On Distributed Online Convex Optimization with Sublinear Dynamic Regret and Fit

Pranay Sharma[1], Prashant Khanduri[2], Lixin Shen[3], Donald J. Bucci Jr.[4] and Pramod K. Varshney[1]
[1]Department of Electrical Engineering and Computer Science, Syracuse University, Syracuse, NY
[2]Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN
[3]Department of Mathematics, Syracuse University, Syracuse, NY
[4]Lockheed Martin Advanced Technology Labs, Cherry Hill, NJ
Email: {psharm04, lshen03, varshney}@syr.edu, khand095@umn.edu, donald.j.bucci.jr@lmco.com

***Abstract*—In this work, we consider a distributed online convex optimization problem, with time-varying (potentially adversarial) constraints. A set of nodes, jointly aim to minimize a global objective function, which is the sum of local convex functions. The objective and constraint functions are revealed locally to the nodes, at each time, after taking an action. Naturally, the constraints cannot be instantaneously satisfied. Therefore, we reformulate the problem to satisfy these constraints in the long term. To this end, we propose a distributed primal-dual mirror descent-based algorithm, in which the primal and dual updates are carried out locally at all the nodes. This is followed by sharing and mixing of the primal variables by the local nodes via communication with the immediate neighbors. To quantify the performance of the proposed algorithm, we utilize the challenging, but more realistic metrics of dynamic regret and fit. Dynamic regret measures the cumulative loss incurred by the algorithm compared to the best dynamic strategy, while fit measures the long term cumulative constraint violations. Without assuming the restrictive Slater's conditions, we show that the proposed algorithm achieves sublinear regret and fit under mild, commonly used assumptions.**

I. Introduction

Many problems of practical interest, including network resource allocation [1], target tracking [2], network routing [3], and spam filtering [4] can be framed in an Online Convex Optimization (OCO) framework. First introduced in [3], the OCO framework aims to minimize a time varying convex objective function which is revealed to the observer in a sequential manner. In this work, we consider a constrained OCO problem, with time-varying (potentially adversarial) constraints.

Recently, distributed OCO frameworks have gained popularity as they distribute the computations across multiple nodes rather than having a central node perform all the operations [2], [5]–[7]. We consider the constrained OCO problem in a distributed framework, where the convex objective is assumed to be decomposed and distributed across multiple communicating agents. Each agent takes its own action with the goal of minimizing the dynamically varying global function while satisfying its individual constraints. Next, we discuss the related work along with the performance metrics we use to evaluate the performance of the proposed algorithm.

P. Sharma, and P. K. Varshney are partially supported by the National Science Foundation (NSF) under grant ENG-1609916. This work was done when P. Khanduri was at Syracuse University. The research of L. Shen is partially supported by NSF under grant DMS-1913039.

A. Related Work

Regret: The performance in OCO problems is quantified in terms of how well the agent does over time, compared to an offline system. In other words, how much the agent "regrets" not having the information, which was revealed to it post-hoc, to begin with. Since regret is cumulative over time, an algorithm that achieves sub-linear increase in regret with time, asymptotically achieves zero average loss. It is naturally desirable to compare against an offline system, the action(s) of which are "optimal" in some sense.

Static Regret: The initial work on OCO, starting with [3], almost exclusively focused on *static regret* $\mathbf{Reg}_T^s$, which uses an optimal *static* solution, as the benchmark. In other words, the fictitious offline adversary w.r.t. which the online system measures its performance, chooses the best *fixed* strategy, assuming it had access to all the information that is revealed to the online system over time horizon T.

$$\mathbf{Reg}_T^s \triangleq \sum\nolimits_{t=1}^T f_t(\mathbf{x}_t) - \min_{\mathbf{x}} \sum\nolimits_{t=1}^T f_t(\mathbf{x}).$$

Under standard regularity conditions, for general OCO problems, a tight upper bound of $O(\sqrt{T})$ has been shown for static regret [3], [8]. However, for applications such as online parameter estimation or tracking moving targets, where the quantity of interest also evolves over time, comparison with a static benchmark is not sufficient. This deficiency led to the development of *dynamic regret* $\mathbf{Reg}_T^d$ [9], [10].

Dynamic Regret: Rather than comparing the performance relative to a *fixed* optimal strategy, at each time instant, our fictitious adversary utilizes one-step look-ahead information to adopt the optimal strategy at the current time instant.

$$\mathbf{Reg}_T^d \triangleq \sum\nolimits_{t=1}^T f_t(\mathbf{x}_t) - \sum\nolimits_{t=1}^T \min_{\mathbf{x}} f_t(\mathbf{x}).$$

In this work, we adopt the notion of dynamic regret as the performance metric. It must, however, be noted that, in the worst case, it is impossible to achieve sublinear dynamic regret [3]. For such problems, the growth of dynamic regret is captured by the regularity measure which measures variations of the minimizer sequence over time (see C_T^* in Theorem V.6).

