Dynamic Regret Analysis of Safe Distributed Online Optimization for Convex and Non-convex Problems

Anonymous authors
Paper under double-blind review

Abstract

This paper addresses safe distributed online optimization over an unknown set of linear safety constraints. A network of agents aims at jointly minimizing a *global*, *time-varying* function, which is only partially observable to each individual agent. Therefore, agents must engage in *local* communications to generate a *safe* sequence of actions competitive with the best minimizer sequence in hindsight, and the gap between the performance of two sequences is quantified via dynamic regret. We propose distributed safe online gradient descent (D-Safe-OGD) with an exploration phase, where all agents estimate the constraint parameters collaboratively to build estimated feasible sets, ensuring the action selection safety during the optimization phase. We prove that for convex functions, D-Safe-OGD achieves a dynamic regret bound of $O(T^{2/3}\sqrt{\log T} + T^{1/3}C_T^*)$, where C_T^* denotes the path-length of the best minimizer sequence. We further prove a dynamic regret bound of $O(T^{2/3}\sqrt{\log T} + T^{2/3}C_T^*)$ for certain non-convex problems, which establishes the first dynamic regret bound for a *safe distributed* algorithm in the *non-convex* setting.

1 Introduction

Online learning (or optimization) is a sequential decision-making problem modeling a repeated game between a learner and an adversary (Hazan, 2016). At each round t, $t \in [T] \triangleq \{1, \dots, T\}$, the learner chooses an action $\mathbf{x}_t$ in a convex set $\mathcal{X} \subseteq \mathrm{R}^d$ using the information from previous observations and suffers the loss $f_t(\mathbf{x}_t)$, where the function $f_t : \mathcal{X} \to \mathrm{R}$ is chosen by the adversary. Due to the sequential nature of the problem, a commonly used performance metric is *regret*, defined as the difference between the cumulative loss incurred by the learner and that of a benchmark comparator sequence. When the comparator sequence is fixed, this metric is called *static* regret, defined as follows

$$\mathbf{Reg}_T^s \triangleq \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}). \tag{1}$$

Static regret is well-studied in the online optimization literature. In particular, it is well-known that online gradient descent (OGD) achieves an $O(\sqrt{T})$ (respectively, $O(\log T)$) static regret bound for convex (respectively, exp-concave and strongly convex) problems (Zinkevich, 2003; Hazan et al., 2007), and these bounds are optimal in the sense of matching the lower bound of regret in the respective problems (Hazan, 2016).

For non-convex problems, however, we expect that the standard regret notion used in the convex setting may not be a tractable measure for gauging the algorithm performance. For example, in the context of online non-convex optimization, Hazan et al. (2017) quantified the regret in terms of the norm of (projected) gradients, consistent with the stationarity measure in offline optimization. More recently, Ghai et al. (2022) showed that under certain geometric and smoothness conditions, OGD applied to non-convex functions is an approximation of online mirror descent (OMD) applied to convex functions under a reparameterization. In view of this equivalence, OGD achieves an $O(T^{2/3})$ static regret that is defined in (1).

A more stringent benchmark for measuring the performance of online optimization algorithms is the *dynamic* regret (Besbes et al., 2015; Jadbabaie et al., 2015), defined as

$$\mathbf{Reg}_T^d \triangleq \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^*), \tag{2}$$

where $\mathbf{x}_t^* \triangleq \operatorname{argmin}_{\mathbf{x}\in\mathcal{X}} f_t(\mathbf{x})$. It is well-known that dynamic regret scales linearly with T in the worst-case scenario, when the function sequence fluctuates drastically over time. Therefore, various works have adopted a number of variation measures to characterize the dynamic regret bound. We provide a detailed review of these measures in Section 2 and describe the *safe distributed* online optimization problem (which is the focus of this work) in the next section.

1.1 Safe Distributed Online Optimization

There are two distinctive components that make "safe distributed online optimization" more challenging than the standard centralized online optimization:

(i) **Distributed Setting:** Distributed online optimization has been widely applied to robot formation control (Dixit et al., 2019b), distributed target tracking (Shahrampour & Jadbabaie, 2017), and localization in sensor networks (Akbari et al., 2015). In this setting, a network of m agents aims at solving the online optimization problem collectively. The main challenge is that the time-varying function sequence is only partially observable to each individual agent. Each agent $j \in [m]$ receives (gradient) information about the "local" function $f_{j,t}(\cdot)$, but the objective is to control the dynamic regret of each agent with respect to the global function $f_t(\cdot) = \sum_{i=1}^m f_{i,t}(\cdot)$, i.e.,

$$\mathbf{Reg}_{j,T}^d \triangleq \sum_{t=1}^T f_t(\mathbf{x}_{j,t}) - \sum_{t=1}^T f_t(\mathbf{x}_t^*) = \sum_{t=1}^T \sum_{i=1}^m f_{i,t}(\mathbf{x}_{j,t}) - \sum_{t=1}^T f_t(\mathbf{x}_t^*). \tag{3}$$

Therefore, despite availability of only local information, the action sequence of agent j is evaluated in the global function and is compared to the global minimizer sequence. It is evident that agents must communicate with each other (subject to a graph/network constraint) to approximate the global function, which is common to distributed problems. The discussion on the network structure and communication protocols is provided in Section 3.3.

(ii) **Safety Constraints:** The literature on distributed online optimization has mostly focused on problems where the constraint set $\mathcal{X}$ is known, and less attention has been given to problems with *unknown* feasible sets (see Section 2 for a comprehensive literature review). However, in many real-world applications, this set represents certain safety constraints that are *unknown* in advance. Examples include voltage regulation constraints in power systems (Dobbe et al., 2020), transmission bandwidth in communication networks due to human safety considerations (Luong et al., 2019) and stabilizing action set in robotics applications (Åström & Murray, 2010). In these scenarios, one needs to perform parameter estimation to *learn* the safety constraints while ensuring that the regret is as small as possible.

1.2 Contributions

In this work, we address the problem of *distributed* online optimization with *unknown* linear safety constraints. In particular, the constraint set

$$\mathcal{X}^s \triangleq \{\mathbf{x} \in \mathrm{R}^d : \mathbf{A}\mathbf{x} \leq \mathbf{b}\},$$

is linear, where $\mathbf{A}$ is *unknown* and must be learned by agents to subsequently choose their actions from this set. The superscript s in $\mathcal{X}^s$ alludes to safety. Our specific objective is to study *dynamic* regret (2) for both convex and non-convex problems when the set $\mathcal{X}^s$ is unknown to agents. Our contributions are three-fold:

1) We propose and analyze safe distributed online gradient descent (D-Safe-OGD) algorithm, which has two phases (exploration and optimization). In the exploration phase, agents individually explore and jointly estimate the constraint parameters in a distributed fashion. Then, each agent constructs a feasible set with its own estimate, which ensures the action selection safety with high probability (Lemma 2). Since the estimates are only local, in the optimization phase, agents apply distributed OGD projected to *different* feasible sets, which brings forward an additional technical challenge. We tackle this using the geometric property of linear constraints (Fereydounian et al., 2020) as well as the sensitivity analysis of perturbed optimization problems with second order regular constraints (Bonnans et al., 1998), which allows us to quantify the distance between projections of a point to two different sets that are "close enough" to each other (Lemma 12).

2) We analyze D-Safe-OGD in the *convex* setting. Due to the challenge discussed in the previous item, we cannot directly apply existing results on distributed online optimization with a common feasible set. The

agents must use the exploration time to estimate their own feasible sets, during which they incur linear regret. Therefore, after striking a trade-off between the exploration and optimization phases, we prove (Theorem 3) a dynamic regret bound of $O(T^{2/3}\sqrt{\log T} + T^{1/3}C_T^*)$, where

$$C_T^* \triangleq \sum_{t=2}^{T} \left\|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\right\|, \tag{4}$$

is the *path-length*, defined with respect to the global minimizer sequence (Mokhtari et al., 2016; Jadbabaie et al., 2015). If the problem is centralized (single agent) and the comparator sequence is fixed, i.e., $\mathbf{x}_t^* = \mathbf{x}$, our result recovers the static regret bound of Chaudhary & Kalathil (2022).

3) We further analyze D-Safe-OGD in the *non-convex* setting. We draw upon the idea of the algorithmic equivalence between OGD and OMD (Ghai et al., 2022) to establish that in certain problem geometries (Assumptions 6-9), D-Safe-OGD can be approximated by a distributed OMD algorithm applied to a reparameterized "convex" problem. We prove that the dynamic regret is upper bounded by $O(T^{2/3}\sqrt{\log T} + T^{2/3}C_T^*)$ in Theorem 5, which is the first dynamic regret bound for a *safe distributed* algorithm in the *non-convex* setting. If the problem is centralized (single agent) and the comparator sequence is fixed, i.e., $\mathbf{x}_t^* = \mathbf{x}$, our result recovers the static regret bound of Ghai et al. (2022) (disregarding log factors).

