On the Sublinear Regret of Distributed Primal-Dual Algorithms for Online Constrained Optimization

Soomin Lee, and Michael M. Zavlanos,

Abstract—This paper introduces consensus-based primal-dual methods for distributed online optimization where the time-varying system objective function \( f_i(x) \) is given as the sum of local agents’ objective functions, i.e., \( f_i(x) = \sum_i f_i(x_i) \), and the system constraint function \( g(x) \) is given as the sum of local agents’ constraint functions, i.e., \( g(x) = \sum_i g_i(x_i) \leq 0 \). At each stage, each agent commits to an adaptive decision pertaining only to the past and locally available information, and incurs a new cost function reflecting the change in the environment. Our algorithm uses weighted averaging of the iterates for each agent to keep local estimates of the global constraints and dual variables. We show that the algorithm achieves a regret of order \( O(\sqrt{T}) \) with the time horizon \( T \), in scenarios when the underlying communication topology is time-varying and jointly-connected. The regret is measured in regard to the cost function value as well as the constraint violation. Numerical results for online routing in wireless multi-hop networks with uncertain channel rates are provided to illustrate the performance of the proposed algorithm.

I. INTRODUCTION

MANY engineering applications concerning the coordination of multi-agent systems can be cast as distributed optimization problems over networks, see e.g., [1]–[4]. In these applications, distributed agents, which only have access to local parameters, often try to minimize a global cost function subject to global constraints. The main feature of most protocols designed to carry this coordination over a network is that the agents only use information received from their immediate (one-hop) neighbors. The underlying communication structure of the agents can be cast as a graph, often directed and time-varying.

A common feature in many practical applications is that they live in dynamically changing and uncertain environments. One of the existing methods which can be used to address the uncertainties arising in these problems is online optimization, where the cost function changes over time and an adaptive decision pertaining only to the past information has to be made at each stage. The objective in online optimization is to reduce regret, a quantity capturing the difference between the accumulated cost incurred up to some arbitrary time and the cost obtained from the best fixed point chosen in hindsight. Online optimization has been studied extensively both in the optimization and machine learning community, see e.g., [5]–[7]. Most of the online optimization algorithms are built on gradient descent methods which exploit the convexity of a cost function.

In this paper, we focus on online distributed optimization problems over large networks using a consensus framework, where at each stage, each agent faces a new cost function reflecting the change in its nearby environment and can share its information only with neighbors. There are two distinctive ways that consensus can be used to obtain distributed optimization methods: (A) a decentralized weighted averaging is interleaved between optimization steps as a means of information mixing and network-wide agreement, and (B) the optimization problem is recast in an equivalent decomposable form by adding consistency constraints on any coupled quantities along all edges.

The consensus framework (A) was used in [8], [9], where a variant of the dual-averaging method by Nesterov [10] for online distributed optimization was proposed for undirected networks [8] and for time-invariant digraphs allowing time-varying weights [9]. Other recent work includes [11], [12] that employ the push-sum protocol [13], [14], which allows for time-varying weight-imbalanced digraphs. Specifically, the work in [11] and [12] is a variant of the distributed subgradient methods [15] and the dual-averaging method [10], respectively, for online distributed optimization. Recently, online distributed ADMM has also been proposed in [16], by incorporating either decentralized dual-averaging or subgradient descent in the ADMM process for time-invariant graph. On the other hand, the consensus framework (B) was used in [17], [18], where distributed online subgradient descent methods were proposed for time-varying weight-balanced digraphs that employed proportional-integral feedback for handling the consistency constraints among neighboring agents.

In this paper, we propose an online distributed variant of the saddle point algorithm by Arrow-Hurwicz-Uzawa [19], also known as Primal-Dual method, which is based on the consensus framework (A). An important contribution of this work is that it can properly handle (time-invariant) global constraints of the form \( \sum_{i=1}^{N} g_i(x_i) \leq 0 \), where the function \( g_i \) is only known to agent \( i \). A lot of practical applications in control [20]–[22], network flow control [23], wireless communications [24], machine learning [25], [26], and image processing [27] involve such globally coupled constraints. The main advantage of the Primal-Dual method is that it allows to decompose the global constraint into a sum of local terms in the dual domain. This results in regret of order \( O(\sqrt{T}) \) over any sequence of jointly connected time-varying digraphs, where \( T \) is the time horizon. The dependence of the regret on the number of agents and on the structural parameters of the problem and the network is also analyzed.

To the best of our knowledge, this is the first work in online distributed optimization showing a sublinearly bounded
regret with globally coupled constraints. All of the work mentioned above only considers a global cost function $f_i(x) = \sum_{i=1}^{N} f_{i,t}(x)$ and projection of the iterates onto a common constraint set $X$. Note that the global constraints $\sum_{i=1}^{N} g_i(x_i) \leq 0$ cannot be represented as $X = \bigcap_{i=1}^{N} X_i$, $X = \Pi_{i=1}^{N} X_i$, or $X = X_i$, where $X_i$ is a local constraint set of agent $i$. Therefore, distributed online optimization problems involving such global constraints cannot be properly handled by the existing methods.

Our method is most closely related to the recent work in [28] where a saddle point algorithm is considered for online distributed optimization. The difference is that [28] is built on the consensus framework (B) on a fixed undirected graph without any global constraints while our work is built on the consensus framework (A) on a sequence of time-varying digraphs with global constraints. Our algorithm uses weighted averaging of the iterates for each agent to keep local estimates of any global information, which allows us to obtain stronger and more general results on the regret according to the cost function value as well as the constraint violation. Our regret analysis also characterizes the dependence on the network parameters, such as the number of nodes and the network connectivity.

On a broader scale, the work in this paper is also related to the distributed primal-dual methods for time-invariant optimization, e.g., the penalty based distributed algorithm by [29] and the distributed primal-dual perturbed algorithm by [30].

This paper is organized as follows. In Section II we introduce notational conventions that are used throughout the paper and briefly review the Lagrangian duality, centralized primal-dual method and saddle point theorem. In Section III we formulate the online distributed constrained optimization problem under consideration and discuss an online routing problem in wireless multi-hop networks with uncertain channel rates as an application of the proposed algorithm. In Section IV we provide our proposed consensus-based online primal-dual algorithm. In Section V we provide assumptions and establish $O(\sqrt{T})$ bounds on the regret in the cost function as well as constraint violation incurred by the algorithm. In Section VI we present simulation results on the online routing problem in Section III. Lastly, in Section VII we summarize our contribution and conclude the paper.

II. PRELIMINARIES

In this section, we introduce basic Lagrangian duality and the related saddle point algorithm for a time-invariant centralized optimization problem that will be useful for the subsequent development. We start by introducing some notational conventions that we use throughout the paper.

A. Notations and Terminologies

Let $\mathbb{R}$ and $\mathbb{R}_+$ denote the set of real and nonnegative real numbers, respectively. All vectors are viewed as column vectors. We denote $\mathbf{0}$ by the vector of all zeros whose dimension varies accordingly. For a vector $a \in \mathbb{R}^n$, we use $[a]_+$ to denote the coordinate-wise projection of $a$ onto $\mathbb{R}_+^n$. We use $[x]_i$ to denote the $i$-th component of a vector $x$. We use $P_X[x]$ to denote the Euclidean projection of a vector $x$ on the set $X$, i.e., $P_X[x] = \arg \min_{y \in X} \|y - x\|^2$. For a matrix $A \in \mathbb{R}^{n \times m}$, we use $[A]_{ij}$ to denote the entry of $i$-th row and $j$-th column. We use $A'$ to denote the transpose of a matrix $A$. The inner product of two vectors $a$ and $b$ is $a'b$. Unless otherwise stated, $\| \cdot \|$ represents the standard Euclidean norm. For any $N \geq 1$, the set of integers $\{1, \ldots, N\}$ is denoted by $[N]$. The symbols $\leq$ and $\geq$ are understood as component-wise inequalities. We use $E[Z]$ to denote the expectation of a random variable $Z$.

A set $X \subseteq \mathbb{R}^n$ is convex if for all $\alpha \in [0,1]$ and $x, y \in X$

$$ \alpha x + (1 - \alpha) y \in X. $$

A function $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is convex in $X$ if $f$ is convex and for all $\alpha \in [0,1]$ and $x, y \in X$

$$ f(\alpha x + (1 - \alpha) y) \leq \alpha f(x) + (1 - \alpha) f(y), $$

and $f$ is strictly convex if the above inequality is strictly satisfied for all $\alpha \in (0,1)$, $x, y \in X$, and $x \neq y$. Equivalently, $f$ is convex in $X$ if the set $X$ is convex and for all $x, y \in X$

$$ f(x) \geq f(y) + \langle \nabla f(y), x - y \rangle. $$

We say a function $g : X \rightarrow \mathbb{R}$ is (strictly) concave over $X$ if $-g$ is (strictly) convex over $X$.

Consider a convex-concave function $L$ defined over $X \times \Lambda \subseteq \mathbb{R}^n \times \mathbb{R}^m$, i.e., $L(x, \lambda) : X \rightarrow \mathbb{R}$ is convex for every $\lambda \in \Lambda$ and $L(x, \cdot) : \Lambda \rightarrow \mathbb{R}$ is concave for every $x \in X$. A vector pair $(x^*, \lambda^*)$ is called a saddle point of the function $L$ over $X \times \Lambda$ if for any $x \in X$ and $\lambda \in \Lambda$

$$ L(x, \lambda) \leq L(x^*, \lambda^*) \leq L(x^*, \lambda). $$

B. Lagrangian Duality and Saddle Point Algorithm

Consider the following constrained optimization problem:

$$ \min_{x \in X} f(x) \quad \text{s.t.} \quad g(x) \leq 0, \quad (1) $$

where $f : \mathbb{R}^n \rightarrow \mathbb{R}$ is a convex function, $X \subseteq \mathbb{R}^n$ is a nonempty closed convex set, and $g : \mathbb{R}^n \rightarrow \mathbb{R}^m$ is a component-wise convex function; specifically, $g = (g_1, \ldots, g_m)$ with each $g_i : \mathbb{R}^n \rightarrow \mathbb{R}$ being convex. We call this problem a primal problem. Let $f^*$ denote its optimal value.