Constraints: The conventional approaches for OCO are based on projection-based gradient descent-like algorithms. However, when working with functional inequality constraints (say, $g_t(\mathbf{x}) \leq \mathbf{0}$), the projection step in itself can be computationally intensive. This led to the development of primal-dual algorithms [11]–[13]. Instead of attempting to satisfy the constraints at each time instant, the constraints are satisfied in the long run. In other words, the cumulative accumulation of instantaneous constraint violations (often simply called **fit**) $\|[\sum_{t=1}^T g_t(\mathbf{x}_t)]_+\|$ is shown to be sublinear in T. This formulation allows constraint violations at some instants to be taken-care-of by strictly feasible actions at other times.[1] Initially the constraints were assumed static across time [11], [12]. However, subsequent literature [1], [14] generalized the analysis to handle time-varying inequality constraints.

Distributed OCO Problems: So far we have only discussed centralized problems. Suppose the OCO system has a network of agents, and local cost (and constraint) functions are revealed to each agent over time. The global objective is to minimize the total cost function, while also satisfying all the constraints. And each agent can only communicate with agents that are in its immediate neighborhood. This distributed OCO problem is more challenging and much less studied in the literature than the centralized problem.

Distributed OCO problems with static constraints have been studied in recent years [2], [6], [7]. Again, the literature on distributed OCO with dynamic regret is much sparser than for static regret. The authors in [2] have proposed a dynamic mirror descent based algorithm, where primal update steps are alternated with local consensus steps. The authors in [6] have proposed a distributed primal-dual algorithm for the OCO problem with *coupled* inequality constraints. The constraint functions are static over time. This has been generalized for time-varying coupled constraints in [7], where the authors have shown sublinearity of regret and fit, both w.r.t. dynamic and static benchmarks. The authors in [5] have studied time-varying non-coupled constraints, in a continuous-time setting. However, the performance is measured using static regret. To the best of our knowledge, the distributed OCO problem with a *dynamic* benchmark, even with static non-coupled inequality constraints has so far not been considered in the literature.

B. Our Contributions

In this work, we consider a distributed online convex optimization problem, where both the cost functions and the time-varying inequality constraints are revealed locally to the individual nodes. We propose a primal-dual mirror-descent based algorithm, which alternates between the local primal and dual update steps and the consensus steps to mix the local primal variables with the immediate neighbors. Importantly, we show that the proposed algorithm achieves sublinear dynamic regret and fit.

Paper Organization: The problem formulation is discussed in Section II. Section III provides some background results and the assumptions required for providing theoretical guarantees. We propose our primal-dual mirror descent based algorithm in Section IV, followed by the theoretical results in Section V.

[1]Some more recent works [13] have considered the more stringent constraint violation metric $\sum_{t=1}^T([g_t(\mathbf{x}_t)]_+)^2$.

Notations: Vectors are denoted with lowercase bold letters, e.g., $\mathbf{x}$, while matrices are denoted using uppercase bold letters, e.g., $\mathbf{X}$. The set of positive integers is represented by $\mathbb{N}_+$. We use $\mathbb{R}_+^n$ to denote the n-dimensional non-negative orthant. For $n \in \mathbb{N}_+$, the set $\{1, \ldots, n\}$ is denoted by $[n]$. We denote by $\|\cdot\|$ the Euclidean norm for vectors, and the induced 2-norm for matrices. $\mathbf{0}$ denotes a zero vector, where the dimension is clear from the context. $[\mathbf{x}]_+$ denotes the projection onto $\mathbb{R}_+^n$.

II. Problem Formulation

We consider a network of n agents. At each time instant t, each agent i takes an action $\mathbf{x}_{i,t} \in \mathcal{X} \subseteq \mathbb{R}^d$, where the set $\mathcal{X}$ is fixed across time, across all the nodes. Then, a set of local loss functions $\{f_{i,t}(\cdot)\}_{i=1}^n$ with $f_{i,t} : \mathcal{X} \to \mathbb{R}$ are revealed to the individual nodes, resulting in individual loss $f_{i,t}(\mathbf{x}_{i,t})$ at node i. Additionally, another set of local functions $\{g_{i,t}(\cdot)\}_{i=1}^n$ with $g_{i,t} : \mathcal{X} \to \mathbb{R}^m$ are revealed, corresponding to local constraints $g_{i,t}(\mathbf{x}_{i,t}) \leq \mathbf{0}$. The network objective is to minimize the global average of the local cost functions $f_t(\mathbf{x}) \triangleq \frac{1}{n}\sum_{i=1}^n f_{i,t}(\mathbf{x})$, while also satisfying all the local constraints $\{g_{i,t}(\cdot)\}_{i=1}^n$.

$$\min_{\mathbf{x}_t \in \mathcal{X}} \; f_t(\mathbf{x}_t) \triangleq \sum\nolimits_{i=1}^n f_{i,t}(\mathbf{x}_t) \text{ s.t. } g_{i,t}(\mathbf{x}_t) \leq \mathbf{0}_m, \forall\, i \in [n]. \quad (1)$$

Since the objective is to minimize the global function $f_t(\cdot)$, the nodes need to communicate among themselves. We next define the metrics used to measure the performance of the proposed approach.