The proofs of our results are provided in the Appendix.

2 Related Literature

I) Centralized Online Optimization: For static regret, it is well-known that the optimal regret bound is $O(\sqrt{T})$ (respectively, $O(\log T)$) for convex (respectively, exp-concave and strongly convex) problems (Hazan, 2016). For dynamic regret, various regularity measures have been considered. Let us first define the dynamic regret with respect to a general comparator sequence $\{\mathbf{u}_t\}_{t=1}^T$ and its corresponding path-length as follows

$$\begin{aligned}\mathbf{Reg}_T^d(\mathbf{u}_1,\ldots,\mathbf{u}_T) &\triangleq \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t),\\ C_T(\mathbf{u}_1,\ldots,\mathbf{u}_T) &\triangleq \sum_{t=2}^{T} \left\|\mathbf{u}_t - \mathbf{u}_{t-1}\right\|.\end{aligned} \tag{5}$$

For the dynamic regret defined in (5), Zinkevich (2003) first showed that for convex functions, OGD attains an upper bound of $O(\sqrt{T}(1 + C_T))$, and the bound was later improved to $O(\sqrt{T(1 + C_T)})$ using expert advice (Zhang et al., 2018a). For the dynamic regret defined with respect to the minimizer sequence (2), Mokhtari et al. (2016) showed a regret bound of $O(C_T^*)$ for strongly convex and smooth functions with OGD. The notion of higher-order path-length $C_{p,T}^* \triangleq \sum_{t=2}^{T} \left\|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\right\|^p$ has also been considered by several works. When the minimizer sequence $\{\mathbf{x}_t^*\}_{t=1}^T$ is uniformly bounded, $O(C_{p,T}^*)$ implies $O(C_{q,T}^*)$ for $q < p$. Zhang et al. (2017a) proved that with multiple gradient queries per round, the dynamic regret is improved to $O(\min\{C_T^*, C_{2,T}^*\})$.

Other regularity measures include the function variation $V_T \triangleq \sum_{t=2}^{T} \sup_{\mathbf{x}\in\mathcal{X}} |f_t(\mathbf{x}) - f_{t-1}(\mathbf{x})|$ (Besbes et al., 2015), the predictive path-length $C'_T(\mathbf{u}_1,\ldots,\mathbf{u}_T) \triangleq \sum_{t=2}^{T} \left\|\mathbf{u}_t - \Psi_t(\mathbf{u}_{t-1})\right\|$ (Hall & Willett, 2013), where Ψ_t is a given dynamics, and the gradient variation $D_T \triangleq \sum_{t=1}^{T} \left\|\nabla f_t(\mathbf{x}_t) - \mathbf{m}_t\right\|^2$ (Rakhlin & Sridharan, 2013), where $\mathbf{m}_t$ is a predictable sequence computed by the learner. Besbes et al. (2015) proposed a restarting OGD and showed that when the learner only has access to noisy gradients, the expected dynamic regret is bounded by $O(T^{2/3}(V_T + 1)^{1/3})$ and $O(\log T\sqrt{T(1 + V_T)})$ for convex and strongly convex functions, respectively. The above measures are not directly comparable to each other. In this regard, Jadbabaie et al. (2015) provided a dynamic regret bound in terms of C_T^*, D_T and V_T for the adaptive optimistic OMD algorithm. Also, Chang & Shahrampour (2021) revisited OGD with multiple queries in the unconstrained setup and established the regret bound of $O(\min\{V_T, C_T^*, C_{2,T}^*\})$ for strongly convex and smooth functions. Dynamic regret has also been studied for functions with a parameterizable structure (Ravier et al., 2019) as well as composite convex functions (Ajalloeian et al., 2020). Besides the dynamic regret, a relevant regret measure called *adaptive* regret (Hazan & Seshadhri, 2009) is defined as follows

$$\mathbf{Reg}_T^a(T_{sub}) \triangleq \max_{[i,i+T_{sub}-1]\subset[T]} \left(\sum_{t=i}^{i+T_{sub}-1} f_t(\mathbf{x}_t) - \min_{\mathbf{x}\in\mathcal{X}} \sum_{t=i}^{i+T_{sub}-1} f_t(\mathbf{x}) \right). \tag{6}$$

Zhang et al. (2018b) analyzed the connection between adaptive regret and dynamic regret and provided adaptive algorithms with provably small dynamic regret for convex, exponentially concave, and strongly convex functions.

In contrast to aforementioned works, where a projection operator is needed, Wan et al. (2021) proposed a projection free online method replacing the projection step with multiple linear optimization steps. Without assuming smoothness, they proved dynamic regret bounds of $O(\max\{T^{2/3}V_T^{1/3}, \sqrt{T}\})$ and $O(\max\{\sqrt{TV_T \log T}, \log T\})$ for convex and strongly convex functions, respectively. On the other hand, Wan et al. (2023) considered the case of smooth convex functions and improved the dynamic regret bound from $O\big(\sqrt{T}(1 + V_T + \sqrt{D_T})\big)$ to $O\big(\sqrt{T(1 + V_T)}\big)$.

II) Distributed Online Optimization: Yan et al. (2012) studied distributed OGD for online optimization and proved a static regret bound of $O(\sqrt{T})$ (respectively, $O(\log T)$) for convex (respectively, strongly-convex) functions. Distributed online optimization for time-varying network structures was then considered in (Mateos-Núnez & Cortés, 2014; Akbari et al., 2015; Hosseini et al., 2016). Shahrampour & Jadbabaie (2018) proposed a distributed OMD algorithm with a dynamic regret bound of $O(\sqrt{T}(1 + C_T^*))$. Dixit et al. (2019a) considered time-varying network structures and showed that distributed proximal OGD achieves a dynamic regret of $O(\log T(1 + C_T^*))$ for strongly convex functions. Zhang et al. (2019) developed a method based on gradient tracking and derived a regret bound in terms of C_T^* and a gradient path-length. More recently, Eshraghi & Liang (2022) showed that dynamic regret for strongly convex and smooth functions can be improved to $O(1 + C_T^*)$ using both primal and dual information boosted with multiple consensus steps. The non-convex case is also recently studied by Lu & Wang (2021), where the regret is characterized by the first-order optimality condition. On the other hand, for the projection free setup, Zhang et al. (2017b) presented a distributed online conditional gradient algorithm with a static regret bound of $O(T^{3/4})$ and a communication complexity of $O(T)$. The communication complexity was further improved to $O(\sqrt{T})$ in (Wan et al., 2020).

Nevertheless, the works mentioned in **(I)** and **(II)** do not consider neither long-term nor safety constraints, which are discussed next.

Table 1: Related works on centralized and distributed constrained online optimization for general functions with regret and constraint violation (CV) guarantees. Let $g(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_n(\mathbf{x}))^\top$ be the vector formed by n convex constraints. Let $c \in (0, 1)$, $\alpha_0 > 1$ and $[a]_+ = \max\{0, a\}$. Problem type 'C' stands for centralized, 'D' stands for distributed (or decentralized), 'CNX' stands for convex cost functions, and 'N-CNX' stands for non-convex cost functions. Notes: † : The CV bound in (Yu & Neely, 2020) can be further reduced to $O(1)$ under a Slater's condition assumption.