Consider the following Lagrange dual problem of (1):

$$ \max_{\lambda \in \mathbb{R}^m_+} q(\lambda), \quad (2) $$

where $q : \mathbb{R}^m_+ \rightarrow \mathbb{R}$ is the dual function given by

$$ q(\lambda) = \min_{x \in X} L(x, \lambda) \quad (3) $$

and $L : X \times \mathbb{R}^m_+ \rightarrow \mathbb{R}$ is the Lagrangian function given by

$$ L(x, \lambda) = f(x) + \lambda' g(x). \quad (4) $$

Let $q^*$ denote the optimal value of (2). It is well known that the weak duality $q^* \leq f^*$ always holds, and under the Slater’s condition, i.e., there exist a vector $\bar{x} \in \mathbb{R}^n$ such that $g_i(\bar{x}) < 0$ for all $i = 1, \ldots, m$, the strong duality $q^* = f^*$ also holds [31, 32].
One of the classical dual methods for solving problem (4) is the saddle point algorithm of Arrow-Hurwicz-Uzawa [19], also known as the Primal-Dual method. Given an initial point \((x_1, \lambda_1) \in X \times \mathbb{R}^m_+\), the method iteratively updates the primal-dual variables as follows: For \(t \geq 1\),
\[
\begin{align*}
\mathbf{x}_{t+1} &= P_X[\mathbf{x}_t - \alpha_t \nabla_x \mathcal{L}(\mathbf{x}_t, \mathbf{\lambda}_t)] \\
\mathbf{\lambda}_{t+1} &= \mathbf{\lambda}_t + \alpha_t \nabla_\mathbf{\lambda} \mathcal{L}(\mathbf{x}_t, \mathbf{\lambda}_t) + \mathbb{1}_+, 
\end{align*}
\]
where \(\{\alpha_t\}\) is a sequence of positive step sizes, \(\nabla_x \mathcal{L}(\mathbf{x}_t, \mathbf{\lambda}_t)\) and \(\nabla_\mathbf{\lambda} \mathcal{L}(\mathbf{x}_t, \mathbf{\lambda}_t)\) are the gradients of the Lagrangian \(\mathcal{L}\) at \((\mathbf{x}_t, \mathbf{\lambda}_t)\) with respect to \(\mathbf{x}\) and \(\mathbf{\lambda}\), respectively. More specifically, in step (5a), the algorithm estimates the primal variables \(\mathbf{x}\) that attain the current function value \(q(\mathbf{\lambda}_t)\) by approximately minimizing \(\mathcal{L}(\mathbf{x}, \mathbf{\lambda}_t)\) using gradient descent. In step (5b), the algorithm iteratively maximizes the dual function using gradient ascent. Here the assumption is that the ascent direction \(\nabla_{\mathbf{\lambda}} q(\mathbf{\lambda}_t)\) is approximately \(\nabla_{\mathbf{\lambda}} \mathcal{L}(\mathbf{x}_t, \mathbf{\lambda}_t)\).

It has been extensively studied that the sequence \((\mathbf{x}_t, \mathbf{\lambda}_t)\) generated by the method (5a)-(5b) converges to a saddle point of the Lagrangian function \(\mathcal{L}\) under the bounded gradient assumption, i.e., under the assumption that there exists a constant \(L\) such that for all \(t \geq 1\):
\[
\|\nabla_x \mathcal{L}(\mathbf{x}_t, \mathbf{\lambda}_t)\| \leq L, \quad \|\nabla_{\mathbf{\lambda}} \mathcal{L}(\mathbf{x}_t, \mathbf{\lambda}_t)\| \leq L, 
\]
(see e.g. [19] for more details). This and the following saddle point theorem say that the Primal-Dual method in (5a)-(5b) solves the original problem (1).

**Theorem 1:** [31, Proposition 5.1.6] (Saddle Point Theorem)
The point \((\mathbf{x}^*, \mathbf{\lambda}^*) \in X \times \mathbb{R}^m_+\) is primal-dual optimal pair of problems (1) and (2) if and only if it is a saddle point of the Lagrangian (4), i.e., for all \(\mathbf{x} \in X\) and \(\mathbf{\lambda} \in \mathbb{R}^m_+\),
\[
\mathcal{L}(\mathbf{x}^*, \mathbf{\lambda}) \leq \mathcal{L}(\mathbf{x}^*, \mathbf{\lambda}^*) \leq \mathcal{L}(\mathbf{x}, \mathbf{\lambda}^*). 
\]

### III. PROBLEM FORMULATION AND APPLICATION

Consider a networked system consisting of \(N\) agents indexed by the set \(\mathcal{V} = [N]\). The communication among the network agents is governed by a sequence of time-varying weighted digraphs \(\mathcal{G}_t = (\mathcal{V}, \mathcal{E}_t, W_t)\), for \(t \geq 0\). There exists a directed link \((i, j) \in \mathcal{E}_t \subseteq \mathcal{V} \times \mathcal{V}\) if and only if agent \(i\) can receive a message from agent \(j\) at time \(t\). Here \(W_t \in \mathbb{R}^{N \times N}\) is a weight matrix respecting the topology of the graph \(\mathcal{G}_t\), i.e., \(W_{i,j}^t > 0\) if and only if \((i, j) \in \mathcal{E}_t\) and \(W_{i,j}^t = 0\), otherwise. Each entry \(W_{i,j}^t\) represents a weight that agent \(i\) allocates to the incoming link \((i, j)\). We denote the set of incoming neighbors of agent \(i\) at time \(t\) by
\[
\mathcal{N}_{i,t} = \{j \mid (i, j) \in \mathcal{E}_t\} \cup \{i\},
\]
where we always include agent \(i\) itself to its own set of neighbors.

We have the following assumption on the communication model.

**Assumption 1:** For all \(t \geq 1\), the weighted graphs \(\mathcal{G}_t = (\mathcal{V}, \mathcal{E}_t, W_t)\) satisfy:
(a) There exists a scalar \(\eta \in (0, 1)\) such that \(W_{i,j}^t \geq \eta\) if \(j \in \mathcal{N}_{i,t}\). Otherwise, \(W_{i,j}^t = 0\).
(b) The weight matrix \(W_t\) is doubly stochastic, i.e., \(\sum_{j=1}^N W_{i,j}^t = 1\) for all \(i \in \mathcal{V}\) and \(\sum_{i=1}^N W_{i,j}^t = 1\) for all \(j \in \mathcal{V}\).
(c) There exists a scalar \(Q > 0\) such that the graph \((\mathcal{V}, \cup_{t=0,\ldots,Q-1} \mathcal{E}_{t+1})\) is strongly connected for any \(t \geq 1\).

Assumption 1(a) ensures that the weight matrices \(W_t\) respect the underlying topology \(\mathcal{G}_t\) for every \(t\) so that the communication is distributed. The \(\eta\)-nondegeneracy states that all positive weights are bounded away from zero by \(\eta\). This ensures that certain entries do not disappear too quickly, but the network agents need not know the \(\eta\) value in running the algorithm. Assumption 1(b) is required for balanced communication as removing the doubly stochasticity might introduce unwanted bias in the optimization [33, 34]. Assumption 1(c) ensures that there exists a path from one agent to every other agent within any bounded interval of length \(Q\). We also say that such a sequence of graphs is \(Q\)-strongly connected.

### A. Online Distributed Constrained Optimization

In online optimization, we model uncertainties as a sequence of time-varying objective functions, which are not known in advance. More specifically, at each time \(t\), each agent \(i \in \mathcal{V}\) must select a state \(\mathbf{x}_{i,t}\) from \(X_i\), which is a closed and bounded subset of \(\mathbb{R}^n\), without the knowledge of the current cost \(f_{i,t}\). After this, agent \(i\) observes \(f_{i,t}\), which is chosen by nature and comes from a fixed class of convex functions \(f_i : X_i \rightarrow \mathbb{R}\), and incurs the cost of \(f_{i,t}(\mathbf{x}_{i,t})\). The network-wide cost that agents want to minimize at time \(t\) is then
\[
\sum_{i=1}^N f_{i,t}(\mathbf{x}_{i,t}), 
\]
where the vector \(\mathbf{x} \in X\) is distributed among the agents, i.e., \(\mathbf{x} = (x_1, \ldots, x_N) \in X\) with \(X = X_1 \times \cdots \times X_N \subseteq \mathbb{R}^n\) and \(n = \sum_{i=1}^N n_i\). We emphasize here that \(f_i\) is neither collectively known to any of the agents nor available at any single location. In addition to this, there exist some global (time-invariant) constraint functions
\[
g(\mathbf{x}) = \sum_{i=1}^N g_i(\mathbf{x}_i) \leq 0, 
\]
where each \(g_i = (g_{i1}, \ldots, g_{im})\) with \(g_{ij} : \mathbb{R}^{n_i} \rightarrow \mathbb{R}\) is only available to agent \(i\). We assume throughout the paper that the feasible set
\[
\mathcal{X} = \{ \mathbf{x} \in X \mid g(\mathbf{x}) \leq 0 \}
\]
is nonempty.

Since the cost functions \(\{f_{i,t}\}_{t \geq 1}\) are only revealed to agent \(i\) and each constraint function \(g_i\) is only known to agent \(i\), each agent must determine its state \(\mathbf{x}_{i,t}\) based on what it “thinks” the network cost function \(g\) is subject to the global constraint [7]. We will postpone the detailed strategy based on which each agent \(i\) chooses such \(\mathbf{x}_{i,t}\) to the next section, except to note that \(\mathbf{x}_{i,t}\) depends only on the local cost functions revealed before time \(t\), i.e., \(\{f_{i,s}\}_{s=1}^{t-1}\), the local constraints \(g_i\), and the coordination of the information received from the neighbors.