A. Performance Metrics - Dynamic Regret and Fit

We use the notion of dynamic regret [9], [10] to measure the performance relative to a time-varying benchmark.

$$\mathbf{Reg}_T^d \triangleq \frac{1}{n}\sum\nolimits_{i=1}^n \sum\nolimits_{t=1}^T f_t(\mathbf{x}_{i,t}) - \sum\nolimits_{t=1}^T f_t(\mathbf{x}_t^*), \quad (2)$$

where $\mathbf{x}_{i,t}$ is the local action of agent i at time t, while $\mathbf{x}_t^*$ is the solution of the following problem

$$\mathbf{x}_t^* \in \arg\min_{\mathbf{x} \in \mathcal{X}} \left\{ f_t(\mathbf{x}) \middle| g_{i,t}(\mathbf{x}) \leq \mathbf{0}, \forall\, i \in [n] \right\}. \quad (3)$$

As pointed out earlier, it is impossible to satisfy the time-varying constraints instantaneously, since they are revealed post-hoc. As a surrogate, to ensure the local constraints are satisfied in the *long run*, we use the distributed extension of *fit* as the performance metric. Fit has been used in the context of both time-invariant [11], as well as time-varying constraints [1], [15], for single node problems. Our definition is motivated by the one given in [5] for continuous time problems. It measures the average accumulation of constraint violations over time.

$$\mathbf{Fit}_T^d \triangleq \frac{1}{n}\sum\nolimits_{i=1}^n \frac{1}{n}\sum\nolimits_{j=1}^n \left\| \left[\sum\nolimits_{t=1}^T g_{i,t}(\mathbf{x}_{j,t}) \right]_+ \right\|. \quad (4)$$

Here, $\sum_{t=1}^T g_{i,t}(\mathbf{x}_{j,t})$ is the constraint violation at agent i, if it adopts the actions of agent j.

Next, we discuss the assumptions and some background required for the analysis of the proposed OCO framework. Note that the following assumptions are standard for decentralized OCO problems [2], [7].

III. Background and Assumptions

A. Network

We assume the n agents are connected together via an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. $\mathcal{V} = \{1, \ldots, n\}$ denotes the set of nodes of the graph, each of which represents an agent. $\mathcal{E}$ is the set of edges between the nodes. $(i, j) \in \mathcal{E}$ implies that nodes i and j are connected in the graph. The set of edges has an associated weight matrix $\mathbf{W}$, such that $[\mathbf{W}]_{ij} > 0$ if $(i, j) \in \mathcal{E}$, and $[\mathbf{W}]_{ij} = 0$ otherwise. The set of neighbors of node i is defined as $\mathcal{N}_i \triangleq \{j : [\mathbf{W}]_{ij} > 0\}$. Note that $j \in \mathcal{N}_i \Leftrightarrow i \in \mathcal{N}_j$.

Assumption A: The network is connected. The weight matrix $\mathbf{W}$ is symmetric and doubly stochastic.

B. Local Objective Functions and Constraints

Assumption B: We assume the following conditions on the set $\mathcal{X}$, the objective and constraint functions.
(B1) The set $\mathcal{X} \subseteq \mathbb{R}^d$ is convex and compact. Therefore, there exists a positive constant $d(\mathcal{X})$ such that

$$\|\mathbf{x} - \mathbf{y}\| \leq d(\mathcal{X}), \ \forall \ \mathbf{x}, \mathbf{y} \in \mathcal{X}. \tag{5}$$

(B2) The local node objective and constraint functions $f_{i,t}(\cdot), g_{i,t}(\cdot), \ \forall \ i \in [n], \forall \ t \in \mathbb{N}_+$ satisfy the following conditions:

1) $f_{i,t}(\cdot), g_{i,t}(\cdot)$ are L-Lipschitz continuous on $\mathcal{X}$
2) $f_{i,t}, g_{i,t}$ are convex and uniformly bounded on the set $\mathcal{X}$, i.e., there exists a constant $F > 0$ such that $\|f_{i,t}(\mathbf{x})\| \leq F, \|g_{i,t}(\mathbf{x})\| \leq F$.
3) The gradients $\nabla f_{i,t}, \nabla g_{i,t}$ exist and are uniformly bounded on $\mathcal{X}$, i.e., there exists a constant $G > 0$ such that $\|\nabla f_{i,t}(\mathbf{x})\| \leq G, \|\nabla g_{i,t}(\mathbf{x})\| \leq G$.

Next, we briefly discuss Bregman Divergence, which is crucial to the proposed mirror descent based approach.

C. Bregman Divergence

Suppose we are given a μ-strongly convex function $\mathcal{R} : \mathcal{X} \to \mathbb{R}$, i.e. $\mathcal{R}(\mathbf{x}) \geq \mathcal{R}(\mathbf{y}) + \langle \nabla \mathcal{R}(\mathbf{y}), \mathbf{x} - \mathbf{y} \rangle + \frac{\mu}{2}\|\mathbf{x} - \mathbf{y}\|^2, \forall \ \mathbf{x}, \mathbf{y} \in \mathcal{X}$. The Bregman Divergence w.r.t. $\mathcal{R}$ is defined as

$$\mathcal{D}_{\mathcal{R}}(\mathbf{x}, \mathbf{y}) \triangleq \mathcal{R}(\mathbf{x}) - \mathcal{R}(\mathbf{y}) - \langle \mathbf{x} - \mathbf{y}, \nabla \mathcal{R}(\mathbf{y}) \rangle . \tag{6}$$

