quad (6)$$

For brevity, let

$$V_T(i) = \sum_{t=q_i+1}^{q_{i+1}-1} \max_{\mathbf{x}\in\mathcal{K}} |f_t(\mathbf{x}) - f_{t-1}(\mathbf{x})|.$$

Then, for any $t \in [q_i, q_{i+1}-1]$, we have

$$\begin{aligned} & f_t(\mathbf{x}_{q_i}^*) - f_t(\mathbf{x}_t^*) \\ =& f_t(\mathbf{x}_{q_i}^*) - f_{q_i}(\mathbf{x}_{q_i}^*) + f_{q_i}(\mathbf{x}_{q_i}^*) - f_t(\mathbf{x}_t^*) \\ \leq& f_t(\mathbf{x}_{q_i}^*) - f_{q_i}(\mathbf{x}_{q_i}^*) + f_{q_i}(\mathbf{x}_t^*) - f_t(\mathbf{x}_t^*) \\ \leq& 2V_T(i). \end{aligned} \quad (7)$$

Combining (7) with (6), we have

$$\min_{\mathbf{x}\in\mathcal{K}} \sum_{t=q_i}^{q_{i+1}-1} f_t(\mathbf{x}) - \sum_{t=q_i}^{q_{i+1}-1} \min_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x}) \leq 2\gamma V_T(i). \quad (8)$$

Substituting (5) and (8) into (4), we have

$$\begin{aligned} \sum_{t=1}^{T} f_t(\mathbf{x}_t^\gamma) - \sum_{t=1}^{T} \min_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x}) \leq& 4rGD\sqrt{\gamma} + 2\gamma \sum_{i=1}^{r} V_T(i) \\ \leq& \frac{8TGD}{\sqrt{\gamma}} + 2\gamma V_T \end{aligned}$$

where the last inequality is due to $r \leq \frac{T}{\gamma} + 1 \leq \frac{2T}{\gamma}$ and $\sum_{i=1}^{r} V_T(i) \leq V_T$.

Proof of Theorem 2

Similar to the proof of Theorem 1, we introduce the following lemma, which gives the static regret bound of Algorithm 2 over each interval in (3) for strongly convex functions.

Lemma 2 *Let* $r = \lceil T/\gamma \rceil$, $q_i = (i-1)\gamma+1$ *for* $i = 1, \cdots, r$ *and* $q_{r+1} = T+1$. *If each* $f_t(\mathbf{x})$ *is* λ*-strongly convex and Assumptions 1 and 2 hold, for any* $\mathbf{x}^* \in \mathcal{K}$ *and* $j \in [r]$, *Algorithm 2 with* $K_\gamma = \gamma^2$ *ensures*

$$\begin{aligned} & \sum_{t=q_j}^{q_{j+1}-1} f_t(\mathbf{x}_t^\gamma) - \sum_{t=q_j}^{q_{j+1}-1} f_t(\mathbf{x}^*) \\ \leq& \frac{\lambda D^2}{2} + 2(G+\lambda D)D + \frac{2(G+\lambda D)^2 \ln(\gamma+1)}{\lambda}. \end{aligned}$$

Let $r = \lceil T/\gamma \rceil$, $q_i = (i-1)\gamma + 1$ for $i = 1, \cdots, r$ and $q_{r+1} = T+1$. It is not hard to verify that (4) and (8) still hold. So, we only need to bound a_i in (4) using Lemma 2 as below

$$a_i \leq c_1 + c_2 \ln(\gamma+1). \quad (9)$$

Then, substituting (9) and (8) into (4), we have

$$\begin{aligned} & \sum_{t=1}^{T} f_t(\mathbf{x}_t^\gamma) - \sum_{t=1}^{T} \min_{\mathbf{x}\in\mathcal{K}} f_t(\mathbf{x}) \\ \leq& r(c_1 + c_2\ln(\gamma+1)) + 2\gamma \sum_{i=1}^{r} V_T(i) \\ \leq& \frac{2T(c_1 + c_2\ln(\gamma+1))}{\gamma} + 2\gamma V_T \end{aligned}$$

where the last inequality is due to $r \leq \frac{T}{\gamma} + 1 \leq \frac{2T}{\gamma}$ and $\sum_{i=1}^{r} V_T(i) \leq V_T$.