Due to the time-varying nature of online optimization, we need a different metric to define its convergence. The objective
in online optimization is to reduce regret, a quantity analogous to the distance from the optimal function value in time-invariant optimization. More specifically, the regret at arbitrary $T \geq 1$ is defined as follows:

$$\mathcal{R}(T) = \sum_{t=1}^{T} \sum_{i=1}^{N} f_{i,t}(x_{i,t}) - \sum_{t=1}^{T} \sum_{i=1}^{N} f_{i,t}(x^*_i),$$  \hspace{1cm} (8)

which computes the difference between the summation of the local cost incurred at each agent’s state $x_{i,t}$ and the best fixed point chosen from an offline and centralized view. Here $x^*$ is defined as

$$x^* \triangleq \inf_{x \in X} \sum_{t=1}^{T} \sum_{i=1}^{N} f_{i,t}(x(t)),$$

i.e., $x^*$ is a single state that could have been achieved with perfect advance knowledge of all cost functions $\{f_{i,t}\}_{i \in V, 1 \leq t \leq T}$ and without any restriction on the communication between any agents.

Essentially, the regret captures the accumulation of error due to the fact that decisions are made without knowledge of the current objective functions. This error will will keep accumulating as the time horizon $T$ grows, however, it is expected to accumulate at a rate that is sublinear, i.e., $\mathcal{R}(T) = o(T)$. If this is the case, then the average accumulated error, i.e., the average regret $\mathcal{R}(T)/T$, will converge to zero as the time horizon $T$ grows large. This means that eventually, the decision makers will be able to learn the objective function and make decisions that are more accurate.

Since this is an online constrained problem, we also need a notion of regret associated with the constraints which is analogous to the regret in (8) associated with the cost function. Specifically, for a penalty function $F: \mathbb{R}^m \rightarrow \mathbb{R}^m_+$, we consider the following cumulative constraint violation for some arbitrary $T \geq 1$:

$$\mathcal{R}^c(T) = \left\| \sum_{t=1}^{T} F \left( \sum_{i=1}^{N} g_i(x_{i,t}) \right) \right\|,$$

which can be used to show that the constraints in (7) are satisfied in the long run. As with the regret in (8), in this case too we need to show that for some arbitrary $T \geq 1$, the cumulative constraint violation is sublinear in $T$, i.e.,

$$\mathcal{R}^c(T) = o(T).$$

Thus, the goal of this paper is to design a distributed update rule for each agent $i \in \mathcal{V}$ such that (a) the regret (8) and constraint violation (10) are sublinear over the time horizon $T$ (such that the average $\mathcal{R}(T)/T$ and $\mathcal{R}^c(T)/T$ converge to zero), and (b) show reasonable dependence on the total number of network agents and on the topology of the communication graphs.

B. Application to optimal wireless networking under channel uncertainty

Consider a wireless network consisting of $N$ source nodes indexed by $i \in \mathcal{V} = \{1, \ldots, N\}$ that route information to $K$ Access Points (APs) indexed by $i \in \mathcal{K} = \{N + 1, \ldots, N + K\}$. We assume that APs only receive and do not transmit any data. Point-to-point connectivity is modeled through a rate function $R_{ij}$ that determines the amount of information that is transmitted from node $i \in \mathcal{V}$ and is correctly decoded by node $j \in \mathcal{V} \cup \mathcal{K}$. We assume that direct communication with the APs is not always possible, so the source nodes need to route data to the APs in a multi-hop fashion. Routing of packets is due to routing decisions $T_{ij}$ that represent the fraction of time that node $i$ selects node $j$ as its intended destination. Note that, since the routing variables $T_{ij}$ represent time slot shares they need to satisfy $0 \leq T_{ij} \leq 1$ and $\sum_{j \in \mathcal{V} \cup \mathcal{K}} T_{ij} = 1$. The products $R_{ij}T_{ij}$ denote the point-to-point rate of information that is transmitted by node $i$ and correctly received by node $j$; see Figure 1.

![Figure 1](image)

Between their generation or arrival from another node and their transmission, packets are stored in a queue. The total rate at which packets arrive at the queue of node $i$ is $\sum_{j \in \mathcal{V}} T_{ji}R_{ji}$. Similarly, the total rate at which packets leave the queue of node $i$ is $\sum_{j \in \mathcal{V} \cup \mathcal{K}} T_{ij}R_{ij}$. Collecting all routing decisions in a vector $T \in [0, 1]^{(N + K)}$, we can express the end-to-end information rate $r_i(T)$ at node $i$ as

$$r_i(T) = \sum_{j \in \mathcal{V} \cup \mathcal{K}} T_{ij}R_{ij} - \sum_{j \in \mathcal{V}} T_{ji}R_{ji}.$$  \hspace{1cm} (11)

This is the rate at which node $i$ can add data in the network. Integrity of the communication network requires that the end-to-end rates $r_i(T)$ exceed minimum thresholds $r_{i,\text{min}}$ that capture the rate of data that is directly generated at nodes, i.e., $r_i(T) \geq r_{i,\text{min}}$ for all $i \in \mathcal{V}$.

In practice, the rates $R_{ij}$ depend on the signal-to-noise ratio [35] and, thus, they are subject to uncertainty. This is because received signal strength in wireless communications is typically affected by path loss due to the distance between the transmitter and the receiver, shadowing due to the presence of obstacles in the environment, and multipath fading due to reflections and refractions of the electromagnetic waves. While path loss and shadowing effects can be captured using predictive models, multipath fading is difficult to predict [36]. As a result, we let the rates $R_{ij}$ be random variables, and consider a stochastic equivalent of (11) as the following:

$$r_i(T) = \sum_{j \in \mathcal{V} \cup \mathcal{K}} T_{ij}E[R_{ij}] - \sum_{j \in \mathcal{V}} T_{ji}E[R_{ji}].$$  \hspace{1cm} (12)
Note that the rates $R_{ij}$ are independent of each other, but they may not be identically distributed.

Therefore, any points satisfying the following inequalities are feasible operating points:

$$
    r_{i, \min} \leq \sum_{j \in \mathcal{V} \cup \mathcal{K}} T_{ij} E[R_{ij}] - \sum_{j \in \mathcal{V}} T_{ji} E[R_{ji}], \quad \forall i \in \mathcal{V} \tag{13a}
$$

$$
    \sum_{j \in \mathcal{V} \cup \mathcal{K}} T_{ij} = 1, \quad \forall i \in \mathcal{V}, \tag{13b}
$$

$$
    0 \leq T_{ij} \leq 1, \quad \forall i \in \mathcal{V}, \quad j \in \mathcal{V} \cup \mathcal{K}. \tag{13c}
$$

Centralized solutions to (13a)-(13c) can incur a large communication cost to collect information about the network topology (contained in the rates $E[R_{ij}]$) at a central location and to communicate the routing variables $T_{ij}$ back to the nodes. They also entail significant delays and are vulnerable to failures. For this reason, we consider distributed algorithms, whereby each node $i$ has access only to variables available to its neighbors.

Denoting by $x_i = \left[ T_{i1}, T_{i2}, \ldots, T_{i(N+K)} \right]^T \in \mathbb{R}^{N+K}$ the vector of decision variables of node $i$, and by $E[R_i] \in \mathbb{R}^{N \times (N+K)}$ the matrix

$$
    E[R_i] = \begin{bmatrix}
    -E[R_{i1}] & 0 & \cdots & 0 \\
    0 & -E[R_{i2}] & \cdots & 0 \\
    \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & \cdots & -E[R_{iN}] \\
    \end{bmatrix},
$$

then, the flow constraints (13a) can be compactly written as

$$
    \sum_{i \in \mathcal{V}} E[R_i] x_i \geq r_{\min}, \tag{14}
$$

where $r_{\min} = [r_{1,\min}, \ldots, r_{J,\min}]$. By introducing auxiliary variables $z_i \in \mathbb{R}^N$ for $i \in \mathcal{V}$, the constraint (14) can be equivalently represented as

$$
    E[R_i] x_i = z_i, \quad \forall i \in \mathcal{V} \tag{15a}
$$

$$
    z_i \geq r_{\min}. \tag{15b}
$$

Since the expectations $E[R_i]$ are not known in advance, we reformulate (15a)-(15b) as an online optimization problem. A time-varying, varying agent $i$ which measures the violation of the constraints in (15a) is defined as:

$$
    f_{i,t}(x_i, z_i) = \frac{1}{2} \| z_i - R_{i,t} x_i \|^2, \tag{16}
$$

where $R_{i,t}$ is the empirical estimation of $E[R_i]$ at time $t$, i.e., $R_{i,t} = \frac{1}{t} \sum_{k=1}^{t} R_{i,k}$, and $R_{i,k}$ is a random realization of $R_i$ at time $k$. Furthermore, the global constraints can be written as:

$$
    \sum_{i=1}^{N} g_i(x_i, z_i) = \sum_{i=1}^{N} z_i + r_{\min} \preceq 0. \tag{17}
$$

### IV. On-line Distributed Algorithms for Globally Constrained Optimization

We now introduce a distributed algorithm for online constrained optimization which combines the saddle point algorithm of Arrow-Hurwicz-Uzawa given in (5a)-(5b) with a consensus framework for coordination. Specifically, given a convex function $F : \mathbb{R}^m \to \mathbb{R}^m$ which satisfies the property $F(x) \preceq 0$ whenever $x \preceq 0$, the Lagrangian (or penalty) function $\mathcal{H}_t : \mathbb{R}^n \times \mathbb{R}^m \to \mathbb{R}$ of the distributed problem (6)-(7) at time $t$ can be written as:

$$
    \mathcal{H}_t(x, \lambda) = \sum_{i=1}^{N} f_i(t)(x_i) + \frac{1}{N} \lambda^T F \left( \sum_{i=1}^{N} g_i(x_i) \right). \tag{18}
$$

Then, the gradients of the Lagrangian at $(x, \lambda)$ with respect to $x_i$ and $\lambda$ are respectively given as:

$$
    \nabla_x \mathcal{H}_t(x, \lambda) = \nabla f_i(t)(x_i) + \frac{1}{N} \nabla g_i(x_i)^T \nabla F \left( \sum_{i=1}^{N} g_i(x_i) \right) \lambda \tag{19a}
$$

$$
    \nabla_{\lambda} \mathcal{H}_t(x, \lambda) = \frac{1}{N} F \left( \sum_{i=1}^{N} g_i(x_i) \right), \tag{19b}
$$

where $\nabla f_i(t)(x_i) \in \mathbb{R}^n$ is the gradient of $f_i(t)$ evaluated at $x_i$, $\nabla g_i(x_i)$ and $\nabla F \left( \sum_{i=1}^{N} g_i(x_i) \right)$ are $m \times n_i$ and $m \times m$ Jacobian matrices evaluated at $x_i$ and $\sum_{i=1}^{N} g_i(x_i)$, respectively. Since the calculation of (19a)-(19b) requires the global knowledge $\lambda$ and $\sum_{i=1}^{N} g_i(x_i)$, (19a) cannot be directly used by agent $i$. Therefore, we let each agent $i$ maintain local sequences $\{\lambda_{i,t}\}_{t \geq 1} \subseteq \mathbb{R}^m_+$ and $\{y_{i,t}\}_{t \geq 1} \subseteq \mathbb{R}^m$, which respectively estimate the true network-wide dual variables and the average of the global function value, i.e., $\frac{1}{N} \sum_{i=1}^{N} g_i(x_i)$.

