
Regret and Cumulative Constraint Violation Analysis for Online Convex Optimization with Long Term Constraints

Xinlei Yi¹ Xiuxian Li² Tao Yang³ Lihua Xie⁴ Tianyou Chai³ Karl H. Johansson¹

Abstract

This paper considers online convex optimization with long term constraints, where constraints can be violated in intermediate rounds, but need to be satisfied in the long run. The cumulative constraint violation is used as the metric to measure constraint violations, which excludes the situation that strictly feasible constraints can compensate the effects of violated constraints. A novel algorithm is first proposed and it achieves an $\mathcal{O}(T^{\max\{c, 1-c\}})$ bound for static regret and an $\mathcal{O}(T^{(1-c)/2})$ bound for cumulative constraint violation, where $c \in (0, 1)$ is a user-defined trade-off parameter, and thus has improved performance compared with existing results. Both static regret and cumulative constraint violation bounds are reduced to $\mathcal{O}(\log(T))$ when the loss functions are strongly convex, which also improves existing results. In order to achieve the optimal regret with respect to any comparator sequence, another algorithm is then proposed and it achieves the optimal $\mathcal{O}(\sqrt{T(1+P_T)})$ regret and an $\mathcal{O}(\sqrt{T})$ cumulative constraint violation, where P_T is the path-length of the comparator sequence. Finally, numerical simulations are provided to illustrate the effectiveness of the theoretical results.

1. Introduction

Online convex optimization is a promising learning framework for modeling sequential tasks and has important appli-

¹School of Electrical Engineering and Computer Science, and Digital Futures, KTH Royal Institute of Technology, Stockholm, Sweden ²Department of Control Science and Engineering, College of Electronics and Information Engineering, Institute for Advanced Study, and Shanghai Research Institute for Intelligent Autonomous Systems, Tongji University, Shanghai, China ³State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang, China ⁴School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. Correspondence to: Tao Yang <yangtao@mail.neu.edu.cn>.

cations in online binary classification (Crammer et al., 2006), online display advertising (Goldfarb & Tucker, 2011), etc. It has been studied since the 1990's (Cesa-Bianchi et al., 1996; Gentile & Warmuth, 1999; Gordon, 1999; Zinkevich, 2003; Hazan et al., 2007; Agarwal et al., 2010; Shalev-Shwartz, 2012; Jadbabaie et al., 2015; Hazan, 2016; Zhang et al., 2018b; 2020; Zhang, 2020). Online convex optimization can be understood as a repeated game between a learner and an adversary (Shalev-Shwartz, 2012). At round t of the game, the learner selects a point x_t from a known closed convex set $\mathcal{X} \subseteq \mathbb{R}^p$, and the adversary chooses a convex loss function $f_t : \mathcal{X} \rightarrow \mathbb{R}$. After that, the loss function f_t is revealed to the learner who suffers a loss $f_t(x_t)$. Note that at each round the loss function can be arbitrarily chosen by the adversary, especially with no probabilistic model imposed on the choices. This is the key difference between online and stochastic convex optimization. The goal of the learner is to choose a sequence $x_{[T]} = (x_1, \dots, x_T)$ such that her regret

$$\text{Reg}(x_{[T]}, y_{[T]}) := \sum_{t=1}^T f_t(x_t) - \sum_{t=1}^T f_t(y_t) \quad (1)$$

is minimized, where T is the total number of rounds and $y_{[T]} = (y_1, \dots, y_T)$ is a comparator sequence. In the literature, there are two commonly used comparator sequences. One is the optimal dynamic decision sequence $y_{[T]} = x_{[T]}^* = (x_1^*, \dots, x_T^*)$ solving the following constrained convex optimization problem when the sequence of loss functions is known a priori:

$$\min_{x_{[T]} \in \mathcal{X}^T} \sum_{t=1}^T f_t(x_t).$$

In this case, $\text{Reg}(x_{[T]}, x_{[T]}^*)$ is called the dynamic regret. The other comparator sequence is the optimal static decision sequence $y_{[T]} = \tilde{x}_{[T]}^* = (\tilde{x}_1^*, \dots, \tilde{x}_T^*)$, where \tilde{x}_T^* is the optimal static decision solving

$$\min_{x \in \mathcal{X}} \sum_{t=1}^T f_t(x).$$

In this case, $\text{Reg}(x_{[T]}, \tilde{x}_{[T]}^*)$ is called the static regret. In online convex optimization, we are usually interested in finding an upper bound on the worst case regret of an algorithm.

Intuitively, an algorithm performs well if its static regret is sublinear as a function of T , since this implies that on the average the algorithm performs as well as the best fixed strategy in hindsight as T goes to infinity (Shalev-Shwartz, 2012; Hazan, 2016).

It is known that the simple and popular projection-based online gradient descent algorithm

$$x_{t+1} = \mathcal{P}_{\mathcal{X}}(x_t - \alpha \nabla f_t(x_t)), \quad (2)$$

where $\mathcal{P}_{\mathcal{X}}(\cdot)$ is the projection onto the closed convex set \mathcal{X} and $\alpha > 0$ is the stepsize, achieves an $\mathcal{O}(\sqrt{T})$ static regret bound for loss functions with bounded subgradients (Zinkevich, 2003), i.e.,

$$\text{Reg}(x_{[T]}, \tilde{x}_{[T]}^*) = \mathcal{O}(\sqrt{T}).$$

It was later shown that $\mathcal{O}(\sqrt{T})$ is a tight bound up to constant factors (Hazan et al., 2007). The static regret bound can be reduced under more stringent strong convexity conditions on the loss functions (Hazan et al., 2007; Shalev-Shwartz, 2012; Hazan, 2016). When the feasible region is bounded, it was also shown in Zinkevich (2003) that the algorithm (2) achieves the following regret bound

$$\text{Reg}(x_{[T]}, y_{[T]}) = \mathcal{O}(\sqrt{T}(1 + P_T)),$$

where

$$P_T = \sum_{t=1}^{T-1} \|y_{t+1} - y_t\|$$

is the path-length of the comparator sequence $y_{[T]}$. By running the projection-based online gradient descent algorithm (2) $\mathcal{O}(\log(T))$ times in parallel with different stepsizes and choosing the smallest regret through an expert-tracking algorithm, the optimal regret bound

$$\text{Reg}(x_{[T]}, y_{[T]}) = \mathcal{O}(\sqrt{T(1 + P_T)})$$

was achieved in Zhang et al. (2018a). The curvature of loss functions, such as strong convexity and smoothness, can be used to reduce the regret bound (Mokhtari et al., 2016; Zhang et al., 2017; 2019; Zhao et al., 2020; Zhao & Zhang, 2020).

