Safe Online Convex Optimization with Multi-Point Feedback

Spencer Hutchinson SHUTCHINSON@UCSB.EDU
University of California, Santa Barbara

Mahnoosh Alizadeh ALIZADEH@UCSB.EDU
University of California, Santa Barbara

Editors: A. Abate, K. Margellos, A. Papachristodoulou

Abstract

Motivated by the stringent safety requirements that are often present in real-world applications, we study a safe online convex optimization setting where the player needs to simultaneously achieve sublinear regret and zero constraint violation while only using zero-order information. In particular, we consider a multi-point feedback setting, where the player chooses $d + 1$ points in each round (where d is the problem dimension) and then receives the value of the constraint function and cost function at each of these points. To address this problem, we propose an algorithm that leverages forward-difference gradient estimation as well as optimistic and pessimistic action sets to achieve $\mathcal{O}(d\sqrt{T})$ regret and zero constraint violation under the assumption that the constraint function is smooth and strongly convex. We then perform a numerical study to investigate the impacts of the unknown constraint and zero-order feedback on empirical performance.

Keywords: bandit convex optimization, safe learning, zero-order optimization

1. Introduction

The online convex optimization (OCO) problem, formalized by Zinkevich (2003), is a sequential decision-making framework where, in each round $t \in [T]$, a player chooses a vector action x_t and subsequently observes the loss function f_t, with the goal of minimizing her cumulative loss $\sum_{t=1}^{T} f_t(x_t)$. The OCO setting has received significant attention due to its practical effectiveness in various fields (e.g. online advertising (McMahan et al. (2013)), network resource allocation (Yu and Neely (2019)) and power systems (Lesage-Landry et al. (2019))) and its role as a fundamental building block in modern learning and control approaches (e.g. online-to-batch (Cutkosky (2019)) and online control (Agarwal et al. (2019); Simchowitz et al. (2020))). At the same time, there has been considerable recent interest in learning and control approaches that can ensure constraints are always satisfied, even when they are a priori unknown (e.g. Sui et al. (2015); Junges et al. (2016); Usmanova et al. (2019)). This is motivated by safety-critical fields, such as clinical trials and power systems, where there is uncertainty about the constraints and constraint violation is not acceptable. Accordingly, in this work, we consider an OCO setting with an unknown constraint that *cannot be violated* while only giving the player *partial feedback* on the constraint and cost functions.

In particular, we generalize the setting of OCO with multi-point feedback and known constraints from Agarwal et al. (2010) to the scenario where the constraint is unknown and the player only receives zero-order constraint information at the played actions. Specifically, we consider an OCO setting where the player chooses multiple actions ($d + 1$ actions to be precise) in each round, and then observes the cost function and constraint function values at each of these points. Despite the limited information available, the player needs to ensure that all of the points that she chooses satisfy the constraints. This is challenging because the player needs to effectively balance constraint

satisfaction with regret minimization, while contending with errors in gradient estimation. Note that this problem generalizes safe zero-order convex optimization as the cost functions do not change in that setting (i.e. $f_t = f$ for all t) and thus the player is freely able to query the cost function as desired.

To address the stated problem, we introduce the algorithm MP-ROGD, which combines the ideas from OCO under multi-point feedback (Agarwal et al. (2010)) with the idea of optimistic and pessimistic action sets from Hutchinson and Alizadeh (2024). We rigorously show that when the player is given $d + 1$ points of feedback in each round (where d is the problem dimension) and the constraint function is smooth and strongly-convex, MP-ROGD always satisfies the constraints and enjoys $\mathcal{O}(d\sqrt{T})$ regret. Then, we perform a numerical study to assess the empirical performance of MP-ROGD against existing algorithms that either have access to zero-order cost information and complete constraint information (i.e. Agarwal et al. (2010)) or first-order cost and constraint information (i.e. Hutchinson and Alizadeh (2024)).

1.1. Related Work

Constraints on the player's actions are a fundamental part of the OCO framework as even the initial formulation (Zinkevich (2003)) assumes that the action set is bounded. However, this classical formulation assumes that these constraints are known, which may not be the case in some applications. To address this gap, a large body of literature has emerged that studies OCO with time-varying constraints that are only revealed after the player commits to an action, e.g. Mannor et al. (2009); Neely and Yu (2017); Cao and Liu (2018); Cao et al. (2018); Yi et al. (2020); Guo et al. (2022). However, due to the limited information given to the player, these works aim for sublinear constraint violation rather than zero constraint violation.

In a different direction, several recent works have considered OCO with fixed constraints and zero constraint violation while providing the learner with limited information on the constraints (Chaudhary and Kalathil (2022); Chang et al. (2023); Hutchinson and Alizadeh (2024)). In particular, Chaudhary and Kalathil (2022) gives an algorithm with $\tilde{\mathcal{O}}(T^{2/3})$ regret guarantees and high probability constraint satisfaction for an OCO setting with a linear constraint function and noisy feedback of the constraint function value at the chosen actions. This method relies on an iid exploration phase within a small safe region to learn the constraint function everywhere, which cannot be readily applied to the nonlinear constraints considered in our setting. The approach taken by Chaudhary and Kalathil (2022) is then extended to the distributed setting by Chang et al. (2023), where they additionally provide dynamic regret guarantees for both the cases of convex and non-convex cost functions. Hutchinson and Alizadeh (2024) take a different approach by assuming that the constraint function is smooth and strongly-convex, and give $\mathcal{O}(\sqrt{T})$ regret guarantees for the case when the player is given first-order feedback of the constraint function at the played actions. In this work, we build on the approach taken in Hutchinson and Alizadeh (2024) to address the more challenging setting where the player is only given multi-point zero-order feedback of the cost and constraint functions.