The proposed distributed online primal-dual algorithm is summarized in Algorithm 1. In this algorithm, each agent $i \in \mathcal{V}$ maintains three local sequences $\{x_{i,t}\}_{t \geq 1} \subseteq X_i$, $\{\lambda_{i,t}\}_{t \geq 1} \subseteq \mathbb{R}^m_+$, and $\{y_{i,t}\}_{t \geq 1} \subseteq \mathbb{R}^m$, which are initialized by an arbitrary $x_{i,0} \in X_i$ and $\lambda_{i,0} \in \mathbb{R}^m_+$ while $y_{i,0}$ is initialized as $g_i(x_{i,0})$. Then, the three iterates $x_{i,t}$, $\lambda_{i,t}$ and $y_{i,t}$ are updated recursively using the update rules (21a)-(21e), where $\{\alpha_k\}_{k \geq 1}$ is a decreasing sequence of positive step sizes, and the vector $s_{i,t} \in \mathbb{R}^m$ is defined as:

$$
    s_{i,t} = \nabla f_i(t)(x_{i,t}) + \frac{1}{N} \nabla g_i(x_{i,t})^T \nabla F \left( N \tilde{y}_{i,t} \right) \lambda_{i,t}. \tag{20}
$$

In the following lemma, we show that the initial condition $y_{i,0} = g_i(x_{i,0})$ and the update rule (21b) and (21e) allow the sum of the iterates $y_{i,t}$ for $i \in \mathcal{V}$ to maintain the true global function value $\sum_{i=1}^{N} g_i(x_{i,t})$ at all time $t \geq 0$. (See Appendix F for the proof.)

**Lemma 1:** Let us define $\tilde{y}_t$ to be the average of $y_{i,t}$ for $i \in \mathcal{V}$, i.e., $\tilde{y}_t \triangleq \frac{1}{N} \sum_{i=1}^{N} y_{i,t}$. Let Assumption 1 hold. Then, for any $t \geq 0$, the following equalities hold:

$$
    \sum_{i=1}^{N} y_{i,t} = \sum_{i=1}^{N} \tilde{y}_{i,t} = \sum_{i=1}^{N} g_i(x_{i,t}). \tag{21b}
$$

Therefore, either $y_{i,t}$ or $\tilde{y}_t$ is a local estimate on the average $\frac{1}{N} \sum_{i=1}^{N} g_i(x_{i,t})$ that is available at agent $i$. Also, in view of this lemma and equation (19)-(20), it is obvious that the steps in (21a)-(21d) become an approximation of the centralized online primal-dual algorithm update (5a)-(5b). This will become more apparent when we bound the network wide disagreement $\| \lambda_{i,t} - \lambda_i \|$ and $\| \tilde{y}_{i,t} - \tilde{y}_t \|$ later in Lemma 4.
Algorithm 1 Distributed Online Primal-Dual

Require: Set $T \geq 1$ and $t = 1$. Initialize locally $x_{i,0} \in X_i$, $\lambda_{i,0} \in \mathbb{R}_+$ and $y_{i,0} = g_i(x_{i,0})$ for every $i \in \mathcal{V}$.

1: If $t = T$, then stop. Otherwise, set for every $i \in \mathcal{V}$:

\begin{align}
\hat{\lambda}_{i,t} &= \sum_{j=1}^{N} [W_{i}]_{ij} \lambda_{j,t} \\
\hat{y}_{i,t} &= \sum_{j=1}^{N} [W_{i}]_{ij} y_{j,t} \\
x_{i,t+1} &= P_{X_i}[x_{i,t} - \alpha_t s_{i,t}] \\
\lambda_{i,t+1} &= \left[\hat{\lambda}_{i,t} + \frac{\alpha_t}{N} F(N\hat{y}_{i,t})\right]_+ \\
y_{i,t+1} &= \hat{y}_{i,t} + g_i(x_{i,t+1}) - g_i(x_{i,t}),
\end{align}

2: Increase $t$ by one and return to Step 1.

where $\hat{\lambda}_i$ and $\hat{y}_i$ denote the average of $\lambda_{i,t}$ and $y_{i,t}$ for $i \in \mathcal{V}$, respectively, i.e., $\hat{\lambda}_i = \frac{1}{N} \sum_{t=1}^{T} \lambda_{i,t}$ and $\hat{y}_i = \frac{1}{N} \sum_{t=1}^{T} y_{i,t}$.

Note that the function $F: \mathbb{R}^m \to \mathbb{R}^m$ can be any arbitrary function as long as it satisfies Assumption 3 in Section V.

Examples include i) $F(x) = x$, for which the dual update rule in (21d) of this algorithm becomes exactly same as that of Arrow-Hurwicz-Uzawa [19], or ii) a smooth surrogate of the max function $|x|_+$, for which the algorithm becomes a variant of the penalty methods (see e.g., [31]).

Remark To the best of our knowledge, there has not been any work considering the primal-dual algorithm even for centralized online problems. Although the focus of this paper is on online optimization problems defined over networks, it is worth mentioning that the online primal-dual algorithm in (21a–21e) works for centralized cases as well. Indeed, a centralized online optimization can be seen as a special case with $N = 1$.

V. Regret Analysis

We now provide a regret analysis of the proposed distributed online primal-dual method. In Section V-A, we present additional assumptions on the problem described in Section III. In Sections V-C and V-B, we present two key lemmas that will be used in the proof of the main results. The main results are provided in Section V-D.

A. Assumptions

Our regret analysis is based on the following assumptions. Assumptions 2 and 3 are typical in any gradient-based algorithms. They ensure the boundedness of the gradients so that the gradient based method in Algorithm 1 is well behaved.

Assumption 2: (a) The functions $\{f_{i,t}\}_{i \in \mathcal{V}, t \geq 1}$ and $\{g_i\}_{i \in \mathcal{V}}$ are convex and continuously differentiable.

1Every result in this paper holds even if $f_{i,t}$ and $g_i$ are nondifferentiable; we mainly stick to this case for simplicity.

(b) The sets $X_i$ for $i \in \mathcal{V}$ are compact. That is, there exists a constant $C_X$ such that

$$\|x\| \leq C_X \text{ for all } x \in X_i \text{ and } i \in \mathcal{V}.$$  

(c) The iterates $\{\lambda_{i,t}\}_{i \in \mathcal{V}, t \geq 1}$ are bounded, i.e., there exists a constant $C_\lambda$ such that for all $t \geq 1$ and $i \in \mathcal{V}$

$$\|\lambda_{i,t}\| \leq C_\lambda.$$  

Direct consequences of Assumption 2(a) and 2(b) are gradient boundedness and Lipschitz continuity of the functions $\{f_{i,t}\}_{i \in \mathcal{V}, t \geq 1}$, i.e., there exists a constant $L_f > 0$ such that for all $i \in \mathcal{V}$

$$\|\nabla f_{i,t}(x)\| \leq L_f, \text{ for all } x \in X_i, \tag{22}$$

$$|f_{i,t}(x) - f_{i,t}(y)| \leq L_f \|x - y\|, \text{ for all } x, y \in X_i. \tag{23}$$

Similarly, for the constraint functions $\{g_i\}_{i \in \mathcal{V}}$, there exists a constant $L_g > 0$ such that for all $i \in \mathcal{V}$

$$\|\nabla g_i(x)\| \leq L_g, \text{ for all } x \in X_i, \tag{24}$$

$$|g_i(x) - g_i(y)| \leq L_g \|x - y\|, \text{ for all } x, y \in X_i. \tag{25}$$

Furthermore, the compactness of $X_i$’s and the continuity of the functions $\{f_{i,t}\}_{i \in \mathcal{V}, t \geq 1}$ and $\{g_i\}_{i \in \mathcal{V}}$ imply that there exist constants $C_f, C_g > 0$ such that for all $i \in \mathcal{V}$

$$|f_{i,t}(x)| \leq C_f, \text{ for all } x \in X_i, \tag{26}$$

$$|g_i(x)| \leq C_g, \text{ for all } x \in X_i. \tag{27}$$

Note that Assumptions 2(a)-(c) ensure the boundedness of the gradient of the Lagrangian (see Eq. (19a)-(19b) and also (20)). Such an assumption is typical in any gradient based method, which is also seen in the Arrow-Hurwicz-Uzawa’s original paper [19].