Despite the simplicity of the algorithm (2), its computational cost is crucial for its applicability. The projection $\mathcal{P}_{\mathcal{X}}(\cdot)$ is easy to compute and even has a closed form solution when \mathcal{X} is a simple set, e.g., a box or a ball. However, in practice, the constraint set \mathcal{X} is often complex. For example, if \mathcal{X} is characterized by inequalities as $\mathcal{X} = \{x : g(x) \leq \mathbf{0}_m, x \in \mathbb{X}\}$, where $\mathbb{X} \subseteq \mathbb{R}^p$ is a closed convex set and $g(x) = (g_1(x), \dots, g_m(x))^{\top}$ with each $g_i : \mathbb{R}^p \rightarrow \mathbb{R}$ being a convex function, then the projection $\mathcal{P}_{\mathcal{X}}(\cdot)$ yields a heavy computational burden. To tackle

this challenge, online convex optimization with long term constraints was considered in Mahdavi et al. (2012). In this case, instead of requiring $g(x_t) \leq \mathbf{0}_m$ at each round, the constraint should only be satisfied in the long run. More specifically, the constraint violation

$$\left\| \left[\sum_{t=1}^T g(x_t) \right]_+ \right\| \quad (3)$$

should grow sublinearly, where $[\cdot]_+$ is the projection onto the nonnegative space. In other words, the learner is allowed sometimes to make decisions that do not belong to the set \mathcal{X} , but the overall sequence of chosen decisions must obey the constraint at the end by a vanishing convergence rate. This problem is normally solved by online primal-dual algorithms (Mahdavi et al., 2012; Jenatton et al., 2016; Yuan & Lamperski, 2018; Yu & Neely, 2020). The problem can be extended to the case where the constraint functions are time-varying and revealed to the learner after her decision is chosen (Sun et al., 2017; Chen et al., 2017; Neely & Yu, 2017; Yu et al., 2017). The problem can also be extended to distributed settings (Li et al., 2020; Yi et al., 2020a;b; 2021; Yuan et al., 2021a;b).

The constraint violation metric defined in (3) allows inequality constraint violations at many rounds as long as they are compensated by a strictly feasible constraint that has a large margin, since it takes the summation over rounds before the projection operation $[\cdot]_+$. In this way, although the constraint violation grows sublinearly, the constraints may not be satisfied at many time instances, which will restrict the theoretical results only to applications where the constraints have cumulative nature. This motivates researchers to consider stricter forms of constraint violation metric. In Yuan & Lamperski (2018), the cumulative constraint violation

$$\left\| \sum_{t=1}^T [g(x_t)]_+ \right\| \quad (4)$$

and the cumulative squared constraint violation

$$\sum_{t=1}^T \|[g(x_t)]_+\|^2 \quad (5)$$

are considered. Both forms of metric (4) and (5) take into account all constraints that are not satisfied, thus both are stricter than the constraint violation metric defined in (3). In this paper, we consider a variant of cumulative constraint violation

$$\sum_{t=1}^T \|[g(x_t)]_+\|, \quad (6)$$

which is equivalent to metric (4) due to

$$\left\| \sum_{t=1}^T [g(x_t)]_+ \right\| \leq \sum_{t=1}^T \|[g(x_t)]_+\| \leq \sum_{t=1}^T \|[g(x_t)]_+\|_1$$

Table 1: Comparison of this paper to related works on online convex optimization with long term constraints.

Reference	Loss functions	Static regret	Regret	Constraint violation	Cumulative constraint violation
(Mahdavi et al., 2012)	Convex	$\mathcal{O}(\sqrt{T})$	Not given	$\mathcal{O}(T^{3/4})$	Not given
(Jenatton et al., 2016)	Convex	$\mathcal{O}(T^{\max\{c, 1-c\}})$	Not given	$\mathcal{O}(T^{1-c/2})$	Not given
	Strongly convex	$\mathcal{O}(T^c)$			
(Yuan & Lamperski, 2018)	Convex	$\mathcal{O}(T^{\max\{c, 1-c\}})$	Not given	$\mathcal{O}(T^{1-c/2})$	
	Strongly convex	$\mathcal{O}(\log(T))$		$\mathcal{O}(\sqrt{\log(T)T})$	
(Yu & Neely, 2020)	Convex	$\mathcal{O}(\sqrt{T})$	Not given	$\mathcal{O}(T^{1/4})^1$	Not given
Algorithm 1	Convex	$\mathcal{O}(T^{\max\{c, 1-c\}})$	$\mathcal{O}(\sqrt{T}(1+P_T))$	$\mathcal{O}(T^{(1-c)/2})$	
	Strongly convex	$\mathcal{O}(\log(T))$	Not given	$\mathcal{O}(\log(T))$	
Algorithm 2	Convex	$\mathcal{O}(T^{\max\{c, 1-c\}})$	$\mathcal{O}(\sqrt{T}(1+P_T))$	$\mathcal{O}(\sqrt{T})$	

$$= \left\| \sum_{t=1}^T [g(x_t)]_+ \right\|_1 \leq \sqrt{m} \left\| \sum_{t=1}^T [g(x_t)]_+ \right\|.$$

Moreover, it is straightforward to see that the cumulative constraint violation (6) is stricter than the cumulative squared constraint violation (5) when the constraint functions are bounded.

Contributions: This paper first proposes a novel algorithm (Algorithm 1) for the problem of online convex optimization with long term constraints. This algorithm achieves an $\mathcal{O}(T^{\max\{c, 1-c\}})$ static regret bound and an $\mathcal{O}(T^{(1-c)/2})$ cumulative constraint violation bound, where $c \in (0, 1)$ is a user-defined trade-off parameter, and hence yields improved performance compared with the results in Mahdavi et al. (2012); Jenatton et al. (2016); Yuan & Lamperski (2018); Yu & Neely (2020). This algorithm is inspired by Yu & Neely (2020) and is the first to achieve a cumulative constraint violation bound strictly better than $\mathcal{O}(T^{3/4})$ while maintaining $\mathcal{O}(\sqrt{T})$ regret for convex loss functions. Both static regret and cumulative constraint violation bounds are reduced to $\mathcal{O}(\log(T))$ when the loss functions are strongly convex, which also improves the results in Jenatton et al. (2016); Yuan & Lamperski (2018). This algorithm is also the first to achieve a cumulative constraint violation bound strictly better than $\mathcal{O}(\sqrt{\log(T)T})$ while maintaining $\mathcal{O}(\log(T))$ regret for strongly convex loss functions.

In order to achieve the optimal regret with respect to any comparator sequence, another algorithm (Algorithm 2) is then proposed and it achieves the optimal $\mathcal{O}(\sqrt{T}(1+P_T))$ regret and an $\mathcal{O}(\sqrt{T})$ cumulative constraint violation. This algorithm is inspired by Zhang et al. (2018a). The basic

¹This bound is reduced to $\mathcal{O}(1)$ if the constraint functions satisfy the Slater condition.

idea of the second algorithm is to run the first algorithm multiple times in parallel, each with a different stepsize that is optimal for a specific path-length, and then to combine them with an expert-tracking algorithm. This algorithm is the first to avoid computing the projection $\mathcal{P}_{\mathcal{X}}(\cdot)$ by considering long term constraints while maintaining the optimal regret and sublinear cumulative constraint violation.