Another related area is "projection-free" OCO, which aims to develop OCO algorithms that do not require the computationally expensive projection operation. Since projections require full knowledge of the constraint, there are some shared interests between projection-free OCO and OCO with unknown constraints. One direction in projection-free OCO is focused on developing cheaper variants of the projection operation that can be used with standard algorithms, e.g. Mhammedi

(2022); Levy and Krause (2019). Another approach to projection-free OCO leverages the cheaper linear optimization oracle, which often involves variants of the Franke-Wolfe algorithm, e.g. Garber and Hazan (2016); Hazan and Minasyan (2020); Kretzu and Garber (2021). A third direction avoids projections by allowing some constraint violation, which shares some techniques with the literature on OCO with time-varying constraints, e.g. Mahdavi et al. (2012); Yu and Neely (2020); Guo et al. (2022). These approaches to projection-free OCO differ from the setting we consider in that they either allow constraint violation or assume access to different constraint oracles than we do, i.e. linear optimization oracle, membership oracle, or constraint function value and gradient at any point.

Our approach is also related to the literature on OCO with bandit feedback (first studied by Flaxman et al. (2005); Kleinberg (2004)), where the learner is only given the cost function value at the played action (or sometimes several played actions) rather than the entire cost function (f_t) at each time step. In fact, our setting can be considered a version of OCO with multi-point bandit feedback (Agarwal et al. (2010)) because we only give the player the cost function value at played actions. As such, we borrow ideas from the multi-point OCO literature and the related zero-order optimization literature, e.g. Agarwal et al. (2010); Duchi et al. (2015). Furthermore, recent works that study zero-order optimization with unknown constraints are relevant (Usmanova et al. (2020); Guo et al. (2023)), although this setting is distinct from ours because it considers a fixed cost function, i.e. $f_t = f$ for all t.

2. Preliminaries

2.1. Notation and Definitions

We use $\mathcal{O}(\cdot)$ to refer to big-O notation. Also, we denote the 2-norm by $\|\cdot\|$. For a natural number n, we use $[n]$ for the set $\{1, 2, ..., n\}$. For a matrix M, we use $M^\top$ to denote the transpose of M. The unit vector in the ith coordinate direction is denoted by e_i. A set $\mathcal{X} \subseteq \mathbb{R}^d$ is referred to as *convex* if $(1-\lambda)x + \lambda y \in \mathcal{X}$ for all $x, y \in \mathcal{X}$ and $\lambda \in [0, 1]$. For a convex set $\mathcal{X}$, a function $f : \mathcal{X} \to \mathbb{R}$ is referred to as *convex* if $f((1-\lambda)x + \lambda y) \leq (1-\lambda)f(x) + \lambda f(y)$ for all $x, y \in \mathcal{X}$ and $\lambda \in [0, 1]$. Also for a closed convex set $\mathcal{X} \subseteq \mathbb{R}^d$ and a vector $x \in \mathbb{R}^d$, we denote the projection operation with $\Pi_{\mathcal{X}}(y) = \arg\min_{x \in \mathcal{X}} \|x - y\|$. A useful fact is that for a closed convex set $\mathcal{X} \subseteq \mathbb{R}^d$ and vectors $y \in \mathbb{R}^d$ and $x \in \mathcal{X}$, it holds that $\|y - x\| \geq \|\Pi_{\mathcal{X}}(y) - x\|$. Lastly, we give the definitions for smooth and strongly convex functions which will be useful later.

Definition 1 (Smooth function) *Given a convex set $\mathcal{X}$, a differentiable convex function $h : \mathcal{X} \to \mathbb{R}$ is said to be* L-smooth *if*

$$h(y) \leq h(x) + \nabla h(x)^\top (y - x) + \frac{L}{2}\|y - x\|^2$$

for all $x, y \in \mathcal{X}$.

Definition 2 (Strongly-convex function) *Given a convex set $\mathcal{X}$, a differentiable convex function $h : \mathcal{X} \to \mathbb{R}$ is said to be* M-strongly convex *if*

$$h(y) \geq h(x) + \nabla h(x)^\top (y - x) + \frac{M}{2}\|y - x\|^2$$

for all $x, y \in \mathcal{X}$.

2.2. Problem Setup

We study an online convex optimization setting with $k = d + 1$ points of zero-order feedback in each round and an unknown constraint. This setting is defined by a *horizon* $T \in \mathbb{N}$, a known closed convex *action set* $\mathcal{X} \subseteq \mathbb{R}^d$, an unknown convex *constraint function* $g : \mathcal{X} \to \mathbb{R}$, and a sequence of adversarially-chosen convex *cost functions* $f_1, ..., f_T$ where $f_t : \mathcal{X} \to \mathbb{R}$ for every $t \in [T]$. The setting can then be specified as an iterative game between a player and an adversary, where at each *round* $t \in [T]$,

1. player chooses actions $x_{t,1}, x_{t,2}, ..., x_{t,k}$ from $\mathcal{X}$,
2. adversary chooses f_t and player incurs the cost $\frac{1}{k}\sum_{i=1}^{k} f_t(x_{t,i})$,
3. player observes $f_t(x_{t,1}), f_t(x_{t,2}), ..., f_t(x_{t,k})$ and $g(x_{t,1}), g(x_{t,2}), ..., g_t(x_{t,k})$.