Assumption 3: The function $F$ is defined, convex and continuously differentiable over $\mathbb{R}^m$. We further assume $F(x) \leq 0$ if $x \preceq 0$. Also, it has bounded and Lipschitz continuous gradients, i.e., there exists some constants $G_f, G_g > 0$ and $L_F > 0$ such that

$$\|\nabla F(x) - \nabla F(y)\| \leq G_F \|x - y\|, \text{ for all } x, y \in \mathbb{R}^m,$n

$$\|\nabla F(x)\| \leq L_F, \text{ for all } x \in \mathbb{R}^m. \tag{28}$$

B. Sublinear Network Disagreement

We will employ a result from [37]. This result shows that a set of sequences recursively updated by weighted averaging followed by perturbation will not drift too much from its average sequence as long as the underlying weight matrices satisfy the properties listed in Assumption 2 and the perturbation behaves nicely.
Lemma 2: Let Assumptions 1 and 2 hold for a sequence of weight matrices \( \{ W_i \}_{i \geq 1} \). Consider a set of sequences \( \{ \theta_{i,t} \} \) for \( i \in [N] \) defined by the following relation:

\[
\theta_{i,t+1} = \sum_{j=1}^{N} [W_i]_{ij} \theta_{j,t} + \epsilon_{i,t+1}, \quad \text{for} \ t \geq 1.
\]  

(29)

Let \( \bar{\theta}_i \) denote the average of \( \theta_{i,t} \) for \( i \in [N] \), i.e., \( \bar{\theta}_i = \frac{1}{N} \sum_{t=1}^{T} \theta_{i,t} \). Then,

\[
\| \theta_{i,t+1} - \bar{\theta}_i \| \leq N \gamma \beta^t \max_j \| \theta_{j,1} \| + \gamma \sum_{t=1}^{T-1} \beta^{t-t} \max_j \| \theta_{j,1} \|
\]

\[
+ \gamma \sum_{t=1}^{T-1} \beta^{t-t} \max_j \| \epsilon_{j,t+1} \| + \frac{1}{N} \sum_{j=1}^{N} \| \epsilon_{j,t+1} \| + \| \epsilon_{i,t+1} \|,
\]

where \( \gamma > 0 \) and \( \beta \) are defined as

\[
\gamma = \left( 1 - \eta \frac{N}{2N^2} \right) \quad \beta = \left( 1 - \eta \frac{N}{2N^2} \right)^{\frac{1}{2}}.
\]

(31)

Note that the iterates \( \lambda_{i,t} \) and \( y_{i,t} \) in algorithm (21d)-(21e) can be represented by relation (29). Indeed, (21d) can be rewritten as

\[
\lambda_{i,t+1} = \sum_{j=1}^{N} [W_i]_{ij} \lambda_{j,t} + \epsilon_{i,t+1},
\]

with

\[
\epsilon_{i,t+1} = \left[ \bar{\lambda}_{i,t} + \frac{\alpha_t}{N} \mathcal{F}(N \bar{y}_{i,t}) \right]_+ - \bar{\lambda}_{i,t}.
\]

(32)

Similarly, (21e) can be rewritten as

\[
y_{i,t+1} = \sum_{j=1}^{N} [W_i]_{ij} y_{j,t} + \epsilon_{i,t+1},
\]

with

\[
\epsilon_{i,t+1} = g_i(x_{i,t+1}) - g_i(x_{i,t}).
\]

(33)

Using equation (33) and Lemma 2, we first show the boundedness of the iterates \( y_{i,t} \) (see Appendix A for the proof).

Lemma 3: Let Assumptions 1 and 2 hold. Then, the iterates \( \{ y_{i,t} \}_{i \in \mathcal{V}, t \geq 1} \) are bounded, i.e., there exists a constant \( C_y \) such that for all \( i \geq 1 \) and \( t \in [T] \),

\[
\| y_{i,t} \| \leq C_y,
\]

where

\[
C_y = \max \left\{ \| y_{i,1} - \bar{y}_1 \|, N \gamma \max_j \| y_{j,1} \| \right\}
\]

\[
+ \frac{2\gamma \beta}{1 - \beta} L_g C_x + 4L_g C_x \right\} + C_y.
\]

(34)

From step (21b), Assumption 3 and Lemma 2, we know that there exists a constant \( C_x \) such that for all \( i \in \mathcal{V} \) and \( t \geq 1 \),

\[
\mathcal{F}(N \bar{y}_{i,t}) \leq C_x.
\]

The following corollary shows that if the perturbation can be bounded and controlled by a decreasing step size, the discrepancy of each sequence from the average sequence summed over an arbitrary time horizon can be sublinearly bounded (see Appendix B for the proof).

Corollary 1: Suppose that for the sequences \( \{ \theta_{i,t} \} \) defined in Lemma 2, there exist a positive scalar sequence \( \{ \alpha_t \}_{t \geq 1} \) and a constant \( K > 0 \) such that \( \| \epsilon_{i,t} \| \leq K \alpha_t \) for all \( i \in [N] \) and \( t \geq 1 \). Then, for any \( T \geq 2 \),

\[
\sum_{t=1}^{T-1} \| \theta_{i,t+1} - \bar{\theta}_{i,t+1} \| \leq \gamma N \beta \max_j \| \theta_{j,1} \|
\]

\[
+ \frac{\gamma N K \beta}{1 - \beta} + 2K \sum_{t=1}^{T-1} \alpha_t,
\]

where \( \gamma > 0 \) and \( \beta \) are defined as

\[
\gamma = \left( 1 - \eta \frac{N}{2N^2} \right) \quad \beta = \left( 1 - \eta \frac{N}{2N^2} \right)^{\frac{1}{2}}.
\]

(31)

We now make use of Corollary 2 in the following lemma to bound the disagreement between each agent \( i \)'s iterates \( \{ \lambda_{i,t} \}_{t \geq 1} \) and \( \{ y_{i,t} \}_{t \geq 1} \) and their corresponding average sequences (see Appendix A for the proof).

Lemma 4: Let Assumptions 1 and 2 hold. Consider sequences \( \{ \lambda_{i,t} \}_{i \in \mathcal{V}, t \geq 1} \) and \( \{ y_{i,t} \}_{i \in \mathcal{V}, t \geq 1} \) generated by the algorithm in (21a)-(21e). Let \( \bar{\lambda}_t \) and \( \bar{y}_t \) denote the average of \( \lambda_{i,t} \) and \( y_{i,t} \) for \( i \in \mathcal{V} \), i.e., \( \bar{\lambda}_t = \frac{1}{N} \sum_{i=1}^{N} \lambda_{i,t} \) and \( \bar{y}_t = \frac{1}{N} \sum_{i=1}^{N} y_{i,t} \). Then, with a stepsize choice of

\[
\alpha_t \geq \frac{1}{\sqrt{t}}.
\]

the network disagreement terms can be upper bounded as follows: For all \( i \in \mathcal{V} \) and \( T \geq 1 \), we have

\[
\sum_{t=1}^{T-1} \sum_{i=1}^{N} \| \lambda_{i,t} - \lambda_i \| \leq B_1(T) + B_2(T) \sqrt{T},
\]

\[
\sum_{t=1}^{T-1} \sum_{i=1}^{N} \| y_{i,t} - y_i \| \leq B_3(T) + B_4(T) \sqrt{T},
\]

where the constants \( B_1(T), B_2(T), B_3(N) \) and \( B_4(N) \) depend on the number of nodes \( N \) as well as the minimum weight \( \eta \) in the averaging matrix and the connectedness of the time-varying graphs \( Q \).

C. Basic Iterate Relations

In this section, we establish some basic relations that hold for the sequences \( \{ x_{i,t} \}, \{ \lambda_{i,t} \} \) and \( \{ y_{i,t} \} \) obtained by the distributed online primal-dual algorithm in Algorithm 1 (see Appendix A for these proof). These relations will play an important role in our analysis of the sublinearly bounded regret in the following Section V-D.

Lemma 5: Let Assumptions 1 and 3 hold. Consider sequences \( \{ x_{i,t} \}_{i \in \mathcal{V}, t \geq 1} \), \( \{ \lambda_{i,t} \}_{i \in \mathcal{V}, t \geq 1} \) and \( \{ y_{i,t} \}_{i \in \mathcal{V}, t \geq 1} \) generated by the distributed online primal-dual algorithm in (21a)-(21e). Let \( \lambda_t = \frac{1}{N} \sum_{i=1}^{N} \lambda_{i,t} \) and \( \lambda_t = \frac{1}{N} \sum_{i=1}^{N} \lambda_{i,t} \). Then, we have:

(a) For any \( x \in \mathcal{X} \) and \( t \geq 1 \),

\[
\mathcal{H}(x, \lambda_t) - \mathcal{H}(x, \lambda_t)
\]
\( \frac{1}{2\alpha_t} \left( \sum_{i=1}^{N} \| x_{i,t} - x_i \|^2 - \sum_{i=1}^{N} \| x_{i,t+1} - x_i \|^2 \right) \)
\[ + \frac{\alpha_t}{2} N \left( L_f + \frac{1}{N} L_g L_F C_\lambda \right)^2 \]
\[ + 2 C_\lambda L_g L_F \sum_{i=1}^{N} \| \tilde{y}_{i,t} - \tilde{y}_i \| \]
\[ + \frac{2}{N} C_\lambda L_g L_F \sum_{i=1}^{N} \| \tilde{\lambda}_{i,t} - \tilde{\lambda}_i \|. \]

(b) For any \( \lambda \in \mathbb{R}_+^m \) and \( t \geq 1 \),
\[ H_t(x_t, \lambda) - H_t(x, \tilde{\lambda}) \]
\[ \leq \frac{1}{2\alpha_t} \left( \sum_{i=1}^{N} \| \lambda_{i,t} - \lambda \|^2 - \sum_{i=1}^{N} \| \lambda_{i,t+1} - \lambda \|^2 \right) + \frac{\alpha_t}{2} N C_F^2 \]
\[ + \frac{C_F}{N} \sum_{i=1}^{N} \| \tilde{\lambda}_{i,t} - \tilde{\lambda}_i \| + 2 C_\lambda L_g \sum_{i=1}^{N} \| \tilde{y}_{i,t} - \tilde{y}_i \|. \]

This result indicates that the first two terms in Lemma 5(a)-(b) can be bounded if the stepsize \( \alpha_t \) is properly chosen, while the remaining terms can be interpreted as errors due to the decentralization and can be bounded using the results in Lemma 4.

### D. Main Results

In this section, we obtain sublinear bounds for the regret and constraint violation defined in (8) and (10), respectively. The next proposition proves sublinearly bounded regret in the cost function.

**Proposition 1:** Let Assumptions [1] - [3] hold. Consider sequences \( \{x_{i,t}\}_{i=1}^{N} \) generated by the online distributed algorithm in (21a) - (21c). Then, with a stepsize choice of
\[ \alpha_t = \frac{1}{\sqrt{t}}, \]
the regret \( R(T) \) with respect to the best fixed offline decision \( x^* \in X \) can be upper bounded as follows: For any \( T \geq 1 \),
\[ R(T) \leq D_1(N) + D_2(N) \sqrt{T}, \]
where the constants \( D_1(N) \) and \( D_2(N) \) are defined in Appendix A.