In summary, the presented results are significant theoretical developments compared to prior works. The comparison of this paper to related studies in the literature is summarized in Table 1. Specifically, Mahdavi et al. (2012) achieved an $\mathcal{O}(\sqrt{T})$ static regret bound and an $\mathcal{O}(T^{3/4})$ constraint violation bound. Jenatton et al. (2016) achieved an $\mathcal{O}(T^{\max\{c, 1-c\}})$ static regret bound and an $\mathcal{O}(T^{1-c/2})$ constraint violation bound, which generalized the results in Mahdavi et al. (2012), and the static regret bound was reduced to $\mathcal{O}(T^c)$ when the loss functions are strongly convex. Yuan & Lamperski (2018) achieved an $\mathcal{O}(T^{\max\{c, 1-c\}})$ static regret bound and an $\mathcal{O}(T^{1-c/2})$ cumulative constraint violation bound, which further generalized the results in Jenatton et al. (2016) by using the stricter constraint violation metric, and these two bounds were respectively reduced to $\mathcal{O}(\log(T))$ and $\mathcal{O}(\sqrt{\log(T)T})$ when the loss functions are strongly convex, which improved the results in Jenatton et al. (2016). Yu & Neely (2020) achieved an $\mathcal{O}(\sqrt{T})$ static regret bound and an $\mathcal{O}(T^{1/4})$ constraint violation bound, which improved the results in Mahdavi et al. (2012); Jenatton et al. (2016).

Outline: The rest of this paper is organized as follows. Section 2 formulates the considered problem. Section 3 proposes two algorithms to solve the problem and analyze their regret and cumulative constraint violation bounds. Section 4 gives numerical simulations. Finally, Section 5 concludes the paper and proofs are given in the supplementary mate-

rial.

Notations: All inequalities and equalities throughout this paper are understood componentwise. \mathcal{X}^T is the T -fold Cartesian product of a set \mathcal{X} . \mathbb{R}^p and \mathbb{R}_+^p stand for the set of p -dimensional vectors and nonnegative vectors, respectively. \mathbb{N}_+ denotes the set of all positive integers. $[T]$ represents the set $\{1, \dots, T\}$ for any positive integer T . $\|\cdot\|$ ($\|\cdot\|_1$) represents the Euclidean norm (1-norm) for vectors and the induced 2-norm (1-norm) for matrices. x^\top denotes the transpose of a vector or a matrix. $\langle x, y \rangle$ represents the standard inner product of two vectors x and y . $\mathbf{0}_m$ denotes the column zero vector with dimension m . $[z]_+$ represents the component-wise projection of a vector $z \in \mathbb{R}^p$ onto \mathbb{R}_+^p . $\lceil \cdot \rceil$ and $\lfloor \cdot \rfloor$ denote the ceiling and floor functions, respectively.

2. Problem Formulation

2.1. Basic Definitions

Definition 1. Let $f : \text{Dom} \rightarrow \mathbb{R}$ be a function, where the set $\text{Dom} \subset \mathbb{R}^p$. A vector $g \in \mathbb{R}^p$ is called a subgradient of function f at point $x \in \text{Dom}$ if

$$f(y) \geq f(x) + \langle g, y - x \rangle, \quad \forall y \in \text{Dom}. \quad (7)$$

Throughout this paper, we use $\partial f(x)$ to denote the subgradient of f at x . Similarly, for a vector function $\tilde{f} = (f_1, \dots, f_m)^\top : \text{Dom} \rightarrow \mathbb{R}^m$, its subgradient at point $x \in \text{Dom}$ is denoted as

$$\partial \tilde{f}(x) = \begin{bmatrix} (\partial f_1(x))^\top \\ (\partial f_2(x))^\top \\ \vdots \\ (\partial f_m(x))^\top \end{bmatrix} \in \mathbb{R}^{m \times p}.$$

Moreover, it is straightforward to check that $\partial[f(x)]_+$ is the subgradient of $[f]_+$ at x , where

$$\partial[f(x)]_+ = \begin{cases} \mathbf{0}_p, & \text{if } f(x) < 0 \\ \partial f(x), & \text{otherwise.} \end{cases}$$

2.2. Problem Formulation

This paper considers the problem of online convex optimization with long term constraints. Let $\mathbb{X} \subseteq \mathbb{R}^p$ be the constrained set and $g : \mathbb{X} \rightarrow \mathbb{R}^m$ be the constrained function, where p and m are positive integers. Both \mathbb{X} and g are known in advance. Suppose $\mathcal{X} = \{x : g(x) \leq \mathbf{0}_m, x \in \mathbb{X}\}$ is non-empty. Let $\{f_t : \mathbb{X} \rightarrow \mathbb{R}\}$ be a sequence of loss functions and each f_t is unknown until the end of round t . The goal of this paper is to propose online algorithms to choose $x_t \in \mathbb{X}$ for each round t such that both regret and cumulative constraint violation grow sublinearly with respect to the total number of rounds T .

We make the following standing assumptions on the loss and constraint functions.

Assumption 1. The set \mathbb{X} is convex and closed. The functions f_t and g are convex.

Assumption 2. There exists a positive constant F such that

$$|f_t(x) - f_t(y)| \leq F, \quad \|g(x)\| \leq F, \quad \forall t \in \mathbb{N}_+, x, y \in \mathbb{X}. \quad (8)$$

Assumption 3. The subgradients $\partial f_t(x)$ and $\partial g(x)$ exist. Moreover, they are uniformly bounded on \mathbb{X} , i.e., there exists a positive constant G such that

$$\|\partial f_t(x)\| \leq G, \quad \|\partial g(x)\| \leq G, \quad \forall t \in \mathbb{N}_+, x \in \mathbb{X}. \quad (9)$$

From (8), we know that the cumulative constraint violation (6) is stricter than the cumulative squared constraint violation (5) due to

$$\sum_{t=1}^T \|[g(x_t)]_+\|^2 \leq F \sum_{t=1}^T \|[g(x_t)]_+\|.$$

3. Main Results

In this section, we propose two novel algorithms for the constrained online convex optimization problem formulated in Section 2, and analyze their regret and cumulative constraint violation bounds.

3.1. The Basic Approach

The basic approach is summarized in Algorithm 1, which is inspired by Algorithm 1 proposed in Yu & Neely (2020). The key difference between Algorithm 1 and the algorithm proposed in Yu & Neely (2020) is that we use the clipped constraint function $[g]_+$ to replace the original constraint function g . With this modification, we can analyze constraint violations under stricter forms of metric. In addition to using the clipped constraint function, Algorithm 1 also has time-varying algorithm parameters. This enables us to consider the nontrivial extensions, such as strongly convex loss functions and the general dynamic regret, which have not been studied in Yu & Neely (2020).

In the following, we analyze regret and cumulative constraint violation bounds for Algorithm 1. We first provide regret and cumulative constraint violation bounds for the general cases in the following lemma.

Lemma 1. Suppose Assumptions 1–3 hold. Let $\{x_t\}$ be the sequence generated by Algorithm 1 with $\gamma_t = \gamma_0/\sqrt{\alpha_t}$, where $\gamma_0 \in (0, 1/(\sqrt{2}G)]$ is a constant. Then, for any comparator sequence $y_{[T]} \in \mathcal{X}^T$,

$$\text{Reg}(x_{[T]}, y_{[T]}) \leq \sum_{t=1}^T \Delta_t(y_t) + \sum_{t=1}^T \frac{G^2 \alpha_t}{2}, \quad (13)$$

Algorithm 1

Input: non-increasing sequence $\{\alpha_t\} \subseteq (0, +\infty)$ and non-decreasing sequence $\{\gamma_t\} \subseteq (0, +\infty)$.