Despite the fact that g is unknown, the player must ensure that $x_{t,1}, x_{t,2}, ..., x_{t,k}$ are in $\mathcal{G} := \{x \in \mathbb{R}^d : g(x) \leq 0\}$ for all $t \in [T]$. We will refer to the *feasible set* as $\mathcal{Y} := \mathcal{X} \cap \mathcal{G}$.

In addition to maintaining constraint satisfaction, the player also aims to minimize her loss relative to the optimal action in hindsight. That is, the player aims to minimize her *regret*, which is defined as

$$R_T := \frac{1}{k}\sum_{t=1}^{T}\sum_{i=1}^{k} f_t(x_{t,i}) - \sum_{t=1}^{T} f_t(x_*),$$

where $x_* = \arg\min_{x \in \mathcal{Y}} \sum_{t=1}^{T} f_t(x)$. Note that this notion of regret is standard in OCO with multi-point feedback, i.e. Agarwal et al. (2010).

2.3. Assumptions

Our approach to this problem uses several assumptions, which are given as follows. First, we assume that the cost functions have bounded gradients (Assumption 1) and that the action set is bounded (Assumption 2), which are standard assumptions in the OCO setting, e.g. Zinkevich (2003); Hazan (2016).

Assumption 1 (Bounded gradients) *For all $t \in [T]$, it holds that f_t is differentiable and $\|\nabla f_t(x)\| \leq G$ for all $x \in \mathcal{X}$.*

Assumption 2 (Bounded action set) *There exists a positive real D such that $\|x - y\| \leq D$ for all $x, y \in \mathcal{X}$.*

Next, we assume that the constraint function is smooth and strongly convex (Assumption 3) and that there is a known point that is strictly feasible (Assumption 4). Assumption 3 is critical to our approach for ensuring low regret because it allows us to construct sets that tightly underestimate and overestimate the constraint set. Assumption 4 ensures that there is a starting point that is known to satisfy the constraint, which is typically assumed in safe learning problems, e.g. Usmanova et al. (2020); Guo et al. (2023). In Assumption 4, the player is also given both the radius of a ball that is within the constraint (r) and an upper bound on the function value at the starting point ($-\epsilon$).

Assumption 3 (Smooth and strongly convex constraint) *The constraint function g is differentiable, L-smooth and M-strongly convex, where $\kappa := L/M > 1$.*[1]

1. If $\kappa = 1$, then the constraint is exactly specified by the smoothness and strongly-convexity assumption, and the problem can be solved with standard OCO methods. Therefore, our assumption that $\kappa > 1$ is not restrictive.

Algorithm 1: Multi-point Restrained Online Gradient Descent (MP-ROGD)

Input: $\mathcal{X}, G, L, M, r, \epsilon, \eta > 0, \delta \in (0,1), \alpha \in (0,1)$.

1 Set $\tilde{x}_1 = \mathbf{0}$ and $x_1 = \mathbf{0}$.
2 **for** $t = 1$ **to** T **do**
3 Play $x_t, x_t + \delta e_1, x_t + \delta e_2, ..., x_t + \delta e_d$.
4 Set $\tilde{\nabla} f_t(x_t) = \frac{1}{\delta} \sum_{i=1}^{d} (f_t(x_t + \delta e_i) - f_t(x_t)) e_i$ and $\tilde{\nabla} g(x_t) = \frac{1}{\delta} \sum_{i=1}^{d} (g(x_t + \delta e_i) - g(x_t)) e_i$.
5 Update $\mathcal{Y}_t^o$ and $\mathcal{Y}_t^p$ with (1) and (2).
6 $\tilde{x}_{t+1} = \Pi_{\mathcal{Y}_t^o}(\tilde{x}_t - \eta \tilde{\nabla} f_t(x_t))$.
7 $\gamma_t = \max\{\mu \in [0,1] : x_t + \mu(\tilde{x}_{t+1} - x_t) \in \mathcal{Y}_t^p\}$.
8 $x_{t+1} = (1 - \alpha)(x_t + \gamma_t(\tilde{x}_{t+1} - x_t))$.
9 **end**

Assumption 4 (Initial feasible point) *It holds that* $\mathbf{0}$ *is in* $\mathcal{X}$ *and* $g(\mathbf{0}) \leq -\epsilon$ *for some* $\epsilon > 0$. *Furthermore, there exists* $r > 0$ *such that* $r\mathbb{B} \subseteq \mathcal{Y}$.

Lastly, we assume that the cost functions are smooth, which ensures that the error in gradient estimation is small as in Agarwal et al. (2010); Duchi et al. (2015). Note that, unlike standard convex optimization, the OCO setting does not enjoy improved regret guarantees when the cost functions are smooth (see Table 3.1 in Hazan (2016)).

Assumption 5 (Smooth cost functions) *For all* $t \in [T]$, *it holds that* f_t *is* L*-smooth.*

3. Proposed Algorithm

To address the stated problem, we propose the algorithm Multi-Point Restrained Online Gradient Descent (MP-ROGD) as stated in Algorithm 1. MP-ROGD operates by using gradient estimators to approximate the gradients of the constraint and cost functions as described in Section 3.1, and then leveraging optimistic and pessimistic action sets to ensure small regret while maintaining constraint satisfaction as described in Section 3.2. We give guarantees that the algorithm is well-defined and that it never violates the constraints in Section 3.3. The regret of MP-ROGD is studied in the following section (Section 4).