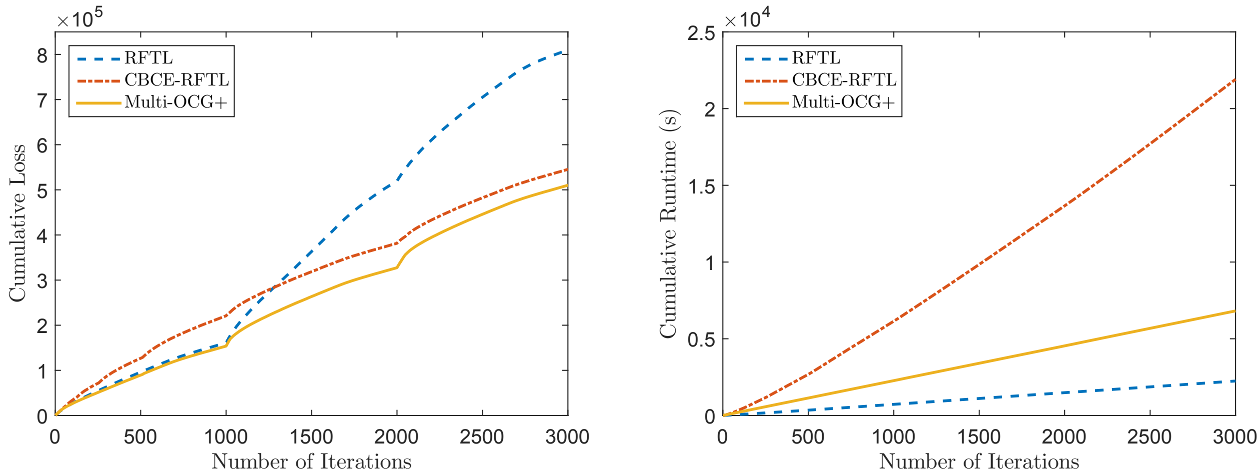
Experiments

In this section, we perform numerical experiments in dynamic environments to verify the efficiency and effectiveness of our Multi-OCG+. All algorithms are implemented wtih Matlab R2016b and tested on a linux machine with 2.4GHz CPU and 768GB RAM.

Settings and Datasets

Following Hazan and Kale (2012), we consider the online matrix completion problem, the goal of which is to construct a low-rank matrix $X \in \mathbb{R}^{m\times n}$ that can approximate a matrix $M \in \mathbb{R}^{m\times n}$ by observing its entries according to the online setting. At each round t, the learner first chooses a matrix X such that $\|X\|_* \leq r$, where $\|X\|_*$ denotes the trace norm of X and r is a constant. Then, the learner receives a loss function $f_t(X) = \sum_{(i,j)\in \mathrm{OB}_t} |X_{ij} - M_{ij}|$ where $\mathrm{OB}_t \subset [m] \times [n]$. We use a publicly available dataset—MovieLens 100K[1], which originally contains 100000 ratings in $\{1, 2, 3, 4, 5\}$ by 943 users on 1682 movies and can be denoted as $\{(i_k, j_k, M_{i_k j_k})\}_{k=1}^{100000}$. To create dynamic environments, we slightly modify the dataset such that the optimal low-rank approximation matrix changes at certain rounds. To be precise, we construct a larger dataset denoted as $\{(i_k, j_k, M_{i_k j_k})\}_{k=1}^{300000}$ by combining three copies of the original MovieLens 100K and flipping the original value of $M_{i_k j_k}$ by multiplying -1 for any $k = 100001, \cdots, 200000$. For simplicity, this dataset is equally divided into $T = 3000$

[1]https://grouplens.org/datasets/movielens/100k/



(a) Comparison of cumulative loss

(b) Comparison of cumulative runtime

Figure 1: Experimental results for online matrix completion in dynamic environments

partitions according to its original sequence, and we denote the set of (i, j) in the t-th partition as OB_t. In this way, the optimal low-rank approximation matrix will change after each 1000 rounds. Moreover, we set $r = 5000$, following Hazan and Kale (2012).

Baselines and Results

Note that for any $t = 1, \cdots, T$, $f_t(X)$ is not strongly convex. So, we first compare our Multi-OCG+ against RFTL (Hazan 2016) to demonstrate that simply running an algorithm with optimal static regret cannot deal with dynamic environments. Specifically, RFTL updates as

$$\mathbf{x}_{t+1} = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \left\{ \eta \sum_{i=1}^{t} \nabla f_i(\mathbf{x}_i)^\top \mathbf{x} + \|\mathbf{x} - \mathbf{x}_1\|_2^2 \right\}$$

where η is a parameter and we set $\eta = c/\sqrt{T}$ by tuning the constant c from the set $\{0.1, 1.0, \cdots, 1e6\}$. To further verify the performance of our Multi-OCG+, we also compare it against the projection-based algorithm proposed by Zhang et al. (2018), which achieves the existing best dynamic regret bound without any prior knowledge of environment changing. Similar as the framework of our Multi-OCG+, their algorithm also consists of two parts:

- A set of expert $\{E^I | I \in \mathcal{I}\}$, each of which is an instance of the expert-algorithm running over an interval $I \in \mathcal{I}$, where $\mathcal{I}$ is the geometric covering intervals proposed by Daniely, Gonen, and Shalev-Shwartz (2015);
- A meta-algorithm—CBCE (Jun et al. 2017), which is able to combine the decisions generated by active experts in each round.