Initialize: $q_0 = \mathbf{0}_m$ and $x_1 \in \mathbb{X}$.

for $t = 2, \dots$ **do**

Observe $\partial f_{t-1}(x_{t-1})$.

Update

$$q_{t-1} = q_{t-2} + \gamma_{t-1}[g(x_{t-1})]_+, \quad (10)$$

$$\hat{q}_{t-1} = q_{t-1} + \gamma_{t-1}[g(x_{t-1})]_+, \quad (11)$$

$$x_t = \arg \min_{x \in \mathbb{X}} \{ \alpha_{t-1} \langle \partial f_{t-1}(x_{t-1}), x \rangle + \alpha_{t-1} \gamma_t \langle \hat{q}_{t-1}, [g(x)]_+ \rangle + \|x - x_{t-1}\|^2 \}. \quad (12)$$

end for

Output: $\{x_t\}$.

$$\begin{aligned} & \sum_{t=1}^T \|[g(x_t)]_+\| \\ & \leq \sqrt{m} \left(\frac{1}{\gamma_T} \|q_T\| + \sum_{t=1}^{T-1} \left(\frac{1}{\gamma_t} - \frac{1}{\gamma_{t+1}} \right) \|q_t\| \right), \quad (14) \end{aligned}$$

$$\begin{aligned} & \frac{1}{2} \|q_{T+1}\|^2 \\ & \leq \sum_{t=1}^T \Delta_t(y_t) + \sum_{t=1}^T \frac{G^2 \alpha_t}{2} + \frac{1}{2} \gamma_1^2 F^2 + FT \\ & \quad + \sum_{t=1}^T (\gamma_{t+1} - \gamma_t)^2 F^2, \quad (15) \end{aligned}$$

where

$$\Delta_t(y_t) = \frac{1}{\alpha_t} (\|y_t - x_t\|^2 - \|y_t - x_{t+1}\|^2).$$

Proof: The proof is given in the supplementary material. ■

We then present the first main result.

Theorem 1. *Suppose Assumptions 1–3 hold. For any $T \in \mathbb{N}_+$, let $x_{[T]}$ be the sequence generated by Algorithm 1 with*

$$\alpha_t = \frac{\alpha_0}{T^c}, \quad \gamma_t = \frac{\gamma_0}{\sqrt{\alpha_t}}, \quad \forall t \in [T], \quad (16)$$

where $\alpha_0 > 0$, $c \in (0, 1)$, and $\gamma_0 \in (0, 1/(\sqrt{2}G)]$ are constants. Then,

$$\text{Reg}(x_{[T]}, \tilde{x}_{[T]}^*) = \mathcal{O}(T^{\max\{c, 1-c\}}), \quad (17)$$

$$\sum_{t=1}^T \|[g(x_t)]_+\| = \mathcal{O}(T^{(1-c)/2}). \quad (18)$$

Proof: The explicit expressions of the right-hand sides of (17)–(18), and the proof are given in the supplementary material. ■

Compared with the results that $\text{Reg}(x_{[T]}, \tilde{x}_{[T]}^*) = \mathcal{O}(T^{\max\{c, 1-c\}})$ and $\|[\sum_{t=1}^T g(x_t)]_+\| = \mathcal{O}(T^{1-c/2})$ achieved in Jenatton et al. (2016), from (17) and (18), we know that Algorithm 1 achieves the same static regret bound but a strictly smaller constraint violation bound under the stricter metric. Similarly, compared with the results that $\text{Reg}(x_{[T]}, \tilde{x}_{[T]}^*) = \mathcal{O}(T^{\max\{c, 1-c\}})$ and $\|[\sum_{t=1}^T g(x_t)]_+\| = \mathcal{O}(T^{1-c/2})$ achieved in Yuan & Lamperski (2018), from (17) and (18), we know that Algorithm 1 achieves the same static regret bound but a strictly smaller cumulative constraint violation bound. By setting $c = 0.5$ in Theorem 1, we have $\text{Reg}(x_{[T]}, \tilde{x}_{[T]}^*) = \mathcal{O}(\sqrt{T})$ and $\sum_{t=1}^T \|[g(x_t)]_+\| = \mathcal{O}(T^{1/4})$. Thus, the optimal $\mathcal{O}(\sqrt{T})$ regret bound achieved in Zinkevich (2003); Mahdavi et al. (2012); Yu & Neely (2020) is recovered. Moreover, the $\mathcal{O}(T^{1/4})$ constraint violation bound achieved in Yu & Neely (2020) is improved by using the stricter metric and the $\mathcal{O}(T^{3/4})$ constraint violation bound achieved in Mahdavi et al. (2012) is not only improved by using the stricter metric but also strictly reduced. However, the authors of Yu & Neely (2020) also showed that constraint violation bound is reduced to $\mathcal{O}(1)$ if the constraint functions satisfy the Slater condition. Noting this, it is natural to ask the question whether Algorithm 1 can achieve $\mathcal{O}(1)$ cumulative constraint violation bound if the Slater condition holds. Unfortunately, we have not found a way to show this. The reason is that the clipped constraint functions do not satisfy the Slater condition since they are always nonnegative.

If the loss functions f_t are strongly convex, then the static regret and cumulative constraint violation bounds achieved in Theorem 1 can be reduced. Moreover, the total number of rounds T is not needed.

Corollary 1. *Suppose Assumptions 1–3 hold. Moreover, for all $t \in \mathbb{N}_+$, $f_t(x)$ are strongly convex over \mathbb{X} , i.e., there exists a constant $\mu > 0$, such that for all $x, y \in \mathbb{X}$,*

$$f_t(x) \geq f_t(y) + \langle x - y, \partial f_t(y) \rangle + \mu \|x - y\|^2. \quad (19)$$

Let $\{x_t\}$ be the sequence generated by Algorithm 1 with

$$\alpha_t = \frac{1}{t\mu}, \quad \gamma_t = \frac{\gamma_0}{\sqrt{\alpha_t}}, \quad \forall t \in \mathbb{N}_+, \quad (20)$$

where $\gamma_0 \in (0, 1/(\sqrt{2}G)]$ is a constant. Then,

$$\text{Reg}(x_{[T]}, \tilde{x}_{[T]}^*) = \mathcal{O}(\log(T)), \quad (21)$$

$$\sum_{t=1}^T \|[g(x_t)]_+\| = \mathcal{O}(\log(T)). \quad (22)$$

Proof: The explicit expressions of the right-hand sides of (21)–(22), and the proof are given in the supplementary material. ■

Corollary 1 is the first to provide the $\mathcal{O}(\log(T))$ regret and cumulative constraint violation for online convex optimization with long term constraints when the loss functions are strongly convex. While $\mathcal{O}(\log(T))$ regret is well known for traditional online convex optimization without long term constraints (Hazan et al., 2007), whether similar bounds exist for online convex optimization with long term constraints is an open problem. Thus, Corollary 1 is a significant result which requires non-trivial analysis. Compared with the results that $\text{Reg}(x_{[T]}, \tilde{x}_{[T]}^*) = \mathcal{O}(T^c)$ and $\|\sum_{t=1}^T g(x_t)_+\| = \mathcal{O}(T^{1-c/2})$ achieved in Jenatton et al. (2016), from (21) and (22), we know that Algorithm 1 achieves strictly smaller static regret and constraint violation bounds under the stricter form of constraint violation metric. Similarly, compared with the results that $\text{Reg}(x_{[T]}, \tilde{x}_{[T]}^*) = \mathcal{O}(\log(T))$ and $\|\sum_{t=1}^T [g(x_t)]_+\| = \mathcal{O}(\sqrt{\log(T)T})$ achieved in Yuan & Lamperski (2018), from (21) and (22), we know that Algorithm 1 achieves the same static regret bound but a strictly smaller cumulative constraint violation bound.