3.1. Gradient Estimation

Because the algorithm does not have access to gradients of the cost functions or the constraint function, it estimates the gradients with only zero-order information. The algorithm does this by playing the current iterate x_t as well as points perturbed away from the current iterate by δ in each coordinate direction (given in line 3). It then estimates the gradient at the current iterate using forward difference (line 4). We give some useful properties of this gradient estimator in the following proposition. Note that an appropriate choice for δ will be specified later.

Proposition 3 (Properties of gradient estimators) *Let Assumptions 1, 3 and 5 hold. Then, for every* $t \in [T]$, *it holds that*

$$\|\tilde{\nabla} f_t(x_t) - \nabla f_t(x_t)\| \leq \frac{1}{2}\sqrt{d}L\delta \quad \textit{and} \quad \|\tilde{\nabla} g(x_t) - \nabla g(x_t)\| \leq \frac{1}{2}\sqrt{d}L\delta.$$

Furthermore, it holds that

$$\|\tilde{\nabla} f_t(x_t)\| \leq dG.$$

The key takeaways from Proposition 3 are that the gradient estimation error shrinks as δ shrinks and that the norm of the gradient estimator can be bounded independently of δ. Since regret will grow as both gradient estimation error and the norm of the gradient estimator increases, Proposition 3 tells us that we can take δ to be small without sacrificing regret. This is important because a large δ might otherwise jeopardize constraint satisfaction, and therefore taking δ to be sufficiently small (see the choice of δ in Theorem 7) will allow for both low regret and constraint satisfaction.

3.2. Optimistic and Pessimistic Action Sets

The proposed algorithm updates the iterate x_t using a technique that leverages both an *optimistic action set* (denoted by $\mathcal{Y}_t^o$) and a *pessimistic action set* (denoted by $\mathcal{Y}_t^p$), which are known to contain the true feasible set and be contained by the true feasible set, respectively. We refer to $\mathcal{Y}_t^o$ (resp. $\mathcal{Y}_t^p$) as the optimistic (resp. pessimistic) action set because it estimates the feasible set while taking the unknown information about the constraint to be as favorable (resp. unfavorable) as reasonably possible given what has been observed.[2] The algorithm uses these sets in each round by updating an *optimistic iterate* ($\tilde{x}_t$) with gradient descent on the optimistic set (line 6) and then moving the *played iterate* (x_t) towards the optimistic iterate while keeping it in the pessimistic set (line 8). This ensures that the optimistic iterates incur low regret while simultaneously keeping the played iterates within the constraint set. The specific construction of the optimistic and pessimistic action sets, which we discuss next, ensures that the played iterates stay near to the optimistic iterates and therefore that the played iterates incur low regret as well. Note that the played iterates are scaled down by $(1-\alpha)$ in line 8 to ensure that the perturbed points ($x_t + \delta e_i$) do not violate the constraints.

The optimistic and pessimistic action sets are constructed by combining the smoothness and strong-convexity of the constraint function with the error bound on the gradient estimator in Proposition 3. Specifically, the optimistic and pessimistic action sets are defined as

$$\mathcal{Y}_t^o := \left\{ x \in \mathcal{X} : g(x_t) - \frac{1}{2}\sqrt{d}L\delta D + \tilde{\nabla} g(x_t)^\top (x - x_t) + \frac{M}{2}\|x - x_t\|^2 \leq 0 \right\}, \tag{1}$$

and,

$$\mathcal{Y}_t^p := \left\{ x \in \mathcal{X} : g(x_t) + \frac{1}{2}\sqrt{d}L\delta D + \tilde{\nabla} g(x_t)^\top (x - x_t) + \frac{L}{2}\|x - x_t\|^2 \leq 0 \right\} \tag{2}$$

respectively. In the following proposition, we show that the optimistic and pessimistic sets do in fact overestimate and underestimate the constraint set, respectively.

Proposition 4 *Let Assumptions 2 and 3 hold. Then, it follows that $\mathcal{Y}_t^p \subseteq \mathcal{Y} \subseteq \mathcal{Y}_t^o$ for all t.*

3.3. Validity and Safety Gaurantee

It is necessary to show that the algorithm is well-defined and that the constraint is satisfied at all rounds. The main point of concern is whether the pessimistic set $\mathcal{Y}_t^p$ is nonempty. In the following proposition, we provide a range of values of δ for which the pessimistic set is guaranteed to be nonempty.

2. We borrow this terminology from the stochastic bandit literature (e.g. Abbasi-Yadkori et al. (2011)) where "optimism in the face of uncertainty" is a popular design paradigm.

Proposition 5 (Validity) *Let Assumptions 3 and 4 hold. If* $\delta \leq \frac{2\alpha\epsilon}{\sqrt{d}LD}$, *then* $x_t \in \mathcal{Y}_t^p$ *(and therefore* $\mathcal{Y}_t^p$ *is nonempty) for all rounds* $t \in [T]$.

Next, we show that all actions satisfy the constraint if δ is chosen appropriately.

Proposition 6 (Safety guarantee) *Let Assumption 3 hold and assume that* $x_t \in \mathcal{Y}_t^p$ *for all* $t \in [T]$. *If* $\delta \leq \alpha r$, *then all actions played by the algorithm, i.e.* $x_t, x_t + \delta e_1, ..., x_t + \delta e_d$ *for all* t, *are in the feasible set* $\mathcal{Y}$.