**Proof:** By adding the two inequalities in Lemma 5(a)-(b), and summing this over \( t = 1, \ldots, T \), we have for any \( x \in X \) and \( \lambda \in \mathbb{R}_+^m \)
\[ \sum_{t=1}^{T} \left[ H_t(x_t, \lambda) - H_t(x, \tilde{\lambda}_t) \right] \]
\[ \leq \sum_{t=1}^{T} \frac{1}{2\alpha_t} \left( \sum_{i=1}^{N} \| x_{i,t} - x_i \|^2 - \sum_{i=1}^{N} \| x_{i,t+1} - x_i \|^2 \right) \]
\[ + \frac{\alpha_t}{2} \sum_{i=1}^{N} \| x_{i,t} - x_i \|^2 - \sum_{i=1}^{N} \| \lambda_{i,t+1} - \lambda_i \|^2 \]
\[ + a_1 \sum_{i=1}^{N} \| \lambda_{i,T} - \tilde{\lambda}_i \| + a_2 \sum_{i=1}^{N} \| \tilde{\lambda}_{i,t} - \tilde{\lambda}_i \| + a_3 \sum_{i=1}^{N} \| \tilde{y}_{i,t} - \tilde{y}_i \|, \]
where
\[ a_1 = \frac{N}{2} \left( L_f + \frac{1}{N} L_g L_F C_\lambda \right)^2 + \frac{1}{2N} C_F^2, \]
\[ a_2 = \frac{1}{N} \left( 2 C_\lambda L_g L_f + C_F \right), \]
\[ a_3 = 2 C_\lambda L_g L_F + 2 C_\lambda L_f. \]

The first term on the right-hand side of (35) can be bounded as
\[ \sum_{t=1}^{T} \frac{1}{2\alpha_t} \left( \sum_{i=1}^{N} \| x_{i,t} - x_i \|^2 - \sum_{i=1}^{N} \| x_{i,t+1} - x_i \|^2 \right) \]
\[ \leq \frac{1}{2\alpha_t} \sum_{i=1}^{N} \| x_{i,t} - x_i \|^2 \]
\[ + \frac{1}{2} \sum_{t=2}^{T} \left( \frac{1}{\alpha_t} - \frac{1}{\alpha_{t-1}} \right) \sum_{i=1}^{N} \| x_{i,t} - x_i \|^2 \]
\[ \leq \frac{2}{\alpha_T} N C_F^2, \]
where we dropped out a negative term \( -\frac{1}{\alpha_T} \sum_{i=1}^{N} \| x_{i,T+1} - x_i \|^2 \) in the first inequality and the second inequality follows from the boundedness of the sets \( X_i \). Similarly, the second term on the right-hand side of (35) can be bounded as
\[ \sum_{t=1}^{T} \frac{1}{\alpha_t} \left( \sum_{i=1}^{N} \| x_{i,t} - x_i \|^2 - \sum_{i=1}^{N} \| \lambda_{i,t+1} - \lambda_i \|^2 \right) \leq \frac{2}{\alpha_T} N C_\lambda^2. \]

Combining these two relations with (35), we obtain
\[ \sum_{t=1}^{T} \left[ H_t(x_t, \lambda) - H_t(x, \tilde{\lambda}_t) \right] \]
\[ \leq \frac{2}{\alpha_T} N C_F^2 + C_F^2 + a_1 \sum_{t=1}^{T} \alpha_t \]
\[ + a_2 \sum_{t=1}^{T} \sum_{i=1}^{N} \| \lambda_{i,t} - \tilde{\lambda}_i \| + a_3 \sum_{t=1}^{T} \sum_{i=1}^{N} \| \tilde{y}_{i,t} - \tilde{y}_i \|. \]

Using the definition of the step size \( \alpha_t \), we further obtain
\[ \sum_{t=1}^{T} \left[ H_t(x_t, \lambda) - H_t(x, \tilde{\lambda}_t) \right] \]
\[ \leq \frac{2}{\alpha_T} N C_F^2 + C_F^2 + a_1 \sum_{t=1}^{T} \alpha_t \]
\[ + a_2 \sum_{t=1}^{T} \sum_{i=1}^{N} \| \tilde{\lambda}_{i,t} - \tilde{\lambda}_i \| + a_3 \sum_{t=1}^{T} \sum_{i=1}^{N} \| \tilde{y}_{i,t} - \tilde{y}_i \|, \]
where we used \( \sum_{t=1}^{T} \alpha_t = \sum_{t=1}^{T} \frac{1}{\sqrt{t}} \leq 1 + \int_{1}^{T} \frac{1}{\sqrt{t}} \, dt \leq 2 \sqrt{T} - 1 \).

Since the above inequality holds for any \( x \in X, \lambda \in \mathbb{R}_+^m \) and \( F \) satisfying Assumption 3 and we now let \( x \not\equiv x^{*} \in X \) (cf. Equation 8) and \( \lambda \not\equiv 0 \). Since \( \sum_{t=1}^{N} \gamma_t(x_t) \leq 0 \), from Assumption 3 we have \( F \left( \sum_{t=1}^{N} \gamma_t(x_t) \right) \leq 0 \) and
\[ \tilde{\lambda}_t F \left( \sum_{i=1}^{N} g_{\ell}(x_{i,t}) \right) \leq 0. \] From this we obtain
\[
\sum_{t=1}^{T} \left[ \mathcal{H}_{\ell}(x_{t}, 0) - \mathcal{H}_{\ell}(x^*, \tilde{\lambda}_t) \right]
= \sum_{t=1}^{T} \sum_{i=1}^{N} \left[ f_{i,t}(x_{i,t}) + \frac{1}{N} 0' F \left( \sum_{\ell=1}^{N} g_{\ell}(x_{\ell,t}) \right) - f_{i,t}(x^*) - \frac{1}{N} \tilde{\lambda}_t' F \left( \sum_{\ell=1}^{N} g_{\ell}(x_{\ell}^*) \right) \right]
\geq \mathcal{R}^c(T).
\]

Using this and Lemma 4 to bound the last two terms on the right-hand side of (37), and arranging all the constants accordingly, we obtain the desired result.

Proposition 1 indicates that the regret is bounded in the order of \( O(\sqrt{T}) \) for any arbitrary \( T \geq 1 \) and grows in the order of \( N^2 \) (cf. see the definition of \( D_1(N) \) and \( D_2(N) \) in Appendix A).

Remark It is worth mentioning that the bound presented in Proposition 1 matches the best known bound for distributed online optimization with time-varying graph topologies (cf. [9], [11], [12]). Note that the bounds in [9], [11] are also in the order of \( N^2 \) (even though they appear to be in the order of \( N \)) as the work in [6] minimizes the average cost \( \frac{1}{N} \sum_{i=1}^{N} f_{i,t}(x) \), and in [11] their initialization term is in the order of \( N \). In [11], [12] weight-imbalanced graphs are considered, but they cannot handle globally coupled constraints. Moreover, an additional assumption is made in [11] that the objective functions are strongly convex.

In the next proposition, we prove a sublinear bound for the constraint violation with an additional assumption on \( F \).

Proposition 2: Let Assumptions 1-3 hold. Assume that the function \( F \) is a penalty function, i.e., \( F \) is chosen such that \( \mathcal{F}(x) \rangle_i > 0 \) if \( |x_i| > 0 \) and \( \mathcal{F}(x) \rangle_i = 0 \), otherwise. Consider sequences \{\( x_{i,t} \)\}_{t \geq 1} \( i \in V \) generated by the online distributed algorithm in (21a)-(21e). Then, with a stepsize choice of
\[ \alpha_t = \frac{1}{\sqrt{t}}. \]

\( \mathcal{R}^c(T) \) in (10) can be upper bounded as follows: For any \( T \geq 1 \),
\[ \mathcal{R}^c(T) \leq D_3(N) + D_4(N) \sqrt{T}, \]
where the constants \( D_3(N) \) and \( D_4(N) \) are defined in Appendix A.

Proof: From algorithm (21d) with \( \Lambda = \mathcal{R}_{\ell}^\alpha \), we have
\[ \tilde{\lambda}_{i,t} + \frac{\alpha_t}{N} \mathcal{F}(N\tilde{y}_{i,t}) \geq \left[ \tilde{\lambda}_{i,t} + \frac{\alpha_t}{N} \mathcal{F}(N\tilde{y}_{i,t}) \right]_+ = \lambda_{i,t+1}. \]

Summing this over \( i \in V \) and rearranging the terms, we have for any \( 1 \leq t \leq T \):
\[ \frac{1}{N} \sum_{i=1}^{N} \mathcal{F}(N\tilde{y}_{i,t}) \leq \frac{\sum_{i=1}^{N} \lambda_{i,t+1} - \sum_{i=1}^{N} \tilde{\lambda}_{i,t}}{\alpha_t} = \frac{\sum_{i=1}^{N} \lambda_{i,t+1} - \sum_{i=1}^{N} \lambda_{i,t}}{\alpha_t} \]  
(38a)
where (38a) follows from the relation
\[ \sum_{i=1}^{N} \lambda_{i,t} = \sum_{i=1}^{N} \sum_{j=1}^{N} (W_{i,j})_{ij} \leq \sum_{j=1}^{N} \sum_{i=1}^{N} (W_{i,j})_{ij} = N, \]
and (38b) follows from the fact that the step-size \( \{\alpha_t\} \) is a decreasing sequence. Summing the preceding relation over \( t = 1, \ldots, T \), we obtain
\[ \frac{1}{N} \sum_{t=1}^{T} \sum_{i=1}^{N} \mathcal{F}(N\tilde{y}_{i,t}) \leq \frac{\sum_{i=1}^{N} \lambda_{i,t+1} - \sum_{i=1}^{N} \lambda_{i,1}}{\alpha_T}, \]
(38b)
where (38a) follows from the telescoping sum of relation (38b) and (38b) follows from \( \sum_{i=1}^{N} \lambda_{i,1} \geq 0 \).