We assume the following conditions on $\mathcal{D}_{\mathcal{R}}(\cdot, \cdot)$.
Assumption C: (C1) Separate Convexity property [16]: Given $\mathbf{x}, \{\mathbf{y}_i\}_{i=1}^m \in \mathbb{R}^d$ and scalars $\{\alpha_i\}_{i=1}^m$ on the m-dimensional probability simplex, the Bregman Divergence satisfies

$$\mathcal{D}_{\mathcal{R}}\left(\mathbf{x}, \sum\nolimits_{i=1}^m \alpha_i \mathbf{y}_i\right) \leq \sum\nolimits_{i=1}^m \alpha_i \mathcal{D}_{\mathcal{R}}\left(\mathbf{x}, \mathbf{y}_i\right). \tag{7}$$

(C2) Lipschitz continuity condition [17]: For any $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathcal{X}$

$$\left|\mathcal{D}_{\mathcal{R}}\left(\mathbf{x}, \mathbf{y}\right) - \mathcal{D}_{\mathcal{R}}\left(\mathbf{z}, \mathbf{y}\right)\right| \leq K \left\|\mathbf{x} - \mathbf{z}\right\|. \tag{8}$$

This condition is satisfied if $\mathcal{R}(\cdot)$ is Lipschitz continuous on $\mathcal{X}$. Consequently,

$$\mathcal{D}_{\mathcal{R}}\left(\mathbf{x}, \mathbf{y}\right) \leq K d((X)), \ \forall \ \mathbf{x}, \mathbf{y} \in \mathcal{X}, \tag{9}$$

where $d((X))$ is defined in (5).

Algorithm 1 Distributed Primal-Dual Mirror Descent

1: **Input**: Non-increasing sequences $\{\alpha_t > 0\}, \{\beta_t > 0\}, \{\gamma_t > 0\}$; Differentiable and strongly-convex $\mathcal{R}$
2: **Initialize**: $\mathbf{x}_{i,0} = \mathbf{0}_d \in \mathcal{X}$, $f_{i,0}(\cdot) \equiv 0, g_{i,0}(\cdot) \equiv \mathbf{0}_m$, $\mathbf{q}_{i,0} = \mathbf{0}_m, \forall \ i \in [n]$.
3: **for** $t = 1$ to T **do**
4:     **for** $i = 1$ to n **do**
5:         Observe $\nabla f_{i,t-1}(\mathbf{x}_{i,t-1}), \nabla g_{i,t-1}(\mathbf{x}_{i,t-1})$ $g_{i,t-1}(\mathbf{x}_{i,t-1})$
6:         $\mathbf{a}_{i,t} = \nabla f_{i,t-1}(\mathbf{x}_{i,t-1}) + [\nabla g_{i,t-1}(\mathbf{x}_{i,t-1})]^T \mathbf{q}_{i,t-1}$
7:         $\mathbf{y}_{i,t} = \arg\min_{\mathbf{x} \in \mathcal{X}} \{\alpha_t \langle \mathbf{x}, \mathbf{a}_{i,t} \rangle + \mathcal{D}_{\mathcal{R}}(\mathbf{x}, \mathbf{x}_{i,t-1})\}$
8:         $\mathbf{b}_{i,t} = [\nabla g_{i,t-1}(\mathbf{x}_{i,t-1})] (\mathbf{y}_{i,t} - \mathbf{x}_{i,t-1})$ $+ g_{i,t-1}(\mathbf{x}_{i,t-1})$
9:         $\mathbf{q}_{i,t} = [\mathbf{q}_{i,t-1} + \gamma_t(\mathbf{b}_{i,t} - \beta_t \mathbf{q}_{i,t-1})]_+$
10:         Broadcast $\mathbf{y}_{i,t}$ to out-neighbors $j \in \mathcal{N}_i$
11:         Obtain $\mathbf{y}_{j,t}$ from in-neighbors $j \in \mathcal{N}_i$
12:         $\mathbf{x}_{i,t} = \sum_{j=1}^n [\mathbf{W}]_{ij} \mathbf{y}_{j,t}$
13:     **end for**
14: **end for**

IV. Distributed Primal-Dual Mirror Descent based Algorithm

We next discuss the proposed distributed primal-dual mirror descent based algorithm for online convex optimization with time-varying constraints. The pseudo-code is outlined in Algorithm 1. The algorithm runs in parallel at all the nodes. At the end of time $t - 1$, $\mathbf{x}_{i,t-1}$ is the action (primal variable) at node i. Following this, the local functions $f_{i,t-1}, g_{i,t-1}$ are revealed to the agent. The corresponding function values and gradients are utilized to carry-out the updates in the next time step t. First, each agent performs the primal update locally (Step 7). This is followed by the dual update (Step 9). Note that the projection $[\cdot]_+$ ensures that the dual variable lies in the non-negative orthant $\mathbb{R}_+^m$. At the end of each time step, an average consensus step is taken across the nodes, where the local updated primal variables $\mathbf{y}_{i,t-1}$ are received from the neighbors, to compute the action $\mathbf{x}_{i,t}$.

Remark 1. Note that the primal and dual update steps employ different step-sizes, α_t and γ_t, respectively. This idea originated in [12] and leads to flexibility in terms of the trade-off between the bounds on dynamic regret and fit.