4. Regret Analysis

In the following theorem, we show that, with an appropriate choice of algorithm parameters (α, δ, η), our proposed algorithm MP-ROGD (Algorithm 1) enjoys $\mathcal{O}(d\sqrt{T})$ regret and ensures that the constraint is always satisfied. A proof sketch of Theorem 7 and the supporting lemmas are given below.

Theorem 7 *Let Assumptions 1, 2, 3 and 4 hold. If* $\alpha = \min(0.5, \frac{dG}{D}(1 - \frac{1}{\kappa})\eta)$ *and*

$$\delta = \min\left(\frac{1}{\left(\frac{1}{2}\sqrt{d}LD + G\right)T}, \frac{2(\kappa-1)\alpha\epsilon}{(\kappa+1)\sqrt{d}LD}, \alpha r\right),$$

then all actions chosen by MP-ROGD (Algorithm 1) satisfy the constraint, and the regret satisfies

$$R_T \leq 2d^2G^2\left(\kappa - \frac{3}{4}\right)\eta T + \frac{D^2}{2\eta} + 1.$$

Furthermore, choosing $\eta = \frac{D}{2\sqrt{(d/4+\kappa-1)dG^2T}}$ *ensures that*

$$R_T \leq 2DG\sqrt{d\left(\frac{1}{4}d + \kappa - 1\right)T} + 1.$$

Proof sketch: First, we seperate the regret due to the iterate x_t from the regret due to the perturbed iterates $x_t + \delta e_1, ..., x_t + \delta e_d$ as

$$\begin{aligned} R_T &= \frac{1}{k}\sum_{t=1}^{T}\sum_{i=1}^{k} f_t(x_{t,i}) - \sum_{t=1}^{T} f_t(x_*) \\ &= \underbrace{\sum_{t=1}^{T}\left(f_t(x_t) - f_t(x_*)\right)}_{\text{Term I}} + \underbrace{\sum_{t=1}^{T}\left(\frac{1}{d+1}\left(f_t(x_t) + \sum_{i=1}^{d} f_t(x_t + \delta e_i)\right) - f_t(x_t)\right)}_{\text{Term II}}, \end{aligned}$$

and note that $\text{Term II} \leq TG\delta$ given that the gradient of f_t is assumed to be bounded by G in Assumption 1. Then, we decompose Term I as

$$\begin{aligned} \text{Term I} = \sum_{t=1}^{T}\left(f_t(x_t) - f_t(x_*)\right) &\leq \sum_{t=1}^{T}\nabla f_t(x_t)^\top (x_t - x_*) \\ &= \underbrace{\sum_{t=1}^{T}\nabla f_t(x_t)^\top (x_t - \tilde{x}_t)}_{\text{Term I.A}} + \underbrace{\sum_{t=1}^{T}\nabla f_t(x_t)^\top (\tilde{x}_t - x_*)}_{\text{Term I.B}}, \end{aligned} \tag{3}$$

where the inequality is due to convexity (using the idea from Zinkevich (2003) of studying the linearized regret). Term I.A is the difference in (linearized) cost between the played iterate x_t and the optimistic iterate $\tilde{x}_t$, while Term I.B can be interpreted as the linearized regret due to the optimistic iterate. In Lemmas 8 and 9 in the following subsection, we show that the specific structure of the optimistic and pessimistic sets ensures that the distance between x_t and $\tilde{x}_t$ is small and therefore that Term I.A is small. Furthermore, the optimistic iterates are updated with gradient descent on the optimistic set, which is known to contain the true feasible set, so we can apply techniques from multi-point OCO (Agarwal et al. (2010)) to bound Term I.B. This approach uses Lemma 10, which is given in the following subsection.

4.1. Supporting Lemmas

The proof of Theorem 7 relies on three key lemmas that are given in this section. The first two lemmas (Lemmas 8 and 9) establish a bound on the distance between the played iterates x_t and the optimistic iterates $\tilde{x}_t$, while the third lemma (Lemma 10) establishes a bound on the linearized regret of the optimistic iterate. In particular, Lemma 8 (given in the following) shows that γ_t is always larger than $1/\kappa$ when δ is chosen sufficiently small.

Lemma 8 *Let Assumptions 3 and 4 hold. If* $\delta \leq \frac{2(\kappa-1)\alpha\epsilon}{(\kappa+1)\sqrt{d}LD}$, *then* $\gamma_t \geq 1/\kappa$ *for all* $t \in [T]$.

This result is then used in Lemma 9 to show that (when δ is sufficiently small) the distance between the optimistic iterate $\tilde{x}_t$ and played iterate x_t is always bounded by a value proportional to η and α. Since α has no other restrictions, we can choose α to be proportional to η and therefore Lemma 9 tells us that $\|x_t - \tilde{x}_t\| \leq \mathcal{O}(\eta)$. At the same time, η needs to be chosen as $\Theta(\frac{1}{\sqrt{T}})$ to ensure optimal regret for gradient descent-based algorithms. As it happens, Lemma 9 implies that such a choice of η also ensures that Term I.B in (3) is $\mathcal{O}(\sqrt{T})$, i.e. that $\sum_{t=1}^{T} \|x_t - \tilde{x}_t\| \leq \mathcal{O}(\sqrt{T})$.