To end this section, let us analyze the bound of general regret for Algorithm 1. If the set \mathbb{X} has bounded diameter, then we can show that Algorithm 1 achieves $\mathcal{O}(\sqrt{T}(1 + P_T))$ regret. Moreover, the total number of rounds T is not needed.

Assumption 4. *The set \mathbb{X} has bounded diameter, i.e., there is a positive constant $d(\mathbb{X})$ such that*

$$\|x - y\| \leq d(\mathbb{X}), \forall x, y \in \mathbb{X}. \quad (23)$$

Assumption 4 is commonly used in the literature, e.g., (Zinkevich, 2003; Mahdavi et al., 2012; Jenatton et al., 2016; Yuan & Lamperski, 2018; Zhang et al., 2018a).

Theorem 2. *Suppose Assumptions 1–4 hold. Let $\{x_t\}$ be the sequence generated by Algorithm 1 with*

$$\alpha_t = \frac{\alpha_0}{t^c}, \gamma_t = \frac{\gamma_0}{\sqrt{\alpha_t}}, \forall t \in \mathbb{N}_+, \quad (24)$$

where $\alpha_0 > 0$, $c \in (0, 1)$, and $\gamma_0 \in (0, 1/(\sqrt{2}G))$ are constants. Then,

$$\text{Reg}(x_{[T]}, y_{[T]}) = \mathcal{O}(T^{1-c} + T^c(1 + P_T)), \quad (25)$$

$$\sum_{t=1}^T \|[g(x_t)]_+\| = \mathcal{O}(T^{(1-c)/2}). \quad (26)$$

Proof: The explicit expressions of the right-hand sides of (25)–(26), and the proof are given in the supplementary material. ■

If the optimal static decision sequence is chosen as the comparator sequence, i.e., $y_{[T]} = \tilde{x}_{[T]}^*$, then $P_T = 0$. In this case, the results in Theorem 2 recover the results in Theorem 1. By setting $c = 0.5$ in Theorem 2, we have $\text{Reg}(x_{[T]}, y_{[T]}) = \mathcal{O}(\sqrt{T}(1 + P_T))$ and $\sum_{t=1}^T \|[g(x_t)]_+\| = \mathcal{O}(T^{1/4})$, which recover the regret bound achieved in Zinkevich (2003). If the path-length P_T is known in advance, then the regret bound can be reduced under the cost that the cumulative constraint violation bound is increased.

Corollary 2. *For any $T \in \mathbb{N}_+$, suppose the path-length P_T is known in advance. Under the same conditions stated in Theorem 2 with $\alpha_0 = (1 + P_T)^c$,*

$$\text{Reg}(x_{[T]}, y_{[T]}) = \mathcal{O}(T^c(1 + P_T)^{1-c} + T^{1-c}(1 + P_T)^c), \quad (27)$$

$$\sum_{t=1}^T \|[g(x_t)]_+\| = \mathcal{O}(T^{(1-c)/2}(1 + P_T)^{c/2}). \quad (28)$$

Proof: The explicit expressions of the right-hand sides of (27)–(28), and the proof are given in the supplementary material. ■

By setting $c = 0.5$ in Corollary 2, we have $\text{Reg}(x_{[T]}, y_{[T]}) = \mathcal{O}(\sqrt{T}(1 + P_T))$ and $\sum_{t=1}^T \|[g(x_t)]_+\| = \mathcal{O}(T^{1/4}(1 + P_T)^{1/4})$, which recover the optimal regret bound achieved in Zhang et al. (2018a). However, in Corollary 2 the path-length P_T needs to be known in advance, which is normally unknown in practice. Thus, Algorithm 1 cannot achieve the optimal $\mathcal{O}(\sqrt{T}(1 + P_T))$ regret bound in general. This motivates us to propose another algorithm such that it can achieve the optimal regret bound without using P_T , which is presented in the next section.

3.2. An Improved Approach

In this section, we propose another algorithm for the constrained online convex optimization problem formulated in Section 2, which achieves $\mathcal{O}(\sqrt{T}(1 + P_T))$ regret and $\mathcal{O}(\sqrt{T})$ cumulative constraint violation without using the path-length P_T to design the algorithm parameters.

The proposed algorithm is summarized in Algorithm 2, which is inspired by the expert-tracking algorithm (Improved Ader) proposed in Zhang et al. (2018a) as well as Algorithm 1. The basic idea of Algorithm 2 is to run Algorithm 1 multiple times in parallel, each with a different stepsize that is optimal for a specific path-length, and then to combine them with an expert-tracking algorithm. By setting $N = 1$ in Algorithm 2, we know that Algorithm 2 becomes Algorithm 1. Algorithm 2 is designed for online convex optimization with long term constraints, while the Improved

Ader algorithm in Zhang et al. (2018a) is for traditional online convex optimization without long term constraints. In other words, the main difference between Algorithm 2 and the Improved Ader algorithm is that we avoid computing the projection $\mathcal{P}_{\mathcal{X}}(\cdot)$ by considering long term constraints. Just as the algorithms in Mahdavi et al. (2012); Jenatton et al. (2016); Yuan & Lamperski (2018); Yu & Neely (2020) are the nontrivial extensions of the classic online gradient descent algorithm in Zinkevich (2003), Algorithm 2 is a nontrivial extension of the Improved Ader algorithm.

Algorithm 2

Input: parameters $N \in \mathbb{N}_+$ and $\beta > 0$; non-increasing sequence $\{\alpha_{i,t}\} \subseteq (0, +\infty)$ and non-decreasing sequence $\{\gamma_{i,t}\} \subseteq (0, +\infty)$, $\forall i \in [N]$.

Initialize: $q_{i,0} = \mathbf{0}_m$, $x_{i,1} \in \mathbb{X}$, $w_{i,1} = \frac{N+1}{i(i+1)N}$, $\forall i \in [N]$, and $x_1 = \sum_{i=1}^N w_{i,1}x_{i,1}$.

for $t = 2, \dots$ **do**

Observe $\partial f_{t-1}(x_{t-1})$.