In the next section, we bound the dynamic regret and fit which result from Algorithm 1, and show them to be sublinear in the time-horizon T.

V. Dynamic Regret and Fit Bounds

First, we discuss some intermediate results required to show the sublinearity of dynamic regret and fit. We have omitted the proofs due to space limitations. Our analysis is inspired by the work in [2], [5], [7].

A. Intermediate Results

Lemma V.1. *Suppose Assumption B holds.* $\forall\, i \in [n]$, $\forall\, t \in \mathbb{N}_+$, $\mathbf{q}_{i,t}$ *generated by Algorithm 1 satisfy*

$$\|\mathbf{q}_{i,t}\| \leq \frac{F}{\beta_t} \tag{10}$$

$$\frac{\Delta_{t+1}}{2\gamma_{t+1}} \leq \frac{nB_1^2}{2}\gamma_{t+1} + \sum_{i=1}^{n} \mathbf{q}_{i,t}^T[\nabla g_{i,t}(\mathbf{x}_{i,t})](\mathbf{y}_{i,t+1} - \mathbf{x}_{i,t}) + \left(\frac{G^2\alpha_{t+1}}{\mu} + \frac{\beta_{t+1}}{2}\right)\sum_{i=1}^{n}\|\mathbf{q}_i\|^2 + \sum_{i=1}^{n}(\mathbf{q}_{i,t} - \mathbf{q}_i)^T g_{i,t}(\mathbf{x}_{i,t}) + \frac{\mu}{4\alpha_{t+1}}\sum_{i=1}^{n}\|\mathbf{y}_{i,t+1} - \mathbf{x}_{i,t}\|^2 \tag{11}$$

where $B_1 = 2F + Gd(\mathcal{X})$,

$$\Delta_{t+1} \triangleq \sum_{i=1}^{n}\left[\|\mathbf{q}_{i,t+1} - \mathbf{q}_i\|^2 - (1 - \gamma_{t+1}\beta_{t+1})\|\mathbf{q}_{i,t} - \mathbf{q}_i\|^2\right],$$

and $\{\mathbf{q}_i\}_i$ *are arbitrary vectors in* $\mathbb{R}_+^m$.

Remark 2. The penalty term $-\beta_t\mathbf{q}_{i,t-1}$ in the dual update (step 9, Algorithm 1) helps in upper bounding the local dual variables in (10). This idea was initially used in [11] and helps get rid of the requirement of Slater's condition. Δ_{t+1} measures the regularized *drift* of the local dual variables. See [9] and [7] for similar results, respectively in centralized and distributed (with coupled constraints) contexts.

Next, we sum (11) over t and define $g_c(\cdot)$ such that

$$g_c(\mathbf{q}_1, \ldots, \mathbf{q}_n) \triangleq \sum_{i=1}^{n}\mathbf{q}_i^T\left(\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{i,t})\right) - \left[\frac{1}{2\gamma_1} + \sum_{t=1}^{T}\left(\frac{G^2\alpha_{t+1}}{\mu} + \frac{\beta_{t+1}}{2}\right)\right]\sum_{i=1}^{n}\|\mathbf{q}_i\|^2$$
$$\leq \frac{nB_1^2}{2}\sum_{t=1}^{T}\gamma_{t+1} + \sum_{t=1}^{T}\sum_{i=1}^{n}\mathbf{q}_{i,t}^T[\nabla g_{i,t}(\mathbf{x}_{i,t})](\mathbf{y}_{i,t+1} - \mathbf{x}_{i,t}) + \sum_{t=1}^{T}\sum_{i=1}^{n}\mathbf{q}_{i,t}^T g_{i,t}(\mathbf{x}_{i,t}) + \sum_{t=1}^{T}\frac{\mu}{4\alpha_{t+1}}\sum_{i=1}^{n}\|\mathbf{y}_{i,t+1} - \mathbf{x}_{i,t}\|^2 - \frac{1}{2}\sum_{t=1}^{T}\left(\frac{1}{\gamma_t} - \frac{1}{\gamma_{t+1}} + \beta_{t+1}\right)\sum_{i=1}^{n}\|\mathbf{q}_{i,t} - \mathbf{q}_i\|^2. \tag{12}$$

Remark 3. The function $g_c(\mathbf{q}_1, \ldots, \mathbf{q}_n)$ will be used later in Lemma V.5 to upper bound both the dynamic regret and fit, by appropriately choosing $\{\mathbf{q}_i\}_i$.

Next, we bound the dynamic regret defined in (2).

Lemma V.2. *Suppose Assumption (B2) holds* $\forall\, i \in [n]$, $\forall\, t \in \mathbb{N}_+$. *For primal iterates* $\{\mathbf{x}_{i,t}\}$ *generated by Algorithm 1*

$$\mathbf{Reg}_T^d \leq \frac{1}{n}\sum_{i=1}^{n}\sum_{t=1}^{T}\{f_{i,t}(\mathbf{x}_{i,t}) - f_{i,t}(\mathbf{x}_t^*)\} + \frac{2L}{n}\sum_{i=1}^{n}\sum_{t=1}^{T}\|\mathbf{x}_{i,t} - \bar{\mathbf{x}}_t\|. \tag{13}$$

Remark 4. Compared to (2), the bound in (13) is decomposed in summation of local errors (first term), which can be bounded using the convexity of functions, and the consensus error.