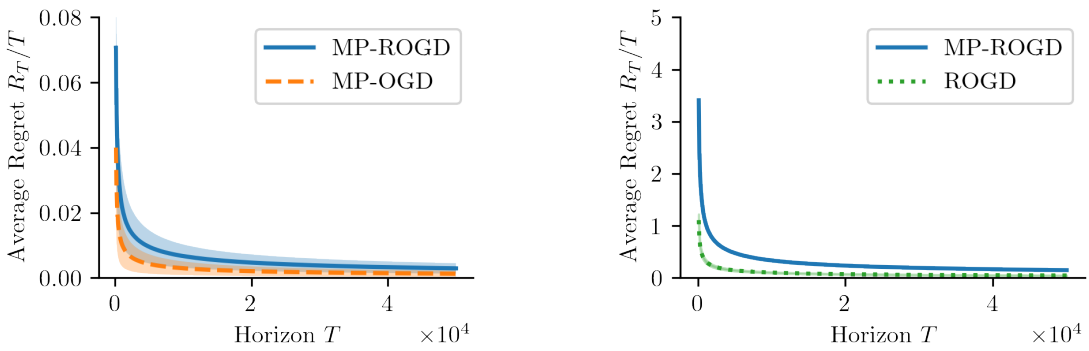
Lemma 9 *Let Assumptions 1, 3 and 4 hold. Fix any* $\rho > 0$. *If* $\eta \leq \frac{1/\kappa}{2dG(1-1/\kappa)}\rho$, $\alpha \leq \frac{1}{2D\kappa}\rho$ *and* $\delta \leq \frac{2(\kappa-1)\alpha\epsilon}{(\kappa+1)\sqrt{d}LD}$, *then it holds that* $\|x_t - \tilde{x}_t\| \leq \rho$ *for all* t.

Lastly, we give Lemma 10, which provides a bound on the (estimated) linearized regret of the optimistic iterates. In particular, it is easy to see that by summing (4) over t, the righthand side telescopes, yielding the bound $\frac{1}{\eta}D^2 + \frac{1}{2}\eta d^2G^2T$. Choosing $\eta = \Theta(1/\sqrt{T})$ ensures that this is $\mathcal{O}(\sqrt{T})$. This can then be used to bound Term I.B in (3), although there will be an additive $\frac{1}{2}\sqrt{d}L\delta RT$ (due to Proposition 3) because (4) is in terms of the estimated gradient $\tilde{\nabla} f_t$ rather than the true gradient ∇f_t. However, choosing $\delta \leq \frac{1}{T}$ ensures that this $\mathcal{O}(1)$.

Lemma 10 *Let Assumptions 1 and 3 hold. Then, for any* $v \in \mathcal{Y}$, *it holds that*

$$\tilde{\nabla} f_t(x_t)^\top (\tilde{x}_t - v) \leq \frac{1}{2\eta}\left(\|\tilde{x}_t - v\|^2 - \|\tilde{x}_{t+1} - v\|^2\right) + \frac{1}{2}\eta d^2 G^2, \tag{4}$$

for all $t \in [T]$.



(*a*) (*b*)

Figure 1: Average regret of MP-ROGD and benchmark algorithms in a setting with linear cost functions and a quadratic constraint function (a) and a setting with quadratic cost functions and quadratic constraint (b). The benchmark algorithms are MP-OGD (Agarwal et al. (2010)) with full constraint information and ROGD (Hutchinson and Alizadeh (2024)) with first-order constraint feedback.

5. Numerical Experiments

In order to assess the empirical performance of MP-ROGD, we compare MP-ROGD with two different benchmark algorithms in toy experimental settings. In the first experimental setting, we study the impact of unknown constraints by running MP-ROGD alongside multi-point online gradient descent (Agarwal et al. (2010)) which uses full constraint information, and in the second setting, we study the impact of zero-order feedback by running MP-ROGD alongside ROGD (Hutchinson and Alizadeh (2024)) which uses first-order feedback.

5.1. Impact of unknown contraints

To study the impact of unknown constraints on empirical performance, we compare MP-ROGD to online gradient descent with $d+1$ points of feedback from Agarwal et al. (2010) (abbreviated MP-OGD) which uses the full constraint information. We run these algorithms in a toy setting with cost functions $f_t(x) = \theta_t^\top x$ with $\theta_t \sim \mathcal{U}[0,1]^d$ and constraint function of the form $g(x) = a\|x-b\|^2 + c$ where the problem dimension is $d = 2$. We consider 10 randomly sampled settings of this form, where $a \sim \mathcal{U}[1,10]$, $b \sim \mathcal{U}(0.2\mathbb{S})$, $c = -\xi^2 a$ and $\xi \sim \mathcal{U}[0.3, 0.8]$. We sample the problem parameters in this manner because it ensures that we can take $\mathcal{X} = \mathbb{B}$ such that $\mathcal{G} \subseteq \mathcal{X}$ which allows for easy computation, and $r = 0.1$ in the sense of Assumption 4. Furthermore, we take $G = \sqrt{2}$ (Assumption 1), $R = 2$ (Assumption 2), $\epsilon = -c, r = 0.1$ (Assumption 4), and $L = 20$, $M = 2$ (Assumption 3). Note that the constraint function is $2a$-smooth and $2a$-strongly convex, but the player does not know this, so we only provide the player with the information that $L = 20$, $M = 2$ which can be deduced from the sampling distribution for a. For MP-ROGD, we choose $\eta = \frac{R}{dG\sqrt{T}}$, $\alpha = dGM(1-1/\kappa)\eta/R$ and $\delta = \min(1/T, (\kappa-1)\alpha\epsilon/((\kappa+1)\sqrt{d}LR), \alpha r)$ which satisfies the conditions in Theorem 7 for $\mathcal{O}(d\sqrt{T})$ regret and no constraint violation. For MP-OGD, which is specified by the update $x_{t+1} = \Pi_{(1-\alpha)\mathcal{Y}}(x_t - \eta\tilde{\nabla} f_t(x_t))$, we choose $\eta = \frac{R}{dG\sqrt{T}}$, $\delta = 1/T$ $\alpha = \delta/\bar{r}$ where $\bar{r} = \xi - 0.2$ (the largest ball radius that is within the constraint).