Update

$$q_{i,t-1} = q_{i,t-2} + \gamma_{i,t-1}[g(x_{i,t-1})]_+, \quad (29)$$

$$\hat{q}_{i,t-1} = q_{i,t-1} + \gamma_{i,t-1}[g(x_{i,t-1})]_+, \quad (30)$$

$$\begin{aligned} x_{i,t} = \arg \min_{x \in \mathbb{X}} \{ & \alpha_{i,t-1} \langle \partial f_{t-1}(x_{t-1}), x \rangle \\ & + \alpha_{i,t-1} \gamma_{i,t} \langle \hat{q}_{i,t-1}, [g(x)]_+ \rangle \\ & + \|x - x_{i,t-1}\|^2 \}, \end{aligned} \quad (31)$$

$$l_{i,t-1} = \langle \partial f_{t-1}(x_{t-1}), x_{i,t-1} - x_{t-1} \rangle, \quad (32)$$

$$w_{i,t} = \frac{w_{i,t-1} e^{-\beta l_{i,t-1}}}{\sum_{i=1}^N w_{i,t-1} e^{-\beta l_{i,t-1}}}, \quad (33)$$

$$x_t = \sum_{i=1}^N w_{i,t} x_{i,t}. \quad (34)$$

end for

Output: $\{x_t\}$.

In Algorithm 2, for each $i \in [N]$, the equations (29)–(32) can be updated in parallel. Define the surrogate loss (van Erven & Koolen, 2016; Zhang et al., 2018a) as

$$\ell_t(x) = \langle \partial f_t(x_t), x - x_t \rangle, \quad \forall x \in \mathbb{X}. \quad (35)$$

It is straightforward to see that $\partial \ell_t(x_{i,t}) = \partial f_t(x_t)$. Thus, for each $i \in [N]$, the updating equations (29)–(31) are exactly (10)–(12) for solving constrained online convex optimization with loss functions $\{\ell_t(x)\}$.

The regret and cumulative constraint violation bounds for Algorithm 2 are provided in the following theorem.

Theorem 3. *Suppose Assumptions 1–4 hold. For any $T \in \mathbb{N}_+$, let $x_{[T]}$ be the sequence generated by Algorithm 2 with*

$$N = \lceil \kappa \log_2(1 + T) \rceil + 1, \quad \beta = \frac{\beta_0}{T^c},$$

$$\alpha_{i,t} = \frac{\alpha_0 2^{i-1}}{t^c}, \quad \gamma_{i,t} = \frac{\gamma_0}{\sqrt{\alpha_{i,t}}}, \quad \forall t \in [T], \quad (36)$$

where $\kappa \in [0, c]$, $c \in (0, 1)$, $\alpha_0 > 0$, $\beta_0 > 0$, and $\gamma_0 \in (0, 1/(\sqrt{2}G))$ are constants. Then,

$$\text{Reg}(x_{[T]}, y_{[T]}) = \mathcal{O}(T^c(1 + P_T)^{1-\kappa} + T^{1-c}(1 + P_T)^\kappa), \quad (37)$$

$$\sum_{t=1}^T \|[g(x_t)]_+\| = \mathcal{O}(T^{(1-c+\kappa)/2}). \quad (38)$$

Proof: The explicit expressions of the right-hand sides of (37)–(38), and the proof are given in the supplementary material. ■

The bounds presented in (37)–(38) still hold if choosing $\alpha_{i,t} = \alpha_0 2^{i-1}/T^c$ in (36). By setting $\kappa = c = 0.5$ in Theorem 3, we have $\text{Reg}(x_{[T]}, y_{[T]}) = \mathcal{O}(\sqrt{T}(1 + P_T))$ and $\sum_{t=1}^T \|[g(x_t)]_+\| = \mathcal{O}(\sqrt{T})$, which recover the optimal regret bound achieved in Zhang et al. (2018a) where long term constraints are not considered. Therefore, Theorem 3 is the first to achieve the optimal regret bound for online convex optimization with long term constraints without knowing the path-length P_T while maintaining sublinear cumulative constraint violation. Due to the long term constraints, the analysis is much more complicated. On the other hand, by comparing Corollary 2 and Theorem 3, we see that Algorithm 2 can achieve the optimal regret bound without knowing P_T under the cost that the (theoretical) cumulative constraint violation bound is increased.

4. Simulations

In this section, we illustrate and verify the proposed algorithms through numerical simulations.

4.1. Online Linear Programming

Similar to Yu & Neely (2020), we consider online convex optimization with linear loss functions $f_t(x) = \langle \theta_t, x \rangle$, where $\theta_t \in \mathbb{R}^p$ is time-varying and unknown at round t ; constraint set $\mathbb{X} \subseteq \mathbb{R}^p$; and constraint functions $Ax \leq b$, where $A \in \mathbb{R}^{m \times p}$ and $b \in \mathbb{R}^m$ are fixed and known in advance. This problem has broad applications in various areas including transportation, energy, telecommunications, and manufacturing.

In the simulations, similar to Yu & Neely (2020), we choose $p = 2$, $m = 3$, and $\mathbb{X} = [-1, 1]^2$. Components of A and b are uniformly distributed random numbers in the intervals $[0, 2]$ and $[0, 5]$, respectively. The total number of rounds is chosen as $T = 5000$. The time-varying loss coefficients are set as $\theta_t = \theta_{1,t} + \theta_{2,t} + \theta_{3,t}$, where components

Table 2: Parameters for each algorithm in online linear programming.

Algorithm	Parameters
Algorithm 1	$\alpha_t = 2/T^c, \gamma_t = T^{c/2}, c = 0.5$
Algorithm 2	$\alpha_{i,t} = 2/T^c, \beta = 3/T^c, \gamma_{i,t} = T^{c/2}, c = \kappa = 0.5$
Yu & Neely (2020)	$\alpha = \sqrt{T}/4, \gamma = T^{1/4}$
Yuan & Lamperski (2018)	$\sigma = 0.25, \eta = 1.5/T^c, c = 0.5$
Jenatton et al. (2016)	$\theta_t = 0.01/t^c, \eta_t = 0.7/t^c, \mu_t = 1/(t+1)/\theta_t, c = 0.5$
Mahdavi et al. (2012)	$\delta = 0.5, \eta = 0.8/\sqrt{T}$

of $\theta_{1,t}$ are uniformly distributed random numbers in the interval $[-t^{1/10}, t^{1/10}]$; components of $\theta_{2,t}$ are uniformly distributed random numbers in the interval $[-1, 0]$ when $t \in [1, 1500] \cup [2000, 3500] \cup [4000, 5000]$ and in the interval $[0, 1]$ otherwise; and $\theta_{3,t} = ((-1)^{\mu(t)}, (-1)^{\mu(t)})^\top$, where $\mu(t)$ is a random permutation of the integers from 1 to T .