CV Type	Problem	Reference	Regret Type	Regret Bound	CV Bound
$\sum_{t=1}^T g_i(\mathbf{x}_t)\ \forall i \in [n]$	C, CNX	(Mahdavi et al., 2012)	Static	$O(\sqrt{T})$	$O(T^{3/4})$
$\sum_{t=1}^T \max_{i\in[n]} g_i(\mathbf{x}_t)$	C, CNX	(Jenatton et al., 2016)	Static	$O(T^{\max\{c,1-c\}})$	$O(T^{1-c/2})$
$\sum_{t=1}^T [g_i(\mathbf{x}_t)]_+\ \forall i \in [n]$	C, CNX	(Yuan & Lamperski, 2018)	Static	$O(T^{\max\{c,1-c\}})$	$O(T^{1-c/2})$
$\sum_{t=1}^T \Vert [g(\mathbf{x}_t)]_+ \Vert$	C, CNX	(Yi et al., 2021)	Static	$O(T^{\max\{c,1-c\}})$	$O(T^{(1-c)/2})$
$\sum_{t=1}^T \Vert [g(\mathbf{x}_t)]_+ \Vert$	C, CNX	(Yi et al., 2021)	Dynamic	$O(\sqrt{T(1+C_T)})$	$O(\sqrt{T})$
$\sum_{t=1}^T g_i(\mathbf{x}_t)\ \forall i \in [n]$	C, CNX	(Yu & Neely, 2020)	Static	$O(\sqrt{T})$	$O(T^{1/4})^\dagger$
$\left[\sum_{t=1}^T g_i(\mathbf{x}_{j,t})\right]_+,\ i = 1, \forall j \in [m]$	D, CNX	(Yuan et al., 2017)	Static	$O(T^{1/2+c/2})$	$O(T^{1-c/4})$
$\sum_{t=1}^T \sum_{j=1}^m \sum_{i=1}^n \left[\mathbf{a}_i^\top \mathbf{x}_{j,t}\right]_+$	D, CNX	(Yuan et al., 2020)	Static	$O(\sqrt{T})$	$O(T^{3/4})$
$\sum_{t=1}^T \sum_{j=1}^m \sum_{i=1}^n [g_i(\mathbf{x}_{j,t})]_+$	D, CNX	(Yuan et al., 2021)	Static	$O(T^{\max\{c,1-c\}})$	$O(T^{1-c/2})$
$\frac{1}{m}\sum_{i=1}^m \sum_{t=1}^T \left\Vert [g_t(\mathbf{x}_{i,t})]_+ \right\Vert$	D, CNX	(Yi et al., 2022)	Dynamic	$O((\alpha_0^2 T^{1-c} + T^c(1 + C_T))/\alpha_0)$	$O(\sqrt{(\alpha_0+1)T^{2-c}})$
$\sum_{t=1}^T \Vert[\mathbf{A}\mathbf{x}_t - \mathbf{b}]_+\Vert$	C, CNX	(Chaudhary & Kalathil, 2022)	Static	$O(\sqrt{\log(T)}T^{2/3})$	0
$\sum_{t=1}^T \sum_{j=1}^m \Vert[\mathbf{A}\mathbf{x}_{j,t} - \mathbf{b}]_+\Vert$	D, CNX	**This work**	Dynamic	$O\left(T^{2/3}\sqrt{\log T} + T^{1/3}C_T^*\right)$	0
$\sum_{t=1}^T \sum_{j=1}^m \Vert[\mathbf{A}\mathbf{x}_{j,t} - \mathbf{b}]_+\Vert$	D, N-CNX	**This work**	Dynamic	$O\left(T^{2/3}\sqrt{\log T} + T^{2/3}C_T^*\right)$	0

III) Constrained Online Optimization: Practical systems come with inherent system imposed constraints on the decision variable. Some examples of such constraints are inventory/budget constraints in one-way trading problem (Lin et al., 2019) and time coupling constraints in networked distributed energy systems (Fan et al., 2020). For a known constraint set, projecting the decisions back to the constraint set is a natural way to incorporate constraints in online convex optimization (OCO) with projected OGD for general cost functions achieving $O(\sqrt{T})$ static regret. However, for complex constraints, projection can induce computational burden. An early work by Hazan & Kale (2012) solves the constrained optimization problem by replacing the quadratic convex program with simpler linear program using Frank-Wolfe. For understanding the following references better, let $\mathcal{X} = \{\mathbf{x} \in \mathrm{X} \subseteq \mathrm{R}^d : g(\mathbf{x}) \leq 0\}$ where $g(\mathbf{x}) = (g_1(\mathbf{x}), g_2(\mathbf{x}), \ldots, g_n(\mathbf{x}))^\top$ is the vector formed by n convex constraints, and X is a closed convex set. The work by Mahdavi et al. (2012) proposed to use a simpler closed form projection in place of true desired projection attaining $O(\sqrt{T})$ static regret with $\sum_{t=1}^{T} g_i(\mathbf{x}_t) \leq O(T^{3/4})$ constraint violation $\forall i \in [n]$. Thus, their method achieves optimal regret with lesser computation burden at the cost of incurring constraint violations. The follow up work by Jenatton et al. (2016) proposed adaptive step size variant of Mahdavi et al. (2012) with $O\left(T^{\max\{c,1-c\}}\right)$ static regret and $O\left(T^{1-c/2}\right)$ constraint violation for $c \in (0,1)$. These bounds were further improved in Yu & Neely (2020) with a static regret bound of $O(\sqrt{T})$ and constraint violation bound of $O(T^{1/4})$. Here, the constraint violation is further reduced to $O(1)$ when $g_i(\mathbf{x})$ satisfy Slater's condition. The work by Yuan & Lamperski (2018) considered stricter 'cumulative' constraint violations of the form $\sum_{t=1}^{T} [g_i(\mathbf{x}_t)]_+ \ \forall i \in [n]$ and proposed algorithms with $O\left(T^{\max\{c,1-c\}}\right)$ static regret and $O\left(T^{1-c/2}\right)$ 'cumulative' constraint violation for $c \in (0,1)$. For strongly convex functions, Yuan & Lamperski (2018) proved $O(\log(T))$ static regret and the constraint violation of respective form is $O(\sqrt{\log(T)T})$. More recently, the work by Yi et al. (2021) proposed an algorithm with $O\left(T^{\max\{c,1-c\}}\right)$ regret and $\sum_{t=1}^{T} \left\|[g(\mathbf{x}_t)]_+\right\| \leq O\left(T^{(1-c)/2}\right)$ 'cumulative' constraint violation. For strongly convex functions, Yi et al. (2021) reduced both static regret and constraint violation bounds to $O(\log(T))$. Further, Yi et al. (2021) presented a bound of $O(\sqrt{T(1+C_T)})$ for dynamic regret with an $O(\sqrt{T})$ 'cumulative' constraint violation. The algorithms in Mahdavi et al. (2012); Jenatton et al. (2016); Yu & Neely (2020); Yuan & Lamperski (2018); Yi et al. (2021) employ some flavor of online primal-dual algorithms. A series of recent works (Sun et al., 2017; Chen et al., 2017; Neely & Yu, 2017; Yu et al., 2017; Cao & Liu, 2018; Liu et al., 2022) have also dealt with time-varying constraints. Yu et al. (2017) specifically work with 'stochastic' time varying constraints.

More recently works in (Yuan et al., 2017; 2020; 2021; Yi et al., 2022) have looked at distributed OCO with long term constraints. The work by Yuan et al. (2017) proposed a consensus based primal-dual sub-gradient algorithm with $O(T^{1/2+\beta_0})$ regret and $O(T^{1-\beta_0/2})$ constraint violation for $\beta_0 \in (0, 0.5)$. Single constraint function was considered in (Yuan et al., 2017), where constraint violation is of the form $[\sum_{t=1}^{T} g_i(\mathbf{x}_{j,t})]_+,\ i = 1, \forall j \in [m]$. Yuan et al. (2020) proposed algorithms for distributed online linear regression with $O(\sqrt{T})$ regret and $O(T^{3/4})$ constraint violation. Here, constraint violation takes the form $\sum_{t=1}^{T}\sum_{j=1}^{m}\sum_{i=1}^{n}\left[\mathbf{a}_i^\top \mathbf{x}_{j,t}\right]_+$, where $\mathbf{a}_i, i \in [n]$ is a constraint vector. Another primal-dual algorithm was presented in Yuan et al. (2021) with $O(T^{\max\{1-c,c\}})$ regret and $O(T^{1-c/2})$ constraint violation of the form $\sum_{t=1}^{T}\sum_{j=1}^{m}\sum_{i=1}^{n}[g_i(\mathbf{x}_{j,t})]_+$ for $c \in (0,1)$. In all of (Yuan et al., 2017; 2020; 2021) constraint functions are known a priori. More recently, Yi et al. (2022) proposed algorithms for distributed OCO with time-varying constraints, and for stricter 'network' constraint violation metric of the form $\frac{1}{m}\sum_{i=1}^{m}\sum_{t=1}^{T}\left\|[g_t(\mathbf{x}_{i,t})]_+\right\|$. The algorithm in (Yi et al., 2022) gives a dynamic regret of $O((\alpha_0^2 T^{1-c} + T^c(1+C_T))/\alpha_0)$ with $O(\sqrt{(\alpha_0+1)T^{2-c}})$ constraint violation for $\alpha_0 > 1$ and $c \in (0,1)$. Additionally, constrained distributed OCO with coupled inequality constraints is considered in (Yi et al., 2020a;b); with bandit feedback on cost function is considered in (Li et al., 2020); with partial constraint feedback is studied in (Sharma et al., 2021). For more references in this problem space, we refer the readers to the survey in (Li et al., 2022).

IV) Safe Online Optimization: Safe online optimization is a fairly nascent field with only a few works studying per-time safety in optimization problems. Amani et al. (2019); Khezeli & Bitar (2020) study the problem of safe linear bandits giving $O(\log(T)\sqrt{T})$ regret with no constraint violation, albeit under an assumption that a lower bound on the distance between the optimal action and safe set's boundary is known. Without the knowledge of such a lower bound, Amani et al. (2019) show $O(\log(T)T^{2/3})$ regret. Safe convex and non-convex optimization is studied in (Usmanova et al., 2019; Fereydounian et al., 2020). Safety in the context of OCO is studied in Chaudhary & Kalathil (2022) with a regret of $O(\sqrt{\log(T)}T^{2/3})$.