We now estimate the following quantity:
\[ \sum_{t=1}^{T} N \mathcal{F} \left( \sum_{i=1}^{N} g_{\ell}(x_{i,t}) \right) \]
\[ = \sum_{t=1}^{T} N \mathcal{F} \left( \sum_{i=1}^{N} g_{\ell}(x_{i,t}) \right) - \sum_{t=1}^{T} \sum_{i=1}^{N} \mathcal{F}(N\tilde{y}_{i,t}) + \sum_{t=1}^{T} \sum_{i=1}^{N} \mathcal{F}(N\tilde{y}_{i,t}) \]
\[ \leq \sum_{t=1}^{T} \sum_{i=1}^{N} \left[ \mathcal{F}(N\tilde{y}_{i,t}) - \mathcal{F}(N\tilde{y}_{i,t}) \right] + \frac{\sum_{i=1}^{N} \lambda_{i,T+1}}{\alpha_T}, \]
(40b)
where (40a) is from addition and subtraction; (40b) follows from Lemma 1 and relation and (39b). Taking the norm on the above relation and dividing by \( N \), we obtain
\[ \left\| \sum_{t=1}^{T} \mathcal{F} \left( \sum_{i=1}^{N} g_{\ell}(x_{i,t}) \right) \right\| \leq L_{\mathcal{F}} \sum_{t=1}^{T} \sum_{i=1}^{N} \| \tilde{y}_{i,t} - \tilde{y}_{i,t} \| + \frac{NC_{\lambda}}{\alpha_T}, \]
where in the last inequality, we applied relation (28) and Assumption 2 for bounding the dual variables. Lastly, by using Lemma 4 for bounding the first term on the right-hand side, we obtain the desired result.

Note that the bounds in propositions 1 and 2 above capture their dependency on the minimum weight \( \eta \) and the connectivity \( Q \) (cf. Assumption 1) through the constants \( \gamma \) and \( \beta \). Specifically, the constants \( D_1(N), D_2(N), D_3(N) \) and \( D_4(N) \) increase as the minimum weight \( \eta \) decreases or the strong connectivity parameter \( Q \) increases.

VI. SIMULATION RESULTS

In this section, we perform some numerical experiments to solve the problem (16)-(17) using Algorithm 1. In these simulations, the average regrets \( \mathcal{R}(T)/T \) and \( \mathcal{R}^c(T)/T \) are monitored as a metric of convergence.

We consider networks consisting of \( N = 10, 15, \) or 20 sources and \( K = 2 \) access points, wherein the agents were randomly and uniformly distributed in rectangular boxes. Note
that only the source nodes participate in the computation. Figure 2 depicts a snapshot of 20 source nodes and 2 access points. The gray lines connecting the nodes represent the current routing rates $R_{ij}$, where thicker lines correspond to higher values.

We model $R_{ij}$ as a twice differentiable, decreasing function of the inter-node distance (normalized between 0 and 1) which is a polynomial fitting of curves found in literature [38]. Specifically, let $p_i, p_j \in \mathbb{R}^2$ be the positions of node $i$ and $j$, and $p_{ij} = p_i - p_j$. Then, $R_{ij}$ is modeled as

$$R_{ij}(p_i, p_j) = a\|p_{ij}\|^2 + b\|p_{ij}\|^2 + c\|p_{ij}\|^2 + d$$

if $l \leq \|p_{ij}\| \leq u$, $R_{ij}(p_i, p_j) = 0$ if $\|p_{ij}\| < l$, and $R_{ij}(p_i, p_j) = 1$ if $\|p_i - p_j\| > u$, where the constants $0 < l < u < 1$ are lower and upper bounds on the inter-node distances, and $a, b, c$ and $d$ are defined as

$$a = \frac{-2}{(l-u)^2}, b = \frac{3(l+u)}{(l-u)^3}, c = \frac{-6lu}{(l-u)^3}, d = \frac{3lu^2}{(l-u)^3}.$$ Figure 3 depicts inter-node distance vs. $R_{ij}$ with $l = 0.5$ and $u = 0.8$.

In the experiment, we corrupt $R_{ij}$ with uniform random noise. When $R_{ij}$ goes above 1 or below 0, we truncate it to 1 and 0. We also take out some of the edges such that each sequence of graphs is $Q$-strongly connected with $Q = 1, 5, 10$ (cf. Assumption [1]). The communication matrices $W_i$ are defined as $W_{ij} = 1/N$ (therefore, $\eta = 1/N$) if the random realization of $R_{ij}$ at time $i$ is nonzero, and $W_{ij} = 0$ otherwise. The diagonal entries are set to be $W_{ii} = 1 - \sum_{j \neq 1} W_{ij}$. We set $r_{i,\text{min}} = 0.001$ for all $i$.

For the function $F(x)$, we choose a smooth surrogate of the max function $|x|_+$. Such a max function can be found, for example, using the smoothing technique in [39]. That is, for an arbitrary $\mu > 0$, a smooth surrogate of $|x|_+$ for $x \in \mathbb{R}$ is defined as

$$[x]_{\pm}^\mu = \begin{cases} \frac{1}{\mu^2}(x + \mu)^2 & \text{if } -\mu \leq x \leq \mu, \\ x & \text{if } \mu > x, \\ 0 & \text{otherwise.} \end{cases}$$ (41)

This function and its gradient are Lipschitz continuous with $L = 1$ and $G = \frac{1}{\mu}$. Therefore, a component wise max surrogate $F(x) = [x]_{+}^\mu$ satisfies Assumption [3] with constants $L_F = \sqrt{m}$ and $G_F = \frac{\sqrt{m}}{\mu}$. Figure 4 depicts $F(x) = [x]_{+}^\mu$ with $\mu = 1$ and $\mu = 5$. Note that the parameter $\mu$ controls the amount of smoothness, i.e., $\mu$ can be made arbitrarily small at the cost of a larger $G_F$. In the simulation, we used $\mu = 0.001$. Note that there are also a number of other methods for finding a penalty function $F$ satisfying Assumption [3] like piecewise polynomial interpolation techniques in [40].

In what follows, we illustrate convergence of our algorithm for different network sizes. Figure 5 shows regrets in terms of cost function value and constraint violation for $N = 10, 15$, and 20 node networks with $Q = 1$. It shows that the regrets are sublinear for all cases and their average $R(T)/T$ and $\mathcal{R}^o(T)/T$ go to zero as the time increases. It also shows that the convergence speed gets slower as the number of network nodes increases. To further investigate the effect of connectivity, in Figure 6 we depict regrets for a $N = 10$ node network with different values of $Q = 1, 5$ and 10. It shows that the convergence speed in the cost function value gets slower as $Q$ increases, while different values of $Q$ does not affect much the convergence in constraint violation.
We developed a new decentralized primal-dual method for online distributed optimization involving global constraints. Under the assumption that the cost and constraint functions are Lipschitz continuous, we showed that the algorithm achieves worst-case individual regret of order $O(\sqrt{T})$ on any sequence of time-varying, weight-balanced and jointly connected graphs. Numerical results illustrating the performance of the proposed algorithm are provided for optimal routing in wireless multi-hop networks with uncertain channel rates.

REFERENCES

[1] M. Rabbat and R. D. Nowak, “Distributed optimization in sensor networks,” in Information Processing in Sensor Networks, 2004, pp. 20–27.

[2] A. Jadbabaie, J. Lin, and A. Morse, “Coordination of groups of mobile autonomous agents using nearest neighbor rules,” IEEE Transactions on Automatic Control, 2003.

[3] S. S. Ram, V. V. Veeravalli, and A. Nedić, “Distributed non-autonomous power control through distributed convex optimization,” in IEEE International Conference on Computer Communications, 2009, pp. 3001–3005.

[4] J. W. Durham, A. Franchi, and F. Bullo, “Distributed pursuit-evasion without mapping or global localization via local frontiers,” Autonomous Robots, vol. 32, no. 1, pp. 81–95, 2012.

[5] M. Zinkevich, “Online convex programming and generalized infinitesimal gradient ascent,” in Proceedings of the 20th International Conference on Machine Learning, 2003, pp. 928–936.

[6] E. Hazan, A. Agarwal, and S. Kale, “Logarithmic regret algorithms for online convex optimization,” Journal of Machine Learning, vol. 69, no. 2-3, pp. 169–192, 2007.

[7] S. Shalev-Shwartz, “Online learning and online convex optimization,” Foundations and Trends in Machine Learning, vol. 4, no. 2.

[8] S. Hosseini, A. Chapman, and M. Mesbahi, “Online distributed optimization via dual averaging,” in IEEE 52nd Conference on Decision and Control, 2013, pp. 1484–1489.

[9] ——, “Online distributed optimization on dynamic networks,” 2014, http://arxiv.org/abs/1412.7215.

[10] Y. Nesterov, “Primal-dual subgradient methods for convex problems,” Mathematical Programming, vol. 120, no. 1, pp. 221–259, 2009.

[11] M. Akbari, B. Gharesifard, and T. Linder, “Distributed subgradient-push online convex optimization on time-varying directed graphs,” in 52nd Annual Allerton Conference on Communication, Control, and Computing, 2014, pp. 264–269.

[12] S. Lee, A. Nedić, and M. Raginsky, “Decentralized online optimization with global objectives and local communication,” 2015, http://arxiv.org/abs/1508.07933.

[13] K. Tsianos, S. Lawlor, and M. Rabbat, “Push-sum distributed dual-averaging for convex optimization,” in Proceedings of the 51st IEEE Conference on Decision and Control, 2012, pp. 5453–5458.

[14] A. Nedić and A. Olshevsky, “Distributed optimization over time-varying directed graphs,” IEEE Transactions on Automatic Control, vol. 60, no. 3, pp. 601–615, 2015.

[15] A. Nedić and A. Ozdaglar, “Distributed subgradient methods for multi-agent optimization,” IEEE Transactions on Automatic Control, vol. 54, no. 1, pp. 48–61, 2009.