We compare Algorithms 1 and 2 with state-of-the-art algorithms: Algorithm 1 in Yu & Neely (2020), Algorithm 1 in Yuan & Lamperski (2018), the algorithm in Jenatton et al. (2016), and Algorithm 1 in Mahdavi et al. (2012). Table 2 lists all the algorithm parameters used in the simulations². Figures 1 and 2 illustrate the evolutions of the cumulative loss $\sum_{t=1}^T f_t(x_t)$ and the cumulative constraint violation $\sum_{t=1}^T \|g(x_t)_+\|$ averaged over 1000 independent experiments generated from the above settings, respectively. Figure 1 shows that Algorithm 2 has the smallest cumulative loss and Algorithm 1 has smaller cumulative loss than other algorithms. Figure 2 shows that Algorithms 1 and 2 have almost the same cumulative constraint violation which is slightly smaller than that achieved by the algorithm in Yu & Neely (2020), and smaller than that achieved by the algorithm in Yuan & Lamperski (2018), and much smaller than that achieved by the algorithms in Jenatton et al. (2016); Mahdavi et al. (2012). Therefore, the simulation results are in accordance with the theoretical results summarized in Table 1.

4.2. Online Quadratic Programming

In order to verify the improved theoretical results for strongly convex functions, we replace the linear loss functions in the above simulations by quadratic loss functions. Specifically, we consider $f_t(x) = \|x - \theta_t\|^2 + 20\langle \theta_t, x \rangle$.

We compare Algorithm 1 with the algorithms in Yuan & Lamperski (2018); Jenatton et al. (2016) since the perfor-

²If we write Algorithm 1 and the algorithm in Yu & Neely (2020) in the same form, then we can see that they use the same algorithm parameters.

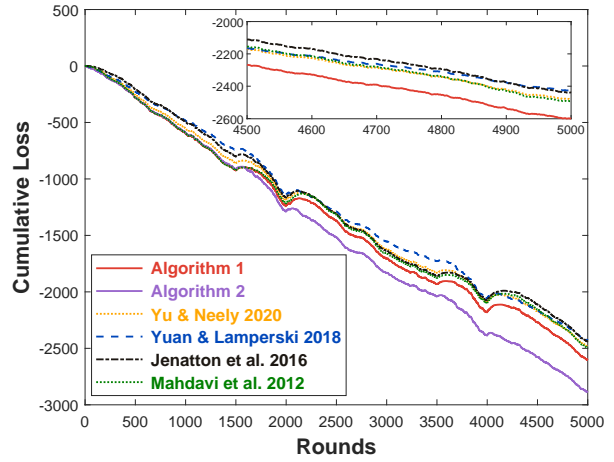


Figure 1: Evolutions of the cumulative loss in online linear programming.

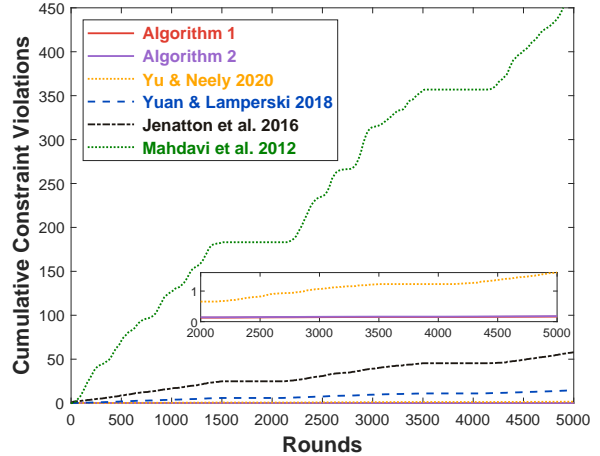


Figure 2: Evolutions of the cumulative constraint violation in online linear programming.

Table 3: Parameters for each algorithm in online quadratic programming.

Algorithm	Parameters
Algorithm 1	$\alpha_t = 6/t, \gamma_t = \sqrt{t}$
Yuan & Lamperski (2018)	$\theta_t = 4.5/(t + 1), \eta_t = 3/(t + 1)$
Jenatton et al. (2016)	$\theta_t = c/t^{0.01}, \eta_t = 2.5/t, \mu_t = 1/(t + 1)/\theta_t, c = 0.001$

mance of these algorithms under the strong convexity assumption has been analyzed. Table 3 lists all the algorithm parameters used in the simulations. Figures 3 and 4 illustrate the evolutions of the cumulative loss $\sum_{t=1}^T f_t(x_t)$ and the cumulative constraint violation $\sum_{t=1}^T \|[g(x_t)]_+\|$ averaged over 1000 independent experiments, respectively. Figure 3 shows that Algorithm 1 and the algorithm in Yuan & Lamperski (2018) have almost the same cumulative loss, which is smaller than that achieved by the algorithm in Jenatton et al. (2016). Figure 4 shows that Algorithm 1 has significant smaller cumulative constraint violation than that achieved by the algorithms in Yuan & Lamperski (2018); Jenatton et al. (2016). Therefore, the simulation results match the theoretical results summarized in Table 1.

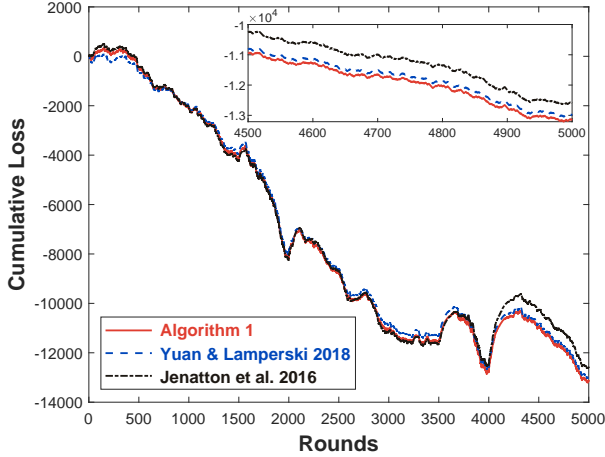


Figure 3: Evolutions of the cumulative loss in online quadratic programming.

5. Conclusions

In this paper, we proposed two algorithms for online convex optimization with long term constraints. We analyzed static regret and cumulative constraint violation bounds for the first algorithm when the loss functions are convex and strongly convex, respectively. We achieved improved performance compared with existing results in the sense that the cumulative constraint violation is a stricter metric than the commonly used constraint violation metric in the liter-

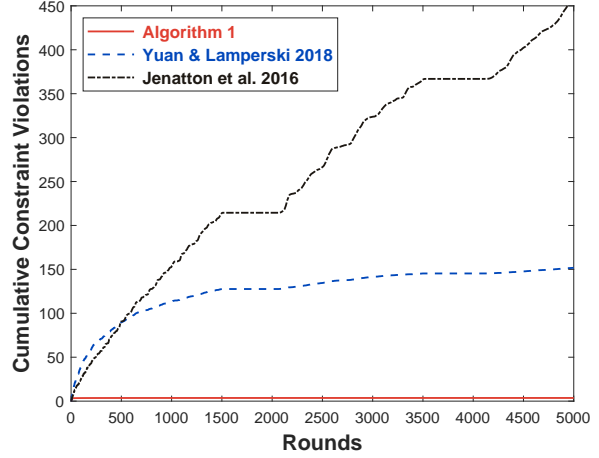


Figure 4: Evolutions of the cumulative constraint violation in online quadratic programming.

ature and smaller cumulative constraint violation bounds are achieved. We also analyzed regret with respect to any comparator sequence for both algorithms and the optimal $\mathcal{O}(\sqrt{T(1 + P_T)})$ regret can be achieved by the second algorithm. In the future, we will investigate how to use the curvature of loss functions to reduce the regret and cumulative constraint violation bounds.