We note that the expert-algorithm could be any online algorithm with an $O(\sqrt{T})$ static regret bound. In our experiments, we select RFTL as the expert-algorithm, and denote this algorithm as CBCE-RFTL. Specifically, CBCE-RFTL contains two parameters including the prior distribution $\boldsymbol{\pi} \in \Delta^{|\mathcal{I}|}$ over all experts and the learning rate η_I for each expert E^I. Following Jun et al. (2017), we set $\boldsymbol{\pi}$ as the uniform distribution, and set $\eta_I = c/\sqrt{|I|}$, where c is selected from $\{0.1, 1.0, \cdots, 1e6\}$. For our Multi-OCG+, we set $\mathcal{H} = \left\{\gamma_i = 2^i | i = 0, \cdots, \lfloor \log_2(T) \rfloor\right\}$. Since $f_t(X)$ is not strongly convex, the parameter τ is set to be $s/\sqrt{T}$, where s is selected from $\{1e-4, 1e-3, \cdots, 1.0\}$. Besides, the parameter η_γ of each expert E^γ is set to be $c/\sqrt{\gamma}$, where c is selected from $\{0.1, 1.0, \cdots, 1e6\}$. Although in theory we may need to set $K_\gamma = \gamma$ for our Multi-OCG+ to achieve an optimal dynamic regret bound for convex loss functions, we find that Multi-OCG+ with a much smaller K can also achieve good performance in our experiments. Therefore, we simply set $K_\gamma = 4$ for our Multi-OCG+ to reduce the time cost.

Figure 1 shows the cumulative loss and runtime of each algorithm in dynamic environments. Obviously, the performance of RFTL becomes worse after the environment changes, which shows that RFTL cannot deal with dynamic environments. By contrast, CBCE-RFTL and our Multi-OCG+ catch up with changing environments very fast. Furthermore, our Multi-OCG+ outperforms CBCE-RFTL, and is significantly faster than it, which verifies the advantages of our algorithm in the dynamic regret and time cost.

Conclusion

In this paper, we propose a projection-free online algorithm named as Multi-OCG+ to minimize the dynamic regret without the smoothness. According to theoretical analysis, our Multi-OCG+ enjoys an optimal dynamic regret bound of $O(\max\{T^{2/3}V_T^{1/3}, \sqrt{T}\})$ for convex functions, which does not require any prior knowledge of V_T. Furthermore, for strongly convex functions, Multi-OCG+ achieves a nearly optimal dynamic regret bound of $O(\max\{\sqrt{TV_T \log T}, \log T\})$. Experiments in dynamic environments demonstrate the efficiency and effectiveness of our Multi-OCG+.

Acknowledgments

This work was partially supported by the NSFC (61976112), JiangsuSF (BK20200064), and Open Research Projects of Zhejiang Lab (NO. 2021KB0AB02).

References

Abernethy, J. D.; Bartlett, P. L.; Rakhlin, A.; and Tewari, A. 2008. Optimal stragies and minimax lower bounds for online convex games. In *Proceedings of the 21st Annual Conference on Learning Theory*, 415–424.

Besbes, O.; Gur, Y.; and Zeevi, A. 2015. Non-stationary stochastic optimization. *Operations Research* 63(5): 1227–1244.

Boyd, S.; and Vandenberghe, L. 2004. *Convex Optimization*. Cambridge University Press.

Cesa-Bianchi, N.; and Lugosi, G. 2006. *Prediction, Learning, and Games*. Cambridge University Press.

Chen, L.; Harshaw, C.; Hassani, H.; and Karbasi, A. 2018. Projection-free online optimization with stochastic gradient: From convexity to submodularity. In *Proceedings of the 35th International Conference on Machine Learning*, 813–822.

Chen, L.; Zhang, M.; and Karbasi, A. 2019. Projection-free bandit convex optimization. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, 2047–2056.

Daniely, A.; Gonen, A.; and Shalev-Shwartz, S. 2015. Strongly adaptive online learning. In *Proceedings of the 32nd International Conference on Machine Learning*, 1405–1411.

Frank, M.; and Wolfe, P. 1956. An algorithm for quadratic programming. *Naval Research Logistics Quarterly* 3(1–2): 95–110.

Garber, D.; and Hazan, E. 2016. A linearly convergent conditional gradient algorithm with applications to online and stochastic optimization. *SIAM Journal on Optimization* 26(3): 1493–1528.

Hall, E. C.; and Willett, R. M. 2013. Dynamical models and tracking regret in online convex programming. In *Proceedings of the 30th International Conference on Machine Learning*, 579–587.