The results of these experiments are shown in Figure 1(*a*). These results are generated by running both algorithms in each randomly sampled setting for every $T \in \{1 \times 10^2, 2 \times 10^2, ..., 5 \times 10^4\}$ and calculating the average regret R_T/T for each. The average and standard deviation of R_T/T across settings is shown in Figure 1(*a*). From these results, we can see that there is a significant performance gap between MP-ROGD with only zero-order constraint feedback, and MP-OGD with full constraint information. Notably, this differs from the case of first-order feedback, for which Hutchinson and Alizadeh (2024) observed little performance difference between ROGD with first-order feedback and online gradient descent with full constraint information. This suggests that the "price of safety" increases as less constraint information is given to the player.

5.2. Impact of zero-order feedback

To study the impact of multi-point feedback on the empirical performance of safe OCO algorithms, we compare MP-ROGD with ROGD from Hutchinson and Alizadeh (2024) which uses first-order constraint feedback. We run these algorithms in a toy setting with cost functions $f_t(x) = (x - b_t)^\top A_t(x - b_t)$ where A_t and b_t are randomly sampled in each round, and constraint function $g(x) = x^\top \tilde{A}x + \tilde{c}$. We generate A_t in each round by sampling $A_{t,\text{raw}} \sim \mathcal{U}[0,1]^{d \times d}$, taking the symmetric part $A_{t,\text{sym}} = 0.5(A_{t,\text{raw}} + A_{t,\text{raw}}^\top)$, normalizing its spectrum $A_{t,\text{norm}} = (A_{t,\text{sym}} - 0.5I)/(d - 0.5)$ and finally by shifting and scaling $A_t = 5(A_{t,\text{norm}} + I)$ to ensure the spectrum is within $[1, 10]$. Also, we sample $b_t \sim \mathcal{U}[1,2]^d$ in each round which will ensure that the constraint is tight on the optimal action. We consider 10 randomly sampled settings with $\tilde{A} = \text{diag}(\tilde{a})$ and $\tilde{c} = \min_i(\tilde{a}_i)$, where $\tilde{a} \sim \mathcal{U}[1,10]^d$. Similar to Section 5.1, this ensures that $\mathcal{G} \subseteq \mathcal{X}$ when $\mathcal{X} = \mathbb{B}$. Furthermore, we choose the problem parameters $G = 60$ (Assumption 1), $R = 2$ (Assumption 2), $\epsilon = 1, r = 1/\sqrt{10}$ (Assumption 4), $L = 20, M = 2$ (Assumption 3, Assumption 5). We choose algorithm parameters of MP-ROGD as $\eta = \frac{R}{dG\sqrt{T}}$, $\alpha = dGM(1 - 1/\kappa)\eta/R$ and $\delta = \min(1/T, (\kappa - 1)\alpha\epsilon/((\kappa + 1)\sqrt{d}LR), \alpha r)$ (same as in Section 5.1). Also, we run ROGD with $\eta = \frac{R}{G\sqrt{T}}$ as suggested in Hutchinson and Alizadeh (2024).

The results of these experiments are shown in Figure 1(*b*), which is computed the same as for the results in Section 5.1. These results show that ROGD outperforms MP-ROGD, suggesting that there is a cost to only having zero-order feedback versus first-order feedback.

6. Conclusion

In this work, we study a safe OCO problem where the player chooses $d+1$ actions in each round and observes the cost and constraint values at each of these points. To address this problem, we present the algorithm MP-ROGD, which enjoys $\mathcal{O}(d\sqrt{T})$ regret and never violates the constraints. One interesting direction for future work is investigating whether it is possible to do safe OCO under nonlinear constraints with less constraint information (e.g. one or two-point feedback), although this might require weaker notions of constraint satisfaction (e.g. in expectation). Another interesting direction for future work is investigating whether our proposed algorithmic approach can be applied to related learning problems such as distributed online optimization or online control.

Acknowledgments

This work was supported by NSF grant #1847096.

References

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in Neural Information Processing Systems*, 24, 2011.

Alekh Agarwal, Ofer Dekel, and Lin Xiao. Optimal algorithms for online convex optimization with multi-point bandit feedback. In *Conference on Learning Theory*, pages 28–40, 2010.

Naman Agarwal, Brian Bullins, Elad Hazan, Sham Kakade, and Karan Singh. Online control with adversarial disturbances. In *International Conference on Machine Learning*, pages 111–119. PMLR, 2019.

Xuanyu Cao and KJ Ray Liu. Online convex optimization with time-varying constraints and bandit feedback. *IEEE Transactions on Automatic Control*, 64(7):2665–2680, 2018.

Xuanyu Cao, Junshan Zhang, and H Vincent Poor. A virtual-queue-based algorithm for constrained online convex optimization with applications to data center resource allocation. *IEEE Journal of Selected Topics in Signal Processing*, 12(4):703–716, 2018.