Remark 1. *Different from the works listed above, we study the problem of safe distributed online optimization with unknown linear constraints. We consider both convex and non-convex cost functions.*

3 Preliminaries

3.1 Notation

$[m]$	The set $\{1, 2, \ldots, m\}$ for any integer m
$\lVert \cdot \rVert_F$	Frobenius norm of a matrix
$\lVert \mathbf{X} \rVert_{\mathbf{V}}$	$\sqrt{\text{trace}(\mathbf{X}^\top \mathbf{V}\mathbf{X})}$, for a matrix $\mathbf{X}$ and a positive-definite matrix $\mathbf{V}$
$\Pi_{\mathcal{X}}[\cdot]$	The operator for the projection to set $\mathcal{X}$
$[\mathbf{A}]_{ij}$	The entry in the i-th row and j-th column of $\mathbf{A}$
$[\mathbf{A}]_{i,:}$	The i-th row of $\mathbf{A}$
$[\mathbf{A}]_{:,j}$	The j-th column of $\mathbf{A}$
$\mathbf{1}$	The vector of all ones
$\mathbf{e}_i$	The i-th basis vector
$J_f(\mathbf{x})$	The Jacobian of a mapping $f(\cdot)$ at $\mathbf{x}$

3.2 Strong Convexity and Bregman Divergence

Definition 1. *A function $f : \mathcal{X} \to \mathrm{R}$ is μ-strongly convex ($\mu > 0$) over the convex set $\mathcal{X}$ if*

$$f(\mathbf{x}) \geq f(\mathbf{y}) + \nabla f(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}) + \frac{\mu}{2} \lVert \mathbf{x} - \mathbf{y} \rVert^2, \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}.$$

Definition 2. *For a strongly convex function $\phi(\cdot)$, the Bregman divergence w.r.t. $\phi(\cdot)$ over $\mathcal{X}$ is defined as*

$$\mathcal{D}_\phi(\mathbf{x}, \mathbf{y}) \triangleq \phi(\mathbf{x}) - \phi(\mathbf{y}) - \nabla \phi(\mathbf{y})^\top (\mathbf{x} - \mathbf{y}), \quad \mathbf{x}, \mathbf{y} \in \mathcal{X}.$$

3.3 Network Structure

The underlying network topology is governed by a symmetric doubly stochastic matrix $\mathbf{P}$, i.e., $[\mathbf{P}]_{ij} \geq 0, \forall i, j \in [m]$, and each row (or column) is summed to one. If $[\mathbf{P}]_{ij} > 0$, agents i and j are considered neighbors, and agent i assigns the weight $[\mathbf{P}]_{ij}$ to agent j when they communicate with each other. We assume that the graph structure captured by $\mathbf{P}$ is connected, i.e., there is a (potentially multi-hop) path from any agent i to another agent $j \neq i$. Each agent is considered as a neighbor of itself, i.e., $[\mathbf{P}]_{ii} > 0$ for any $i \in [m]$. These constraints on the communication protocol imply a geometric mixing bound for $\mathbf{P}$ (Liu, 2008), such that $\sum_{j=1}^{m} \left| [\mathbf{P}^k]_{ji} - 1/m \right| \leq \sqrt{m}\beta^k$, for any $i \in [m]$, where β is the second largest singular value of $\mathbf{P}$.

Remark 2. *In all of the algorithms proposed in the paper, we will see $\mathbf{P}$ as an input. This does not contradict the decentralized nature of the algorithms, as agent i only requires the knowledge of $[\mathbf{P}]_{ji} > 0$ for any j in its neighborhood. The knowledge of $\mathbf{P}$ is not global, and each agent only has local information about it.*

4 Safe Set Estimation

To keep the regret small, we first need to identify the linear safety constrains. It is impossible to learn the safety constraints if the algorithm receives no information that can be used to estimate the unknown constraints (Chaudhary & Kalathil, 2022). In our problem setup, we assume that the algorithm receives noisy observations of the form

$$\hat{\mathbf{x}}_{i,t} = \mathbf{A}\mathbf{x}_{i,t} + \mathbf{w}_{i,t} \qquad \forall i \in [m],$$

at every time step t, where the nature of noise $\mathbf{w}_{i,t}$ is described below. Here, $\mathbf{A} \in \mathrm{R}^{n \times d}$, $\mathbf{b} \in \mathrm{R}^n$, and n is the number of constraints. Note that all agent updates are synchronous.

4.1 Assumptions

We make the following assumptions common to both the convex and the non-convex problem settings.

Assumption 1. *The set $\mathcal{X}^s$ is a closed polytope, hence, convex and compact. Also, $\lVert \mathbf{x} \rVert \leq L, \forall \mathbf{x} \in \mathcal{X}^s$, and* $\max_{i \in [n]} \lVert [\mathbf{A}]_{i,:} \rVert_2 \leq L_A$.

Assumption 2. *The constraint noise sequence* $\{\mathbf{w}_{i,t}, t \in [T]\}$ *is R-sub-Gaussian with respect to the filtration* $\{\mathcal{F}_{i,t}, t \in [T]\}$*, i.e.,* $\forall t \in [T], \forall i \in [m]$*,* $\mathbb{E}[\mathbf{w}_{i,t}|\mathcal{F}_{i,t-1}] = 0$ *and we have for any* $\sigma \in \mathrm{R}$

$$\mathbb{E}[\exp(\sigma \mathbf{x}^\top \mathbf{w}_{i,t}) \,|\, \mathcal{F}_{t-1}] \leq \exp(\sigma^2 R^2 \|\mathbf{x}\|^2/2).$$

Assumption 3. *Every agent has knowledge of a safe baseline action* $\mathbf{x}^s \in \mathcal{X}^s$ *such that* $\mathbf{A}\mathbf{x}^s = \mathbf{b}^s < \mathbf{b}$*. The agents are aware of* $\mathbf{x}^s$ *and* $\mathbf{b}^s$ *and thus, the safety gap* $\Delta^s \triangleq \min_{i\in[n]}(b_i - b_i^s)$*, where* b_i *(respectively,* b_i^s*) denotes the* i*-th element of* $\mathbf{b}$ *(respectively,* $\mathbf{b}^s$*).*

The first assumption is typical to online optimization, and the second assumption on the noise is standard. The third assumption stems from the requirement to be absolutely safe at every time step. The assumption warrants the need for a safe starting point which is readily available in most practical problems of interest. Similar assumptions can be found in previous literature on safe linear bandits (Amani et al., 2019; Khezeli & Bitar, 2020), safe convex and non-convex optimization (Usmanova et al., 2019; Fereydounian et al., 2020), and safe online convex optimization (Chaudhary & Kalathil, 2022).

4.2 Explore and Estimate

In this section, we present an algorithmic subroutine, Algorithm 1, for agents to obtain sufficiently good local estimates of $\mathcal{X}^s$ before beginning to perform OGD. For the first T_0 time steps, each agent safely explores around the baseline action $\mathbf{x}^s$. Each exploratory action is a γ-weighted combination of the baseline action and an i.i.d random vector $\zeta_{i,t}$. Here, for the agent $i \in [m]$ at time step $t \in [T_0]$, $\gamma \in [0, 1)$, and $\zeta_{i,t}$ is zero mean i.i.d random vector with $\|\zeta_{i,t}\| \leq L$ and $Cov(\zeta_{i,t}) = \sigma_\zeta^2 \mathbf{I}$. Performing exploration in this manner ensures per time step safety requirement as noted in Lemma 1. The proof of lemma is immediate from the assumptions.

Lemma 1. *(Lemma 1 in Chaudhary & Kalathil (2022)) Let Assumptions 1-3 hold. With* $\gamma = \frac{\Delta^s}{LL_A}$*,* $\mathbf{A}\mathbf{x}_{i,t} \leq \mathbf{b}$ *for each* $\mathbf{x}_{i,t} = (1-\gamma)\mathbf{x}^s + \gamma\zeta_{i,t}$ $\forall i \in [m], t \in [T_0]$*.*

Once the data collection phase is finished, each agent $i \in [m]$ constructs local function $l_i(\mathbf{A})$ of the form

$$l_i(\mathbf{A}) \triangleq \sum_{t=1}^{T_0} \|\mathbf{A}\mathbf{x}_{i,t} - \hat{\mathbf{x}}_{i,t}\|^2 + \frac{\lambda}{m}\|\mathbf{A}\|_F^2.$$

Then, for time steps $t \in [T_0 + 1, T_0 + T_1]$, Alg. **EXTRA** (Shi et al., 2015) is used to solve the global Least Squares (LS) estimation problem $\sum_{i=1}^m l_i(\mathbf{A})$ in a distributed fashion with a proper choice of α.