Acknowledgements

The authors thank anonymous reviewers for the valuable comments and suggestions, and also thank the meta-reviewer for handling the review of this work.

This work was supported by Knut and Alice Wallenberg Foundation; Swedish Foundation for Strategic Research; Swedish Research Council; Ministry of Education of Republic of Singapore under Grant AcRF TIER 1- 2019-T1-001-088 (RG72/19); National Natural Science Foundation of China under Grants 61991403, 61991404, 61991400, and 62003243; 2020 Science and Technology Major Project of Liaoning Province under Grant 2020JH1/10100008; Shanghai Municipal Commission of Science and Technology under No. 19511132101; and Shanghai Municipal Science and Technology Major Project under No. 2021SHZDZX0100.

References

- Agarwal, A., Dekel, O., and Xiao, L. Optimal algorithms for online convex optimization with multi-point bandit feedback. In *Conference on Learning Theory*, pp. 28–40, 2010.
- Cesa-Bianchi, N., Long, P. M., and Warmuth, M. K. Worst-case quadratic loss bounds for prediction using linear functions and gradient descent. *IEEE Transactions on Neural Networks*, 7(3):604–619, 1996.
- Chen, T., Ling, Q., and Giannakis, G. B. An online convex optimization approach to proactive network resource allocation. *IEEE Transactions on Signal Processing*, 65(24):6350–6364, 2017.
- Crammer, K., Dekel, O., Keshet, J., Shalev-Shwartz, S., and Singer, Y. Online passive aggressive algorithms. *Journal of Machine Learning Research*, 7:551–585, 2006.
- Gentile, C. and Warmuth, M. K. Linear hinge loss and average margin. In *Advances in Neural Information Processing Systems*, pp. 225–231, 1999.
- Goldfarb, A. and Tucker, C. Online display advertising: Targeting and obtrusiveness. *Marketing Science*, 30(3):389–404, 2011.
- Gordon, G. J. Regret bounds for prediction problems. In *Conference on Learning Theory*, pp. 29–40, 1999.
- Hazan, E. Introduction to online convex optimization. *Foundations and Trends in Optimization*, 2(3-4):157–325, 2016.
- Hazan, E., Agarwal, A., and Kale, S. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.
- Jadbabaie, A., Rakhlin, A., Shahrampour, S., and Sridharan, K. Online optimization: Competing with dynamic comparators. In *International Conference on Artificial Intelligence and Statistics*, pp. 398–406, 2015.
- Jenatton, R., Huang, J., and Archambeau, C. Adaptive algorithms for online convex optimization with long-term constraints. In *International Conference on Machine Learning*, pp. 402–411, 2016.
- Li, X., Yi, X., and Xie, L. Distributed online optimization for multi-agent networks with coupled inequality constraints. *IEEE Transactions on Automatic Control*, 2020.
- Mahdavi, M., Jin, R., and Yang, T. Trading regret for efficiency: Online convex optimization with long term constraints. *Journal of Machine Learning Research*, 13(81):2503–2528, 2012.
- Mokhtari, A., Shahrampour, S., Jadbabaie, A., and Ribeiro, A. Online optimization in dynamic environments: Improved regret rates for strongly convex problems. In *IEEE Conference on Decision and Control*, pp. 7195–7201, 2016.
- Neely, M. J. and Yu, H. Online convex optimization with time-varying constraints. *arXiv:1702.04783*, 2017.
- Shalev-Shwartz, S. Online learning and online convex optimization. *Foundations and Trends in Machine Learning*, 4(2):107–194, 2012.
- Sun, W., Dey, D., and Kapoor, A. Safety-aware algorithms for adversarial contextual bandit. In *International Conference on Machine Learning*, pp. 3280–3288, 2017.
- van Erven, T. and Koolen, W. M. Metagrad: Multiple learning rates in online learning. In *Advances in Neural Information Processing Systems*, pp. 3666–3674, 2016.
- Yi, X., Li, X., Xie, L., and Johansson, K. H. Distributed online convex optimization with time-varying coupled inequality constraints. *IEEE Transactions on Signal Processing*, 68:731–746, 2020a.
- Yi, X., Li, X., Yang, T., Xie, L., Johansson, K. H., and Chai, T. Distributed bandit online convex optimization with time-varying coupled inequality constraints. *IEEE Transactions on Automatic Control*, 2020b.
- Yi, X., Li, X., Yang, T., Xie, L., Chai, T., and Johansson, K. H. Regret and cumulative constraint violation analysis for distributed online constrained convex optimization. *arXiv preprint arXiv:2105.00321*, 2021.
- Yu, H. and Neely, M. J. A low complexity algorithm with $O(\sqrt{T})$ regret and $O(1)$ constraint violations for online convex optimization with long term constraints. *Journal of Machine Learning Research*, 21(1):1–24, 2020.
- Yu, H., Neely, M., and Wei, X. Online convex optimization with stochastic constraints. In *Advances in Neural Information Processing Systems*, pp. 1428–1438, 2017.
- Yuan, D., Proutiere, A., and Shi, G. Distributed online linear regression. *IEEE Transactions on Information Theory*, 67(1):616–639, 2021a.
- Yuan, D., Proutiere, A., and Shi, G. Distributed online optimization with long-term constraints. *IEEE Transactions on Automatic Control*, 2021b.
- Yuan, J. and Lamperski, A. Online convex optimization for cumulative constraints. In *Advances in Neural Information Processing Systems*, pp. 6140–6149, 2018.

- Zhang, L. Online learning in changing environments. In *International Joint Conference on Artificial Intelligence*, pp. 5178–5182, 2020.
- Zhang, L., Yang, T., Yi, J., Rong, J., and Zhou, Z.-H. Improved dynamic regret for non-degenerate functions. In *Advances in Neural Information Processing Systems*, pp. 732–741, 2017.
- Zhang, L., Lu, S., and Zhou, Z.-H. Adaptive online learning in dynamic environments. In *Advances in Neural Information Processing Systems*, pp. 1323–1333, 2018a.
- Zhang, L., Yang, T., Jin, R., and Zhou, Z.-H. Dynamic regret of strongly adaptive methods. In *International Conference on Machine Learning*, pp. 5882–5891, 2018b.
- Zhang, L., Liu, T.-Y., and Zhou, Z.-H. Adaptive regret of convex and smooth functions. In *International Conference on Machine Learning*, pp. 7414–7423, 2019.
- Zhang, Y.-J., Zhao, P., and Zhou, Z.-H. A simple online algorithm for competing with dynamic comparators. In *Conference on Uncertainty in Artificial Intelligence*, pp. 390–399, 2020.
- Zhao, P. and Zhang, L. Improved analysis for dynamic regret of strongly convex and smooth functions. *arXiv:2006.05876*, 2020.
- Zhao, P., Zhang, Y.-J., Zhang, L., and Zhou, Z.-H. Dynamic regret of convex and smooth functions. In *Advances in Neural Information Processing Systems*, 2020.
- Zinkevich, M. Online convex programming and generalized infinitesimal gradient ascent. In *International Conference on Machine Learning*, pp. 928–936, 2003.