Hazan, E. 2016. Introduction to online convex optimization. *Foundations and Trends in Optimization* 2(3–4): 157–325.

Hazan, E.; Agarwal, A.; and Kale, S. 2007. Logarithmic regret algorithms for online convex optimization. *Machine Learning* 69(2): 169–192.

Hazan, E.; and Kale, S. 2012. Projection-free online learning. In *Proceedings of the 29th International Conference on Machine Learning*, 1843–1850.

Hazan, E.; and Minasyan, E. 2020. Faster projection-free online learning. In *Proceedings of the 33rd Annual Conference on Learning Theory*, 1877–1893.

Jadbabaie, A.; Rakhlin, A.; Shahrampour, S.; and Sridharan, K. 2015. Online optimization: Competing with dynamic comparators. In *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics*, 398–406.

Jaggi, M. 2013. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning*, 427–435.

Jun, K.-S.; Orabona, F.; Wright, S.; and Willett, R. 2017. Improved strongly adaptive online learning using coin betting. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics*, 943–951.

Kalhan, D. S.; Bedi, A. S.; Koppel, A.; Rajawat, K.; Gupta, A.; and Banerjee, A. 2020. Projection free dynamic online learning. In *Proceedings of the 45th IEEE International Conference on Acoustics, Speech and Signal Processing*, 3957–3961.

Kalhan, D. S.; Bedi, A. S.; Koppel, A.; Rajawat, K.; Hassani, H.; Gupta, A.; and Banerjee, A. 2019. Dynamic online learning via Frank-Wolfe algorithm.

Levy, K. Y.; and Krause, A. 2019. Projection free online learning over smooth sets. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, 1458–1466.

Mokhtari, A.; Shahrampour, S.; Jadbabaie, A.; and Ribeiro, A. 2016. Online optimization in dynamic environments: Improved regret rates for strongly convex problems. In *Proceedings of 55th Conference on Decision and Control*, 7195–7201.

Shalev-Shwartz, S. 2011. Online learning and online convex optimization. *Foundations and Trends in Machine Learning* 4(2): 107–194.

Shalev-Shwartz, S.; and Singer, Y. 2007. A primal-dual perspective of online learning algorithm. *Machine Learning* 69(2–3): 115–142.

van Erven, T.; and Koolen, W. M. 2016. MetaGrad: Multiple learning rates in online learning. In *Advances in Neural Information Processing Systems 29*, 3666–3674.

Wan, Y.; Tu, W.-W.; and Zhang, L. 2020. Projection-free distributed online convex optimization with $O(\sqrt{T})$ communication complexity. In *Proceedings of the 37th International Conference on Machine Learning*, 9818–9828.

Wan, Y.; and Zhang, L. 2021. Projection-free online learning over strongly convex sets. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*.

Wang, G.; Lu, S.; Hu, Y.; and Zhang, L. 2020. Adapting to smoothness: A more universal algorithm for online convex optimization. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, 6162–6169.

Wang, G.; Lu, S.; and Zhang, L. 2019. Adaptivity and optimality: A universal algorithm for online convex optimization. In *Proceedings of 35th Conference on Uncertainty in Artificial Intelligence*.

Xie, J.; Shen, Z.; Zhang, C.; Wang, B.; and Qian, H. 2020. Efficient projection-free online methods with stochastic recursive gradient. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence*, 6446–6453.

Yang, T.; Zhang, L.; Jin, R.; and Yi, J. 2016. Tracking slowly moving clairvoyant: Optimal dynamic regret of online learning with true and noisy gradient. In *Proceedings of the 33rd International Conference on Machine Learning*.

Zhang, L. 2020. Online learning in changing environments. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence*, 5178–5178.

Zhang, L.; Lu, S.; and Yang, T. 2020. Minimizing dynamic regret and adaptive regret simultaneously. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, 309–319.

Zhang, L.; Lu, S.; and Zhou, Z.-H. 2018. Adaptive online learning in dynamic environments. In *Advances in Neural Information Processing Systems 31*, 1323–1333.

Zhang, L.; Yang, T.; Jin, R.; and Zhou, Z.-H. 2018. Dynamic regret of strongly adaptive methods. In *Proceedings of the 35th International Conference on Machine Learning*, 5877–5886.

Zhang, L.; Yang, T.; Yi, J.; Jin, R.; and Zhou, Z.-H. 2017a. Improved dynamic regret for non-degenerate functions. In *Advances in Neural Information Processing Systems 30*, 732–741.

Zhang, W.; Zhao, P.; Zhu, W.; Hoi, S. C. H.; and Zhang, T. 2017b. Projection-free distributed online learning in networks. In *Proceedings of the 34th International Conference on Machine Learning*, 4054–4062.

Zinkevich, M. 2003. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning*, 928–936.