Lemma 2. *Suppose Assumptions 1-2 hold. Let Algorithm 1 run with* $T_0 = \Omega(\frac{L^2}{m\gamma^2\sigma_\zeta^2}\log(\frac{d}{\delta}))$ *for data collection and* $T_1 = \Theta(\log T^\rho)$*, where* ρ *is a positive constant. Denote the final output of the algorithm as* $\widehat{\mathbf{A}}_i$ *for agent* $i \in [m]$*. Then, with probability at least* $(1-2\delta)$*, we have* $\forall k \in [n]$ *and* $\forall i, j \in [m]$

$$\begin{aligned}\|[\widehat{\mathbf{A}}_i]_{k,:} - [\mathbf{A}]_{k,:}\| &\leq \frac{1}{T^\rho} + \frac{R\sqrt{d\log\left(\frac{1+mT_0L^2/\lambda}{\delta/n}\right)} + \sqrt{\lambda}L_A}{\sqrt{\frac{1}{2}m\gamma^2\sigma_\zeta^2 T_0}}, \\ \|[\widehat{\mathbf{A}}_i]_{k,:} - [\widehat{\mathbf{A}}_j]_{k,:}\| &\leq \frac{2}{T^\rho},\end{aligned} \tag{7}$$

where $[\widehat{\mathbf{A}}_i]_{k,:}$ *and* $[\mathbf{A}]_{k,:}$ *are the* k*-th rows of* $\widehat{\mathbf{A}}_i$ *and* $\mathbf{A}$*, respectively.*

It is worth noting that the safety gap Δ^s affects the estimation error according to Lemma 2. As we mentioned earlier, the exploratory action $\mathbf{x}_{i,t} = (1-\gamma)\mathbf{x}^s + \gamma\zeta_{i,t}$, where the coefficient $\gamma = \frac{\Delta^s}{LL_A}$. Intuitively, if Δ^s is larger, we can put more weight on the exploration through $\zeta_{i,t}$, which is beneficial to the estimation accuracy. We see from Equation (7) that when γ is larger, the estimation error bound is tighter.

Let us also discuss the computational complexity of Algorithm 1. The time cost of the data-collection phase is $O(mdT_0)$, assuming that it takes $O(d)$ time to compute each action. For the estimation phase, to perform a single update, each agent spends $O(mnd)$ time for the calculation of the weighted average and $O(T_0nd)$ time for the gradient computation. Accordingly, the total time cost of Algorithm 1 is $O\big(mT_1(mnd + T_0nd)\big)$.

Algorithm 1 Distributed Constraint Parameter Estimation

Require: number of agents m, doubly stochastic matrix $\mathbf{P} \in \mathrm{R}^{m\times m}$, $\tilde{\mathbf{P}} \triangleq \frac{\mathbf{I}+\mathbf{P}}{2}$, hyper-parameters α, γ and λ, data-collection duration T_0, constraint-estimation duration T_1, a strictly feasible point $\mathbf{x}^s$ (safe baseline action).
Explore around baseline action:
for $t = 1, 2, \dots, T_0$ **do**
  for $i = 1, 2, \dots, m$ **do**
    Select action $\mathbf{x}_{i,t} = (1-\gamma)\mathbf{x}^s + \gamma\zeta_{i,t}$
    Receive noisy observation $\hat{\mathbf{x}}_{i,t} = \mathbf{A}\mathbf{x}_{i,t} + \mathbf{w}_{i,t}$
  end for
end for
Form local functions using collected data:

$$l_i(\mathbf{A}) \triangleq \sum_{t=1}^{T_0} \left\|\mathbf{A}\mathbf{x}_{i,t} - \hat{\mathbf{x}}_{i,t}\right\|^2 + \frac{\lambda}{m}\left\|\mathbf{A}\right\|_F^2.$$

Use EXTRA (Shi et al., 2015) **to solve global LS problem** $\sum_{i=1}^m l_i(\mathbf{A})$ **in a distributed fashion:**
Randomly generate $\widehat{\mathbf{A}}_i^{T_0} \in \mathrm{R}^{n\times d}$ for all $i \in [m]$.
$\forall i \in [m]$, $\widehat{\mathbf{A}}_i^{T_0+1} = \sum_{j=1}^m [\mathbf{P}]_{ji}\widehat{\mathbf{A}}_j^{T_0} - \alpha\nabla l_i(\widehat{\mathbf{A}}_i^{T_0})$, where the gradient is computed based on the following expression:

$$\nabla l_i(\mathbf{A}) = \sum_{t=1}^{T_0}\left[2\mathbf{A}\mathbf{x}_{i,t}\mathbf{x}_{i,t}^\top - 2\hat{\mathbf{x}}_{i,t}\mathbf{x}_{i,t}^\top\right] + \frac{2\lambda}{m}\mathbf{A}.$$

for $t = T_0, \dots, T_0 + T_1 - 2$ **do**
  for $i = 1, 2, \dots, m$ **do**
    $\widehat{\mathbf{A}}_i^{t+2} = \sum_{j=1}^m 2[\tilde{\mathbf{P}}]_{ji}\widehat{\mathbf{A}}_j^{t+1} - \sum_{j=1}^m [\tilde{\mathbf{P}}]_{ji}\widehat{\mathbf{A}}_j^{t} - \alpha[\nabla l_i(\widehat{\mathbf{A}}_i^{t+1}) - \nabla l_i(\widehat{\mathbf{A}}_i^{t})]$.
  end for
end for

Let us now define the estimated safe set for each agent $i \in [m]$. Let the parameter estimate for agent $i \in [m]$ at the end of $T_0 + T_1$ time step be denoted by $\widehat{\mathbf{A}}_i$. For each row $k \in [n]$ of $\widehat{\mathbf{A}}_i$, a ball centered at $[\widehat{\mathbf{A}}_i]_{k,:}$ with a radius of $\mathcal{B}_r$ can be defined as follows

$$\mathcal{C}_{i,k} \triangleq \{\mathbf{a} \in \mathrm{R}^d : \left\|\mathbf{a} - [\widehat{\mathbf{A}}_i]_{k,:}\right\| \le \mathcal{B}_r\}, \tag{8}$$

where $\mathcal{B}_r \triangleq \frac{1}{T^\rho} + \frac{R\sqrt{d\log\left(\frac{1+mT_0L^2/\lambda}{\delta/n}\right)}+\sqrt{\lambda}L_A}{\sqrt{\frac{1}{2}m\gamma^2\sigma_\zeta^2T_0}}$. The true parameter $[\mathbf{A}]_{k,:}$ lies inside the set $\mathcal{C}_{i,k}$ with a high probability. Now, using (8) the safe estimated set for agent $i \in [m]$ can be constructed as follows:

$$\widehat{\mathcal{X}}_i^s \triangleq \{\mathbf{x} \in \mathrm{R}^d : \tilde{\mathbf{a}}_k^\top\mathbf{x} \le b_k,\ \forall\tilde{\mathbf{a}}_k \in \mathcal{C}_{i,k},\ \forall k \in [n]\}. \tag{9}$$

The safe estimated set above will be used by each agent for the projection step in subsequent algorithms.

5 Dynamic Regret Analysis for the Convex Setting

During the first T_0+T_1 time steps in Algorithm 1 agents do not expend any effort to minimize the regret. This is due to the fact that without the knowledge of the feasible set, they cannot perform any projection. In this section, we propose D-Safe-OGD, which allows agents to carry out a safe distributed online optimization, and we analyze D-Safe-OGD in the *convex* setting.

D-Safe-OGD is summarized in Algorithm 2, where in the exploration phase, all agents collaboratively estimate the constraint parameters based on Algorithm 1, and then each agent constructs the feasible set based on its own estimate. In the optimization phase, the network applies distributed OGD, where all agents first perform gradient descent with their local gradients, and then they communicate their iterates with neighbors based on the network topology imposed by $\mathbf{P}$. We note that the projection operator of each agent is defined w.r.t. the local estimated feasible set (line 8 of Algorithm 2), thereby making the feasible sets close enough but slightly different from each other. Therefore, previous regret bounds for distributed online optimization over a common feasible set (e.g., (Yan et al., 2012; Hosseini et al., 2016; Shahrampour & Jadbabaie, 2018; Eshraghi & Liang, 2022)) are not immediately applicable. We tackle this challenge by exploiting the geometric property of linear constraints (Fereydounian et al., 2020) as well as the sensitivity analysis of perturbed optimization problems with second order regular constraints (Bonnans et al., 1998), and we present an upper bound on the dynamic regret in terms of the path-length regularity measure.