First, we upper bound the first term in (13).

Lemma V.3. *Suppose Assumptions A-C hold.* $\forall\, i \in [n]$, $\forall\, t \in \mathbb{N}_+$, *if* $\{\mathbf{x}_{i,t}\}$ *is the sequence generated by Algorithm 1. Then,*

$$\sum_{t=1}^{T}\sum_{i=1}^{n}[f_{i,t}(\mathbf{x}_{i,t}) - f_{i,t}(\mathbf{x}_t^*)] \leq \frac{nG^2}{\mu}\sum_{t=1}^{T}\alpha_{t+1} - \sum_{t=1}^{T}\sum_{i=1}^{n}\frac{\mu}{4\alpha_{t+1}}\|\mathbf{y}_{i,t+1} - \mathbf{x}_{i,t}\|^2 - \sum_{t=1}^{T}\sum_{i=1}^{n}\mathbf{q}_{i,t}^T\Big[g_{i,t}(\mathbf{x}_{i,t}) + \nabla g_{i,t}(\mathbf{x}_{i,t})\,(\mathbf{y}_{i,t+1} - \mathbf{x}_{i,t})\Big] + \sum_{i=1}^{n}\left[\frac{1}{\alpha_2}\mathcal{D}_{\mathcal{R}}(\mathbf{x}_1^*, \mathbf{x}_{i,1}) - \frac{1}{\alpha_{T+2}}\mathcal{D}_{\mathcal{R}}(\mathbf{x}_{T+1}^*, \mathbf{x}_{i,T+1})\right] + \frac{nK}{\alpha_{T+2}}\sum_{t=1}^{T}\|\mathbf{x}_{t+1}^* - \mathbf{x}_t^*\| + \frac{nKd((X))}{\alpha_{T+2}}. \tag{14}$$

Next, we upper bound the second term in (13). This is the consensus error of the primal variables.

Lemma V.4. *(Network Error): Suppose Assumptions A-C hold. Then, the local estimates* $\{\mathbf{x}_{i,t}\}$ *generated by Algorithm 1 satisfy*

$$\|\mathbf{x}_{i,t} - \bar{\mathbf{x}}_t\| \leq \sum_{\tau=0}^{t-1}\sqrt{n}\sigma_2^{t-\tau}(\mathbf{W})\frac{G\alpha_{\tau+1}}{\mu}\left(1 + \frac{F}{\beta_{\tau+1}}\right) \tag{15}$$

$\forall\, i \in [n]$, *where* $\bar{\mathbf{x}}_t = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_{i,t}$. $\sigma_2(\mathbf{W})$ *is the second largest eigenvalue of* $\mathbf{W}$ *in magnitude.*

Remark 5. The network error bound is (15) is independent of the node index i. The dependence on $\sigma_2(\mathbf{W})$ captures the speed with which mixing of iterates happens. The smaller the value of $\sigma_2(\mathbf{W})$, the faster the network error diminishes. Moreover, the choice of the primal update step sizes $\{\alpha_t\}$ and the dual update regularization parameters $\{\beta_t\}$ has a crucial role to play in bounding the network error. As we shall see in Theorem V.6, carefully choosing these leads to sublinear regret and fit.

Next, we combine (12) and Lemma V.3 resulting in two intermediate bounds, which shall be needed to subsequently bound the dynamic regret and fit respectively.

Lemma V.5. *Suppose Assumptions A-C hold. Then, the sequences* $\{\mathbf{x}_{i,t}, \mathbf{q}_{i,t}\}$ *generated by Algorithm 1 satisfy*

$$\sum_{t=1}^{T}\sum_{i=1}^{n}(f_{i,t}(\mathbf{x}_{i,t}) - f_{i,t}(\mathbf{x}_t^*)) \leq \frac{nB_1^2}{2}\sum_{t=1}^{T}\gamma_{t+1} + \frac{nG^2}{\mu}\sum_{t=1}^{T}\alpha_{t+1} + \frac{nK}{\alpha_{T+2}}\sum_{t=1}^{T}\|\mathbf{x}_{t+1}^* - \mathbf{x}_t^*\| + \frac{nKd((X))}{\alpha_{T+2}} + \sum_{i=1}^{n}\left[\frac{1}{\alpha_2}\mathcal{D}_{\mathcal{R}}(\mathbf{x}_1^*, \mathbf{x}_{i,1}) - \frac{1}{\alpha_{T+2}}\mathcal{D}_{\mathcal{R}}(\mathbf{x}_{T+1}^*, \mathbf{x}_{i,T+1})\right] - \frac{1}{2}\sum_{t=1}^{T}\left(\frac{1}{\gamma_t} - \frac{1}{\gamma_{t+1}} + \beta_{t+1}\right)\sum_{i=1}^{n}\|\mathbf{q}_{i,t}\|^2, \tag{16}$$