Ting-Jui Chang, Sapana Chaudhary, Dileep Kalathil, and Shahin Shahrampour. Dynamic regret analysis of safe distributed online optimization for convex and non-convex problems. *Transactions on Machine Learning Research*, 2023.

Sapana Chaudhary and Dileep Kalathil. Safe online convex optimization with unknown linear safety constraints. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 6175–6182, 2022.

Ashok Cutkosky. Anytime online-to-batch, optimism and acceleration. In *International Conference on Machine Learning*, pages 1446–1454. PMLR, 2019.

John C Duchi, Michael I Jordan, Martin J Wainwright, and Andre Wibisono. Optimal rates for zero-order convex optimization: The power of two function evaluations. *IEEE Transactions on Information Theory*, 61(5):2788–2806, 2015.

Abraham D Flaxman, Adam Tauman Kalai, and H Brendan McMahan. Online convex optimization in the bandit setting: gradient descent without a gradient. In *Proceedings of the sixteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 385–394, 2005.

Dan Garber and Elad Hazan. A linearly convergent variant of the conditional gradient algorithm under strong convexity, with applications to online and stochastic optimization. *SIAM Journal on Optimization*, 26(3):1493–1528, 2016.

Baiwei Guo, Yuning Jiang, Maryam Kamgarpour, and Giancarlo Ferrari-Trecate. Safe zeroth-order convex optimization using quadratic local approximations. In *2023 European Control Conference (ECC)*, pages 1–8. IEEE, 2023.

Hengquan Guo, Xin Liu, Honghao Wei, and Lei Ying. Online convex optimization with hard constraints: Towards the best of two worlds and beyond. *Advances in Neural Information Processing Systems*, 35:36426–36439, 2022.

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

Elad Hazan and Edgar Minasyan. Faster projection-free online learning. In *Conference on Learning Theory*, pages 1877–1893. PMLR, 2020.

Spencer Hutchinson and Mahnoosh Alizadeh. Safe online convex optimization with first-order feedback. In *2024 American Control Conference (ACC)*. IEEE, 2024.

Sebastian Junges, Nils Jansen, Christian Dehnert, Ufuk Topcu, and Joost-Pieter Katoen. Safety-constrained reinforcement learning for mdps. In *International conference on tools and algorithms for the construction and analysis of systems*, pages 130–146. Springer, 2016.

Robert Kleinberg. Nearly tight bounds for the continuum-armed bandit problem. *Advances in Neural Information Processing Systems*, 17, 2004.

Ben Kretzu and Dan Garber. Revisiting projection-free online learning: the strongly convex case. In *International Conference on Artificial Intelligence and Statistics*, pages 3592–3600. PMLR, 2021.

Antoine Lesage-Landry, Han Wang, Iman Shames, Pierluigi Mancarella, and Joshua A Taylor. Online convex optimization of multi-energy building-to-grid ancillary services. *IEEE Transactions on Control Systems Technology*, 28(6):2416–2431, 2019.

Kfir Levy and Andreas Krause. Projection free online learning over smooth sets. In *International Conference on Artificial Intelligence and Statistics*, pages 1458–1466. PMLR, 2019.

Mehrdad Mahdavi, Rong Jin, and Tianbao Yang. Trading regret for efficiency: online convex optimization with long term constraints. *The Journal of Machine Learning Research*, 13(1): 2503–2528, 2012.

Shie Mannor, John N Tsitsiklis, and Jia Yuan Yu. Online learning with sample path constraints. *Journal of Machine Learning Research*, 10(3), 2009.

H Brendan McMahan, Gary Holt, David Sculley, Michael Young, Dietmar Ebner, Julian Grady, Lan Nie, Todd Phillips, Eugene Davydov, Daniel Golovin, et al. Ad click prediction: a view from the trenches. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1222–1230, 2013.

Zakaria Mhammedi. Efficient projection-free online convex optimization with membership oracle. In *Conference on Learning Theory*, pages 5314–5390. PMLR, 2022.

Michael J Neely and Hao Yu. Online convex optimization with time-varying constraints. *arXiv preprint arXiv:1702.04783*, 2017.

Max Simchowitz, Karan Singh, and Elad Hazan. Improper learning for non-stochastic control. In *Conference on Learning Theory*, pages 3320–3436. PMLR, 2020.

Yanan Sui, Alkis Gotovos, Joel Burdick, and Andreas Krause. Safe exploration for optimization with gaussian processes. In *International Conference on Machine Learning*, pages 997–1005. PMLR, 2015.

Ilnura Usmanova, Andreas Krause, and Maryam Kamgarpour. Safe convex learning under uncertain constraints. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pages 2106–2114. PMLR, 2019.

Ilnura Usmanova, Andreas Krause, and Maryam Kamgarpour. Safe non-smooth black-box optimization with application to policy search. In *Learning for Dynamics and Control*, pages 980–989. PMLR, 2020.

Xinlei Yi, Xiuxian Li, Lihua Xie, and Karl H Johansson. Distributed online convex optimization with time-varying coupled inequality constraints. *IEEE Transactions on Signal Processing*, 68: 731–746, 2020.

Hao Yu and Michael J Neely. Learning-aided optimization for energy-harvesting devices with outdated state information. *IEEE/ACM Transactions on Networking*, 27(4):1501–1514, 2019.

Hao Yu and Michael J. Neely. 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.

Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *International Conference on Machine Learning*, pages 928–936, 2003.