We adhere to the following standard assumption in the context of OCO:

Assumption 4. *The cost functions $f_{i,t}$ are convex $\forall i \in [m]$ and $\forall t \in [T]$, and they have a bounded gradient, i.e., $\|\nabla f_{i,t}(\mathbf{x})\| \leq G$ for any $\mathbf{x} \in \mathcal{X}^s$.*

Algorithm 2 Distributed Safe OGD with linear constraints

1: **Require:** number of agents m, doubly stochastic matrix $\mathbf{P} \in \mathrm{R}^{m \times m}$, hyper-parameters $\alpha, \gamma, \eta, \delta, \lambda$, time horizon T, a strictly feasible point $\mathbf{x}^s$.
2: Specify T_0 and T_1 based on given hyper-parameters and run Algorithm 1 to learn agents estimates $\{\hat{\mathbf{A}}_i\}_{i \in [m]}$ in a distributed fashion.
3: For all $i \in [m]$, construct the safe set $\widehat{\mathcal{X}}_i^s$ from the estimate $\widehat{\mathbf{A}}_i$.
4: **Distributed online gradient descent over different feasible sets:**
5: Let $T_s \triangleq (T_0 + T_1 + 1)$ and initialize all agents at the same point $\mathbf{x}_{i,T_s} = \mathbf{x}_{T_s}$ chosen randomly.
6: **for** $t = T_s, \ldots, T$ **do**
7:     **for** $i = 1, 2, \ldots, m$ **do**
8:
$$\mathbf{y}_{i,t} = \Pi_{\widehat{\mathcal{X}}_i^s} \left[\mathbf{x}_{i,t} - \eta \nabla f_{i,t}(\mathbf{x}_{i,t})\right].$$
9:     **end for**
10:     For all $i \in [m]$,
11:
$$\mathbf{x}_{i,t+1} = \sum_{j=1}^{m} [\mathbf{P}]_{ji} \mathbf{y}_{j,t}.$$
12: **end for**

Theorem 3. *Suppose Assumptions 1-4 hold and $T = \Omega\left(\left(\frac{L^2}{m\gamma^2\sigma_\zeta^2}\log(\frac{d}{\delta})\right)^{3/2}\right)$. By running Algorithm 2 with $\gamma \leq \frac{\Delta^s}{LL_A}$, $\eta = \Theta(T^{-1/3})$, $T_0 = \Theta(T^{2/3})$ and $T_1 = \Theta(\log T)$, we have with probability at least $(1 - 2\delta)$*

$$\mathbf{x}_{i,t} \in \mathcal{X}^s, \ \forall i \in [m], t \in [T], \quad \textit{and}$$

$$\mathbf{Reg}_{i,T}^d = O\left(T^{2/3}\sqrt{\log(T/\delta)} + \frac{\beta}{(1-\beta)}T^{2/3} + T^{1/3}C_T^*\right), \ \forall i \in [m].$$

Theorem 3 establishes a dynamic regret bound for D-Safe-OGD that is at least $O(T^{2/3}\sqrt{\log T})$, and for a large enough path-length the bound becomes $O(T^{1/3}C_T^*)$. We can also see the impact of network topology through β, the second largest singular value of $\mathbf{P}$. When the network connectivity is stronger (i.e., β is smaller), the regret bound is tighter. For the dependence on other parameters, we refer readers to the exact upper bound expression (Equation (38)).

Corollary 4. *Suppose that the comparator sequence is fixed over time, i.e., $\mathbf{x}_t^* = \mathbf{x}^*, \ \forall t \in [T]$. Then, the individual regret bound is $O(T^{2/3}\sqrt{\log T})$, which recovers the static regret bound of the centralized case in (Chaudhary & Kalathil, 2022) in terms of order.*

Remark 3. *Note that when $\mathbf{A}$ is known, there is no estimation error, and the trade-off in terms of η and T_0 no longer exists. In other words, the agents do not incur the initial regret of $T_0 + T_1 = O(T^{2/3})$, caused by estimation. Then, by choosing $\eta = \Theta(\frac{1}{\sqrt{T}})$, the resulting bound is $O(\sqrt{T}(1 + C_T^*))$, which recovers the result of Shahrampour & Jadbabaie (2018) in terms of order.*

6 Dynamic Regret Analysis for the Non-convex Setting

In this section, we study the *non-convex* setting for safe distributed online optimization. Even for offline optimization in the non-convex setting, the standard metric for the convergence analysis is often stationarity, i.e., characterizing the decay rate of the gradient norm. In online optimization, we can also expect that standard regret notions, used in the convex setting, may not be tractable for understanding the algorithm performance. However, in a recent work by Ghai et al. (2022), the authors studied an algorithmic equivalence property between OGD and OMD for certain problem geometries, in the sense that OGD applied to non-convex problems can be approximated by OMD applied to convex functions under reparameterization, using which a sub-linear static regret bound is guaranteed.

More specifically, for a centralized problem, suppose that there is a bijective non-linear mapping q, such that $\mathbf{u}_t = q(\mathbf{x}_t)$, and consider the OGD and OMD updates

Centralized OGD:

$$\mathbf{x}_{t+1} = \text{argmin}_{\mathbf{x}\in\mathcal{X}} \left\{\nabla f_t(\mathbf{x}_t)^\top(\mathbf{x}-\mathbf{x}_t) + \tfrac{1}{2\eta}\left\|\mathbf{x}-\mathbf{x}_t\right\|^2\right\}, \tag{10}$$

Centralized OMD:

$$\mathbf{u}_{t+1} = \text{argmin}_{\mathbf{u}\in\mathcal{X}'} \left\{\nabla \tilde{f}_t(\mathbf{u}_t)^\top(\mathbf{u}-\mathbf{u}_t) + \tfrac{1}{\eta}\mathcal{D}_\phi(\mathbf{u},\mathbf{u}_t)\right\}, \tag{11}$$

where $\mathcal{X}'$ is the image of $\mathcal{X}$ under the mapping q. Ghai et al. (2022) quantified the deviation $\left\|\mathbf{u}_{t+1} - q(\mathbf{x}_{t+1})\right\|$ as $O(\eta^{3/2})$ under the following technical assumptions (together with boundedness of gradient norms):

Assumption 5. *There exists a bijective mapping $q : \mathcal{X} \to \mathcal{X}'$ such that $[\nabla^2\phi(\mathbf{u})]^{-1} = J_q(\mathbf{x})J_q(\mathbf{x})^\top$ where $\mathbf{u} = q(\mathbf{x})$.*

Assumption 6. *Let $W > 1$ be a constant. Assume that $q(\cdot)$ is W-Lipschitz, $\phi(\cdot)$ is 1-strongly convex and smooth with its first and third derivatives upper bounded by W. The first and second derivatives of $q^{-1}(\cdot)$ are also bounded by W. For all $\mathbf{u} \in \mathcal{X}'$, $\mathcal{D}_\phi(\mathbf{u},\cdot)$ is W-Lipschitz over $\mathcal{X}'$.*

Examples that satisfy these assumptions are provided in Section 3.1 of Ghai et al. (2022). For example, if ϕ is the negative entropy (respectively, log barrier), we can use quadratic (respectively, exponential) reparameterization for q. Amid & Warmuth (2020) showed that in the continuous-time setup when Assumption 5 holds, the mirror descent regularization induced by ϕ can be transformed back to the Euclidean regularization by q^{-1}, which implies the equivalence between OMD for convex functions and OGD for non-convex functions. This is due to the fact that higher than second order factors vanish in continuous time, and this assures that the mirror flow and the reparameterized gradient flow coincide. Though in the discrete-time case, the exact equivalence does not hold, Ghai et al. (2022) showed that OGD for non-convex functions can still be approximated as OMD for convex functions, and the corresponding static regret bound is $O(T^{2/3})$ under the assumption that $f_t(\mathbf{x}) = \tilde{f}_t(q(\mathbf{x}))$, where $\tilde{f}_t(\cdot)$ is convex. However, we need more technical assumptions to handle the discrete-time setup as higher order terms are relevant and must be judiciously analyzed. Ghai et al. (2022) characterized the sufficient condition to achieve Assumption 5, which entails an implicit OMD reparameterization for a non-convex OGD. We state these (two assumptions) by tailoring them to our problem setting:

Assumption 7. $\left\|\nabla\tilde{f}_{i,t}(\mathbf{u})\right\| \le G_F$ *for all* $\mathbf{u}\in\mathcal{X}^{s\prime}$ *and* $\sup_{\mathbf{u},\mathbf{z}\in\mathcal{X}^{s\prime}}\mathcal{D}_\phi(\mathbf{u},\mathbf{z}) \le D'$.