and

$$\sum_{i=1}^{n}\left\|\left[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{i,t})\right]_+\right\|^2 \leq 4\left[\frac{1}{2\gamma_1} + \sum_{t=1}^{T}\left(\frac{G^2\alpha_{t+1}}{\mu} + \frac{\beta_{t+1}}{2}\right)\right]\Big\{2nFT + \frac{nB_1^2}{2}\sum_{t=1}^{T}\gamma_{t+1} + \frac{nG^2}{\mu}\sum_{t=1}^{T}\alpha_{t+1} + \sum_{i=1}^{n}\left[\frac{1}{\alpha_2}\mathcal{D}_{\mathcal{R}}(\mathbf{x}_1^*, \mathbf{x}_{i,1}) - \frac{1}{\alpha_{T+2}}\mathcal{D}_{\mathcal{R}}(\mathbf{x}_{T+1}^*, \mathbf{x}_{i,T+1})\right] + \frac{nK}{\alpha_{T+2}}\sum_{t=1}^{T}\|\mathbf{x}_{t+1}^* - \mathbf{x}_t^*\| + \frac{nKd((X))}{\alpha_{T+2}} - \frac{1}{2}\sum_{t=1}^{T}\left(\frac{1}{\gamma_t} - \frac{1}{\gamma_{t+1}} + \beta_{t+1}\right)\sum_{i=1}^{n}\|\mathbf{q}_{i,t} - \bar{\mathbf{q}}_i\|^2\Big\}. \tag{17}$$

Remark 6. (16) follows by adding (12) and (14), and substituting $\mathbf{q}_i = \mathbf{0}_m$, $\forall\, i \in [n]$. The first, second and fourth terms in (16) depend only on the step-size sequences $\{\alpha_t, \beta_t, \gamma_t\}_t$. The fifth term is a telescopic sum. The third term depends on $\sum_{t=1}^{T} \|\mathbf{x}_{t+1}^* - \mathbf{x}_t^*\|$, the accumulated dynamic variation of the comparator sequence $\{\mathbf{x}_t^*\}$. This quantity is a property of the system and not in control of the agents. Therefore, by carefully choosing the step-sizes, and assuming well-behaved variation of the comparator sequence, we can bound the dynamic regret (see Theorem V.6).

Remark 7. Similarly, (17) is obtained by adding (12) and (14), and substituting

$$\bar{\mathbf{q}}_i = \frac{\left[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{i,t})\right]_+}{\gamma_1^{-1} + \sum_{t=1}^{T} (2G^2\alpha_{t+1}/\mu + \beta_{t+1})}, \ \forall\ i \in [n]. \quad (18)$$

As in (16), the upper bound depends on the step-size sequences $\{\alpha_t, \beta_t, \gamma_t\}_t$, and $\sum_{t=1}^{T} \|\mathbf{x}_{t+1}^* - \mathbf{x}_t^*\|$.

Before presenting out final result, we need to use the following upper bound to bound the fit (4).

$$\begin{aligned} &\frac{1}{n}\sum_{i=1}^{n} \frac{1}{n}\sum_{j=1}^{n} \left\| \left[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{j,t})\right]_+ \right\|^2 \quad (19)\\ &\leq 2\left[2L\sum_{t=1}^{T} \|\mathbf{x}_{i,t} - \bar{\mathbf{x}}_t\|\right]^2 + \frac{2}{n}\sum_{i=1}^{n} \left\| \left[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{i,t})\right]_+ \right\|^2. \end{aligned}$$

This follows from Lipschitz continuity of the constraint functions (Assumption (B2)). Since, we have bounded both the terms in (19) (the first term in Lemma V.4, and the second term in Lemma V.5), we are now ready to present our final result on the sublinearity of both dynamic regret and fit.

B. Dynamic Regret and Fit Bounds

Theorem V.6. *Suppose Assumptions A-C hold, and $\{\mathbf{x}_{i,t}\}$ be the sequence of local estimates generated by Algorithm 1. We choose the step sizes*

$$\alpha_t = t^{-a},\ \beta_t = t^{-b},\ \gamma_t = t^{b-1}, \quad \forall\ t \in \mathbb{N}_+ \quad (20)$$

where, $a, b \in (0,1)$ and $a > b$. Then for any $T \in \mathbb{N}_+$.

$$\mathbf{Reg}_T^d \leq R_1 T^{\max\{a, 1-a+b\}} + 2KT^a C_T^*, \quad (21)$$

$$\begin{aligned} &\frac{1}{n}\sum_{i=1}^{n} \frac{1}{n}\sum_{j=1}^{n} \left\| \left[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{j,t})\right]_+ \right\|^2 \\ &\qquad \leq D_1 T^{2-b} + D_2 T^{1+a-b} C_T^* + D_3 T^{2+2b-2a}. \quad (22) \end{aligned}$$

Here, $R = \frac{4FLG\sqrt{n}\sigma_2(\mathbf{W})}{\mu(1-a)(1-\sigma_2(\mathbf{W}))}, R_1 = R + \frac{B_1^2}{2b} + \frac{G^2}{\mu(1-a)} + 2Kd((X)), D = 2 + \frac{4G^2}{\mu(1-a)} + \frac{2}{1-b}, D_1 = 2D(2F + 2Kd((X)) + \frac{B_1^2}{2b} + \frac{G^2}{\mu(1-a)} + 2Kd((X)))$, $D_2 = 4KD$ and $D_3 = 16L^2R^2$ are constants independent of T, and

$$C_T^* \triangleq \sum_{t=1}^{T} \|\mathbf{x}_{t+1}^* - \mathbf{x}_t^*\| \quad (23)$$

is the accumulated dynamic variation of the comparator sequence $\{\mathbf{x}_t^*\}$.