Assumption 8. *Properties of the mapping $q(\cdot)$:*

- *There exists a mapping $q(\cdot)$ such that $f_{i,t}(\mathbf{x}) = \tilde{f}_{i,t}(q(\mathbf{x}))$, where $\tilde{f}_{i,t}(\cdot)$ is convex.*
- *$q(\cdot)$ is a C^3-diffeomorphism, and $J_q(\mathbf{x})$ is diagonal.*
- *For any $\mathcal{X} \subset \mathcal{X}^s$ which is compact and convex, $\mathcal{X}' \triangleq q(\mathcal{X})$ is convex and compact.*

We again refer the reader to Section 3.1 of Ghai et al. (2022) for examples related to Assumption 8.

In this work, we extend this equivalence to "distributed" variants of OGD and OMD under the additional complexity that the constraint set is unknown, and it can only be approximated via Algorithm 1. Our focus is on analyzing the effect of (i) the constraint estimation as well as (ii) the distributed setup in non-convex online learning, and we also generalize the analysis of Ghai et al. (2022) to the *dynamic* regret. For the technical analysis of the non-convex setting, we also use the following assumption.

Assumption 9. *Let* $\mathbf{u}$ *and* $\{\mathbf{y}_i\}_{i=1}^m$ *be vectors in* R^d*. The Bregman divergence satisfies the separate convexity in the following sense*

$$\mathcal{D}_\phi(\mathbf{u}, \sum_i^m \alpha_i \mathbf{y}_i) \leq \sum_i^m \alpha_i \mathcal{D}_\phi(\mathbf{u}, \mathbf{y}_i),$$

where $\alpha \in \Delta_m$ *is on the* $m-$*dimensional simplex.*

This assumption is satisfied by commonly used Bregman divergences, e.g., Euclidean distance and KL divergence. We refer interested readers to (Bauschke & Borwein, 2001; Shahrampour & Jadbabaie, 2018) for more information.

In the following theorem, we prove that with high probability, the dynamic regret bound of D-Safe-OGD is $O(T^{2/3}\sqrt{\log T} + T^{2/3}C_T^*)$.

Theorem 5. *Suppose Assumptions 1-3 and 6-9 hold and* $T = \Omega\left(\left(\frac{L^2}{m\gamma^2\sigma_\zeta^2}\log(\frac{d}{\delta})\right)^{3/2}\right)$*. By running Algorithm 2 with* $\gamma \leq \frac{\Delta^s}{LL_A}$*,* $\eta = \Theta(T^{-2/3})$*,* $T_0 = \Theta(T^{2/3})$ *and* $T_1 = \Theta(\log T)$*, we have with probability at least* $(1-2\delta)$

$$\mathbf{x}_{i,t} \in \mathcal{X}^s, \ \forall i \in [m], t \in [T], \quad \textit{and}$$

$$\mathbf{Reg}_{i,T}^d = O(T^{2/3}\sqrt{\log(T/\delta)} + T^{2/3}C_T^*), \ \forall i \in [m].$$

The complete proof is provided in the Appendix, and the dependence on other problem parameters can be found in Equation (54). The idea is to show that distributed OMD and distributed OGD iterates are close enough to each other if the reference points of both updates are identical, i.e., $\mathbf{u}_{i,t} = q(\mathbf{x}_{i,t})$ for all $i \in [m]$. Then, distributed OGD can be viewed as a perturbed version of distributed OMD, and under the assumption of convexity of $\tilde{f}_{i,t}$ the regret bound can be established. We further have the following corollary that shows our result is a valid generalization of Ghai et al. (2022) to the *distributed*, *dynamic* setting.

Corollary 6. *Suppose that the comparator sequence is static over time, i.e.,* $\mathbf{x}_t^* = \mathbf{x}^*, \ \forall t \in [T]$*. Then, the individual regret bound becomes* $O(T^{2/3}\sqrt{\log T})$*, which recovers the static regret bound of Ghai et al. (2022) up to log factors.*

It is worth noting that though in the convex case, the estimation of unknown constraints exacerbates the regret bound (due to $O(T^{2/3})$ time spent on exploration), for the non-convex case, the resulting bound still matches the static regret of Ghai et al. (2022), where there is no estimation error. In other words, there is no trade-off in this case as the static regret (without estimation error) is $O(T^{2/3})$ (Ghai et al., 2022) (disregarding log factors).

Conclusion

In this work, we considered safe distributed online optimization with an unknown set of linear constraints. The goal of the network is to ensure that the action sequence selected by each agent, which only has partial information about the global function, is competitive to the centralized minimizers in hindsight without violating the safety constraints. To address this problem, we proposed D-Safe-OGD, where starting from a safe region, it allows all agents to perform exploration to estimate the unknown constraints in a distributed fashion. Then, distributed OGD is applied over the feasible sets formed by agents estimates. For convex functions, we proved a dynamic regret bound of $O(T^{2/3}\sqrt{\log T} + T^{1/3}C_T^*)$, which recovers the static regret bound of Chaudhary & Kalathil (2022) for the centralized case (single agent). Then, we showed that for the non-convex setting, the dynamic regret is upper bounded by $O(T^{2/3}\sqrt{\log T} + T^{2/3}C_T^*)$, which recovers the static regret bound of Ghai et al. (2022) for the centralized case (single agent) up to log factors. Possible future directions include improving the regret using adaptive techniques and/or deriving comprehensive regret bounds in terms of other variation measures, such as V_T.

References

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems*, 24, 2011.

Amirhossein Ajalloeian, Andrea Simonetto, and Emiliano Dall'Anese. Inexact online proximal-gradient method for time-varying convex optimization. In *American Control Conference (ACC)*, pp. 2850–2857, 2020.

Mohammad Akbari, Bahman Gharesifard, and Tamás Linder. Distributed online convex optimization on time-varying directed graphs. *IEEE Transactions on Control of Network Systems*, 4(3):417–428, 2015.

Sanae Amani, Mahnoosh Alizadeh, and Christos Thrampoulidis. Linear stochastic bandits under safety constraints. *Advances in Neural Information Processing Systems*, 32, 2019.

Ehsan Amid and Manfred KK Warmuth. Reparameterizing mirror descent as gradient descent. *Advances in Neural Information Processing Systems*, 33:8430–8439, 2020.

Karl Johan Åström and Richard M Murray. *Feedback systems*. Princeton university press, 2010.

Heinz H Bauschke and Jonathan M Borwein. Joint and separate convexity of the bregman distance. In *Studies in Computational Mathematics*, volume 8, pp. 23–36. Elsevier, 2001.

Omar Besbes, Yonatan Gur, and Assaf Zeevi. Non-stationary stochastic optimization. *Operations research*, 63(5): 1227–1244, 2015.

J Frédéric Bonnans, Roberto Cominetti, and Alexander Shapiro. Sensitivity analysis of optimization problems under second order regular constraints. *Mathematics of Operations Research*, 23(4):806–831, 1998.

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.

Ting-Jui Chang and Shahin Shahrampour. On online optimization: Dynamic regret analysis of strongly convex and smooth problems. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(8):6966–6973, 2021.

Sapana Chaudhary and Dileep Kalathil. Safe online convex optimization with unknown linear safety constraints. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI)*, 36(6):6175–6182, 2022.

Tianyi Chen, Qing Ling, and Georgios B Giannakis. An online convex optimization approach to proactive network resource allocation. *IEEE Transactions on Signal Processing*, 65(24):6350–6364, 2017.

Rishabh Dixit, Amrit Singh Bedi, Ketan Rajawat, and Alec Koppel. Distributed online learning over time-varying graphs via proximal gradient descent. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 2745–2751. IEEE, 2019a.

Rishabh Dixit, Amrit Singh Bedi, Ruchi Tripathi, and Ketan Rajawat. Online learning with inexact proximal online gradient descent algorithms. *IEEE Transactions on Signal Processing*, 67(5):1338–1352, 2019b.

Roel Dobbe, Patricia Hidalgo-Gonzalez, Stavros Karagiannopoulos, Rodrigo Henriquez-Auba, Gabriela Hug, Duncan S Callaway, and Claire J Tomlin. Learning to control in power systems: Design and analysis guidelines for concrete safety problems. *Electric Power Systems Research*, 189:106615, 2020.

Nima Eshraghi and Ben Liang. Improving dynamic regret in distributed online mirror descent using primal and dual information. In *Learning for Dynamics and Control Conference*, pp. 637–649. PMLR, 2022.

Shuai Fan, Guangyu He, Xinyang Zhou, and Mingjian Cui. Online optimization for networked distributed energy resources with time-coupling constraints. *IEEE Transactions on Smart Grid*, 12(1):251–267, 2020.

Mohammad Fereydounian, Zebang Shen, Aryan Mokhtari, Amin Karbasi, and Hamed Hassani. Safe learning under uncertain objectives and constraints. *arXiv preprint arXiv:2006.13326*, 2020.

Udaya Ghai, Zhou Lu, and Elad Hazan. Non-convex online learning via algorithmic equivalence. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:22161–22172, 2022.

Eric Hall and Rebecca Willett. Dynamical models and tracking regret in online convex programming. In *International Conference on Machine Learning*, pp. 579–587. PMLR, 2013.

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

Elad Hazan and Satyen Kale. Projection-free online learning. In *International Coference on International Conference on Machine Learning (ICML)*, pp. 1843–1850, 2012.

Elad Hazan and Comandur Seshadhri. Efficient learning algorithms for changing environments. In *International Conference on Machine Learning (ICML)*, pp. 393–400, 2009.

Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2-3):169–192, 2007.

Elad Hazan, Karan Singh, and Cyril Zhang. Efficient regret minimization in non-convex games. In *International Conference on Machine Learning (ICML)*, pp. 1433–1441. PMLR, 2017.

Saghar Hosseini, Airlie Chapman, and Mehran Mesbahi. Online distributed convex optimization on dynamic networks. *IEEE Transactions on Automatic Control*, 61(11):3545–3550, 2016.

Ali Jadbabaie, Alexander Rakhlin, Shahin Shahrampour, and Karthik Sridharan. Online optimization: Competing with dynamic comparators. In *Artificial Intelligence and Statistics*, pp. 398–406, 2015.

Rodolphe Jenatton, Jim Huang, and Cédric Archambeau. Adaptive algorithms for online convex optimization with long-term constraints. In *International Conference on Machine Learning*, pp. 402–411. PMLR, 2016.

Kia Khezeli and Eilyan Bitar. Safe linear stochastic bandits. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(06):10202–10209, 2020.

Jueyou Li, Chuanye Gu, Zhiyou Wu, and Tingwen Huang. Online learning algorithm for distributed convex optimization with time-varying coupled constraints and bandit feedback. *IEEE transactions on cybernetics*, 2020.

Xiuxian Li, Lihua Xie, and Na Li. A survey of decentralized online learning. *arXiv preprint arXiv:2205.00473*, 2022.

Qiulin Lin, Hanling Yi, John Pang, Minghua Chen, Adam Wierman, Michael Honig, and Yuanzhang Xiao. Competitive online optimization under inventory constraints. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(1):1–28, 2019.

Jun S Liu. *Monte Carlo strategies in scientific computing*. Springer Science & Business Media, 2008.

Qingsong Liu, Wenfei Wu, Longbo Huang, and Zhixuan Fang. Simultaneously achieving sublinear regret and constraint violations for online convex optimization with time-varying constraints. *ACM SIGMETRICS Performance Evaluation Review*, 49(3):4–5, 2022.

Kaihong Lu and Long Wang. Online distributed optimization with nonconvex objective functions: Sublinearity of first-order optimality condition-based regret. *IEEE Transactions on Automatic Control*, 67(6):3029–3035, 2021.

Nguyen Cong Luong, Dinh Thai Hoang, Shimin Gong, Dusit Niyato, Ping Wang, Ying-Chang Liang, and Dong In Kim. Applications of deep reinforcement learning in communications and networking: A survey. *IEEE Communications Surveys & Tutorials*, 21(4):3133–3174, 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.

David Mateos-Núnez and Jorge Cortés. Distributed online convex optimization over jointly connected digraphs. *IEEE Transactions on Network Science and Engineering*, 1(1):23–37, 2014.

Aryan Mokhtari, Shahin Shahrampour, Ali Jadbabaie, and Alejandro Ribeiro. Online optimization in dynamic environments: Improved regret rates for strongly convex problems. In *IEEE 55th Conference on Decision and Control (CDC)*, pp. 7195–7201, 2016.

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

Alexander Rakhlin and Karthik Sridharan. Online learning with predictable sequences. In *Conference on Learning Theory*, pp. 993–1019. PMLR, 2013.

Robert J Ravier, A Robert Calderbank, and Vahid Tarokh. Prediction in online convex optimization for parametrizable objective functions. In *IEEE 58th Conference on Decision and Control (CDC)*, pp. 2455–2460, 2019.

Shahin Shahrampour and Ali Jadbabaie. An online optimization approach for multi-agent tracking of dynamic parameters in the presence of adversarial noise. In *American Control Conference (ACC)*, pp. 3306–3311, 2017.

Shahin Shahrampour and Ali Jadbabaie. Distributed online optimization in dynamic environments using mirror descent. *IEEE Transactions on Automatic Control*, 63(3):714–725, 2018.

Pranay Sharma, Prashant Khanduri, Lixin Shen, Donald J Bucci, and Pramod K Varshney. On distributed online convex optimization with sublinear dynamic regret and fit. In *2021 55th Asilomar Conference on Signals, Systems, and Computers*, pp. 1013–1017. IEEE, 2021.

Wei Shi, Qing Ling, Gang Wu, and Wotao Yin. Extra: An exact first-order algorithm for decentralized consensus optimization. *SIAM Journal on Optimization*, 25(2):944–966, 2015.

Wen Sun, Debadeepta Dey, and Ashish Kapoor. Safety-aware algorithms for adversarial contextual bandit. In *International Conference on Machine Learning*, pp. 3280–3288. PMLR, 2017.

Joel A Tropp et al. An introduction to matrix concentration inequalities. *Foundations and Trends® in Machine Learning*, 8(1-2):1–230, 2015.

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

Yuanyu Wan, Wei-Wei Tu, and Lijun Zhang. Projection-free distributed online convex optimization with $o(\sqrt{T})$ communication complexity. In *International Conference on Machine Learning*, pp. 9818–9828. PMLR, 2020.

Yuanyu Wan, Bo Xue, and Lijun Zhang. Projection-free online learning in dynamic environments. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 10067–10075, 2021.

Yuanyu Wan, Lijun Zhang, and Mingli Song. Improved dynamic regret for online frank-wolfe. *arXiv preprint arXiv:2302.05620*, 2023.

Feng Yan, Shreyas Sundaram, SVN Vishwanathan, and Yuan Qi. Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties. *IEEE Transactions on Knowledge and Data Engineering*, 25(11):2483–2493, 2012.

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, 2020a.

Xinlei Yi, Xiuxian Li, Tao Yang, Lihua Xie, Tianyou Chai, and Karl Henrik Johansson. Distributed bandit online convex optimization with time-varying coupled inequality constraints. *IEEE Transactions on Automatic Control*, 66(10):4620–4635, 2020b.

Xinlei Yi, Xiuxian Li, Tao Yang, Lihua Xie, Tianyou Chai, and Karl Johansson. Regret and cumulative constraint violation analysis for online convex optimization with long term constraints. In *International Conference on Machine Learning*, pp. 11998–12008. PMLR, 2021.

Xinlei Yi, Xiuxian Li, Tao Yang, Lihua Xie, Tianyou Chai, and H Karl. Regret and cumulative constraint violation analysis for distributed online constrained convex optimization. *IEEE Transactions on Automatic Control*, 2022.

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.

Hao Yu, Michael Neely, and Xiaohan Wei. Online convex optimization with stochastic constraints. *Advances in Neural Information Processing Systems*, 30, 2017.

Deming Yuan, Daniel WC Ho, and Guo-Ping Jiang. An adaptive primal-dual subgradient algorithm for online distributed constrained optimization. *IEEE transactions on cybernetics*, 48(11):3045–3055, 2017.

Deming Yuan, Alexandre Proutiere, and Guodong Shi. Distributed online linear regressions. *IEEE Transactions on Information Theory*, 67(1):616–639, 2020.

Deming Yuan, Alexandre Proutiere, and Guodong Shi. Distributed online optimization with long-term constraints. *IEEE Transactions on Automatic Control*, 67(3):1089–1104, 2021.

Jianjun Yuan and Andrew Lamperski. Online convex optimization for cumulative constraints. *Advances in Neural Information Processing Systems*, 31, 2018.

Lijun Zhang, Tianbao Yang, Jinfeng Yi, Rong Jin, and Zhi-Hua Zhou. Improved dynamic regret for non-degenerate functions. In *Advances in Neural Information Processing Systems*, pp. 732–741, 2017a.

Lijun Zhang, Shiyin Lu, and Zhi-Hua Zhou. Adaptive online learning in dynamic environments. In *Advances in neural information processing systems*, pp. 1323–1333, 2018a.

Lijun Zhang, Tianbao Yang, Zhi-Hua Zhou, et al. Dynamic regret of strongly adaptive methods. In *International conference on machine learning*, pp. 5882–5891. PMLR, 2018b.

Wenpeng Zhang, Peilin Zhao, Wenwu Zhu, Steven CH Hoi, and Tong Zhang. Projection-free distributed online learning in networks. In *International Conference on Machine Learning*, pp. 4054–4062. PMLR, 2017b.

Yan Zhang, Robert J Ravier, Michael M Zavlanos, and Vahid Tarokh. A distributed online convex optimization algorithm with improved dynamic regret. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 2449–2454. IEEE, 2019.

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