Remark 8. The dynamic regret $\mathbf{Reg}_T^d$ is sublinear as long as the cumulative consecutive variations of the dynamic comparators C_T^* is sublinear ($o(T^{1-a})$ for $a > 0$). This is the standard requirement for sublinearity of dynamic regret [2], [7], [9].

Remark 9. A similar argument as above holds for (22). As long as C_T^* is sublinear, we have

$$\frac{1}{n}\sum_{i=1}^{n} \frac{1}{n}\sum_{j=1}^{n} \left\| \left[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{j,t})\right]_+ \right\|^2 = o(T^2). \quad (24)$$

Note that (22) has $\|[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{j,t})]_+\|^2$, while fit (4) is defined with $\|[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{j,t})]_+\|$. However, for large enough T, each of the constituent terms in (24) are $o(T^2)$. Consequently, $\|[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{j,t})]_+\|^2 = o(T^2)$, $\forall\ i, j \in [n]$. Therefore, we get

$$\mathbf{Fit}_T^d = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{n}\sum_{j=1}^{n} \left\| \left[\sum_{t=1}^{T} g_{i,t}(\mathbf{x}_{j,t})\right]_+ \right\| = o(T). \quad (25)$$

VI. Conclusion

In this work, we considered a distributed OCO problem, with time-varying (potentially adversarial) constraints. We proposed a distributed primal-dual mirror descent based approach, in which the primal and dual updates are carried out locally at all the nodes. We utilized the challenging, but more realistic metric of dynamic regret and fit. Without assuming the more restrictive Slater's conditions, we achieved sublinear regret and fit under mild, commonly used assumptions. To the best of our knowledge, this is the first work to consider distributed OCO problem with non-coupled local time-varying constraints, and achieve sublinear dynamic regret and fit.

References

[1] T. Chen, Q. Ling, and G. B. Giannakis, "An online convex optimization approach to proactive network resource allocation," *IEEE Transactions on Signal Processing*, vol. 65, no. 24, pp. 6350–6364, 2017.

[2] S. Shahrampour and A. Jadbabaie, "Distributed online optimization in dynamic environments using mirror descent," *IEEE Transactions on Automatic Control*, vol. 63, no. 3, pp. 714–725, 2017.

[3] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, 2003, pp. 928–936.

[4] E. Hazan *et al.*, "Introduction to online convex optimization," *Foundations and Trends® in Optimization*, vol. 2, no. 3-4, pp. 157–325, 2016.

[5] S. Paternain, S. Lee, M. M. Zavlanos, and A. Ribeiro, "Distributed constrained online learning," *IEEE Transactions on Signal Processing*, vol. 68, pp. 3486–3499, 2020.

[6] X. Li, X. Yi, and L. Xie, "Distributed online optimization for multi-agent networks with coupled inequality constraints," *IEEE Transactions on Automatic Control*, 2020.

[7] X. Yi, X. Li, L. Xie, and K. H. Johansson, "Distributed online convex optimization with time-varying coupled inequality constraints," *IEEE Transactions on Signal Processing*, vol. 68, pp. 731–746, 2020.

[8] E. Hazan, A. Agarwal, and S. Kale, "Logarithmic regret algorithms for online convex optimization," *Machine Learning*, vol. 69, no. 2-3, pp. 169–192, 2007.

[9] E. C. Hall and R. M. Willett, "Online convex optimization in dynamic environments," *IEEE Journal of Selected Topics in Signal Processing*, vol. 9, no. 4, pp. 647–662, 2015.

[10] O. Besbes, Y. Gur, and A. Zeevi, "Non-stationary stochastic optimization," *Operations research*, vol. 63, no. 5, pp. 1227–1244, 2015.

[11] M. Mahdavi, R. Jin, and T. Yang, "Trading regret for efficiency: online convex optimization with long term constraints," *Journal of Machine Learning Research*, vol. 13, no. Sep, pp. 2503–2528, 2012.

[12] R. Jenatton, J. Huang, and C. Archambeau, "Adaptive algorithms for online convex optimization with long-term constraints," in *International Conference on Machine Learning*, 2016, pp. 402–411.

[13] J. Yuan and A. Lamperski, "Online convex optimization for cumulative constraints," in *Advances in Neural Information Processing Systems*, 2018, pp. 6137–6146.

[14] W. Sun, D. Dey, and A. Kapoor, "Safety-aware algorithms for adversarial contextual bandit," in *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org, 2017, pp. 3280–3288.

[15] S. Paternain and A. Ribeiro, "Online learning of feasible strategies in unknown environments," *IEEE Transactions on Automatic Control*, vol. 62, no. 6, pp. 2807–2822, 2016.

[16] H. H. Bauschke and J. M. Borwein, "Joint and separate convexity of the bregman distance," in *Studies in Computational Mathematics*. Elsevier, 2001, vol. 8, pp. 23–36.

[17] A. Jadbabaie, A. Rakhlin, S. Shahrampour, and K. Sridharan, "Online optimization: Competing with dynamic comparators," in *Artificial Intelligence and Statistics*, 2015, pp. 398–406.