[16] S. Hosseini, A. Chapman, and M. Mesbahi, “Online distributed admm on networks,” 2014, http://arxiv.org/abs/1412.7716.
B. Proof of Lemma 2

Equality (a) follows directly from the definition of $\tilde{y}_t$. Equality (b) is also straightforward from the algorithm definition in (24b). That is,

$$\sum_{i=1}^{N} \tilde{y}_{i,t} = \sum_{i=1}^{N} \sum_{j=1}^{N} |W_{ij}| y_{j,t} = \sum_{j=1}^{N} y_{j,t} \sum_{i=1}^{N} |W_{ij}| = \sum_{j=1}^{N} y_{j,t},$$

(42)

where the last equality follows from the doubly stochasticity of $W_i$. Lastly, we prove equality (c) by induction on $t$. From the initialization assumption, we have $y_{i,0} = g_i(x_{i,0})$ for all $i \in \mathcal{V}$, and therefore, $\sum_{i=1}^{N} y_{i,0} = \sum_{i=1}^{N} g_i(x_{i,0})$. Assume that for some $t > 0$ there holds

$$\sum_{i=1}^{N} y_{i,t} = \sum_{i=1}^{N} g_i(x_{i,t}).$$

(43)

Then, we have

$$\sum_{i=1}^{N} y_{i,t+1} = \sum_{i=1}^{N} \tilde{y}_{i,t} + \sum_{i=1}^{N} g_i(x_{i,t+1}) - \sum_{i=1}^{N} g_i(x_{i,t})$$

(44a)

$$= \sum_{i=1}^{N} y_{i,t} + \sum_{i=1}^{N} g_i(x_{i,t+1}) - \sum_{i=1}^{N} g_i(x_{i,t})$$

(44b)
where (44a) follows from relation (21e); (44b) follows from equality (42); and (44c) follows from the induction hypothesis in (43). Therefore, equation (43) also holds for \( t + 1 \) and this proves that equation (43) holds for any \( t \geq 1 \).

C. Proof of Lemma 3

Using the triangle inequality, we can write

\[
\|y_{i,t} - \bar{y}_t\| \leq \|y_{i,t} - \tilde{y}_t\| + \|\tilde{y}_t\| \quad (45)
\]

From Lemma 1 and relation (27), the second term can be bounded as

\[
\|\tilde{y}_t\| = \frac{1}{N} \left\| \sum_{i=1}^{N} g_i(x_{i,t}) \right\| \leq C_g. \quad (46)
\]

In order to bound the first term in (45), we make use of Lemma 2. Note that from Assumption 2, the error in (32) can be bounded for any \( i \in \mathcal{V} \) and \( t \geq 1 \)

\[
\|\epsilon_{i,t+1}\| = \|g_i(x_{i,t+1}) - g_i(x_{i,t})\|
\leq L_g \|x_{i,t+1} - x_{i,t}\| \leq 2L_g C_x.
\]

Therefore, from Lemma 2 and this relation, we have for any \( t \geq 1 \)

\[
\|y_{i,t+1} - \bar{y}_{t+1}\| \leq N \gamma \max_j \|y_{j,i}\|
+ \frac{2\gamma \beta}{1 - \beta} L_g C_x + 4L_g C_x,
\]

and

\[
\|y_{i,t} - \bar{y}_t\| \leq \max \left\{ \|y_{i,1} - \bar{y}_1\|, \, N \gamma \max_j \|y_{j,1}\|ight.
+ \left. \frac{2\gamma \beta}{1 - \beta} L_g C_x + 4L_g C_x \right\}. \quad (49)
\]

Substituting this and relation (46) into (45), we can obtain the desired result.

D. Proof of Corollary 7

By summing the relation (30) over \( t = 1, \ldots, T - 1 \), we obtain

\[
\sum_{t=1}^{T-1} \|\theta_{i,t+1} - \bar{\theta}_{t+1}\| \leq N \gamma \max_j \|\theta_{j,1}\| \sum_{t=1}^{T-1} \beta^t \quad (47)
\]

and

\[
\sum_{t=1}^{T-1} \|\epsilon_{i,t+1}\| \leq \frac{\gamma \max_j \|y_{j,1}\|}{1 - \beta} \sum_{t=1}^{T-1} \beta^t \quad (48)
\]

Using \( \sum_{t=1}^{T-1} \beta^t \leq \frac{\beta}{1 - \beta} \), relation (48), and the assumption \( \|\epsilon_{i,t}\| \leq K \alpha_t \) to upper bound inequality (47), we obtain the desired result.

E. Proof of Lemma 4

First, we obtain the following chain of relations:

\[
\sum_{i=1}^{N} \|\bar{\lambda}_{i,t} - \hat{\lambda}_i\| \leq \sum_{i=1}^{N} \|W_i\| \|\lambda_{j,t} - \bar{\lambda}_i\| \quad (49a)
\]

\[
\leq \sum_{j=1}^{N} \|\lambda_{j,t} - \bar{\lambda}_i\| \sum_{i=1}^{N} \|W_i\| \quad (49b)
\]

\[
\leq \sum_{i=1}^{N} \|\lambda_{i,t} - \bar{\lambda}_i\|. \quad (49c)
\]

where (49a) follows from the definition of \( \bar{\lambda}_{i,t} \) in (21a) and the convexity of the norm function and the doubly stochasticity of \( W_i \); (49b) follows from reordering of the summations; and (49c) follows from the doubly stochasticity of \( W_i \). Similarly, we can also show that

\[
\sum_{i=1}^{N} \|\tilde{y}_{i,t} - \bar{y}_t\| \leq \sum_{i=1}^{N} \|y_{i,t} - \tilde{y}_t\|. \quad (50)
\]

In what follows, we estimate \( \|\epsilon_{i,t+1}\| \) in (32) to make use of Corollary 1.

\[
\|\epsilon_{i,t+1}\| = \|P_A \left[ \bar{\lambda}_{i,t} + \frac{\alpha_t}{N} F(N \tilde{y}_{i,t}) \right] - \hat{\lambda}_{i,t}\|
\leq \frac{\alpha_t}{N} \|F(N \tilde{y}_{i,t})\| \leq \frac{\alpha_t}{N} C_F, \quad (51a)
\]

where (51a) follows from the nonexpansiveness of the projection operator together with the fact that \( \bar{\lambda}_{i,t} \in A \) and (51b) follows from the relation (34). Now we can invoke Corollary 1 with \( \theta_{i,t} := \bar{\lambda}_{i,t} \) and \( R := \frac{\rho_F}{N} \). Hence, we have for all \( i \in \mathcal{V} \)

\[
\sum_{t=1}^{T-1} \|\lambda_{i,t+1} - \bar{\lambda}_{t+1}\| \leq \gamma \max_j \|\lambda_{j,1}\|
+ \left( \frac{2C_F \beta}{1 - \beta} + \frac{2C_F}{N} \right) \sum_{t=1}^{T-1} \alpha_t,
\]

and consequently

\[
\sum_{t=1}^{T} \|\lambda_{i,t} - \bar{\lambda}_i\| \leq \|\lambda_{i,1} - \bar{\lambda}_i\| + \gamma \max_j \|\lambda_{j,1}\|
+ \left( \frac{2C_F \beta}{1 - \beta} + \frac{2C_F}{N} \right) \sum_{t=1}^{T} \alpha_t. \quad (52)
\]

From the definition of the step size \( \alpha_t \), we have

\[
\sum_{t=1}^{T} \alpha_t = \frac{1}{\sqrt{t}} \leq 1 + \int_1^{T} \frac{1}{\sqrt{t}} dt \leq 2\sqrt{T} - 1.
\]
Combining this with relation (52), using Assumption 2 and relation (49), we obtain the desired result.

Similarly, the error $|e_{i,t+1}|$ in (33) can be estimated as follows:

$$
\|e_{i,t+1}\| = \|g_i(x_{i,t+1}) - g_i(x_{i,t})\| \leq L_g \|x_{i,t+1} - x_{i,t}\| = L_g \|P_{X_i}[x_i,t - \alpha_i s_{i,t}] - x_{i,t}\| \leq \alpha_i L_g \left( L_f + \frac{1}{N} L_g L_f C_X \right),
$$

where (53b) follows from relation (25), (53c) follows from the definition of $x_{i,t+1}$ in algorithm (21c) and the nonexpansiveness of the projection operator $P_{X_i}[\cdot]$; and (53d) follows from the definition of $s_{i,t}$ in (20) and $\|s_{i,t}\| \leq L_f + \frac{1}{N} L_g L_f C_X$. By invoking Corollary 1 with $\theta_{i,t} := \tilde{y}_{i,t}$ and $K := L_g L_f + \frac{1}{N} L_g L_f C_X$ and using the exactly same line of arguments in estimating $\sum_{t=1}^{T} \|\lambda_{i,t} - \tilde{\lambda}_i\|$, the desired result follows.

F. Proof of Lemma 5

(a) From algorithm (21c) and the nonexpansiveness of the projection operator $P_{X_i}[\cdot]$, we have for any $x \in X$ and $t \geq 1$:

$$
\sum_{i=1}^{N} \|x_{i,t+1} - x_i\|^2 \leq \sum_{i=1}^{N} \|x_{i,t} - \alpha_i s_{i,t} - x_i\|^2 = \sum_{i=1}^{N} \|x_{i,t} - x_i\|^2 + \alpha_i^2 \sum_{i=1}^{N} \|s_{i,t}\|^2 - 2\alpha_i \sum_{i=1}^{N} s_{i,t}^N (x_{i,t} - x_i).
$$

We can rewrite the last term on the right-hand side as the following:

$$
\sum_{i=1}^{N} s_{i,t}^N (x_{i,t} - x_i) = \sum_{i=1}^{N} \nabla f_{i,t}(x_{i,t})'(x_{i,t} - x_i)
$$

$$
= \sum_{i=1}^{N} \nabla g_i(x_{i,t})'(x_{i,t} - x_i) + \frac{1}{N} \sum_{i=1}^{N} \left( \frac{\|\nabla g_i(x_{i,t})\|}{\|\nabla f_{i,t}(x_{i,t})\|} \|\nabla f_{i,t}(x_{i,t})\| \right) (x_{i,t} - x_i)
$$

$$
= \sum_{i=1}^{N} \left( \frac{\|\nabla g_i(x_{i,t})\|}{\|\nabla f_{i,t}(x_{i,t})\|} \|\nabla f_{i,t}(x_{i,t})\| \right) (x_{i,t} - x_i) + \frac{1}{N} \sum_{i=1}^{N} [\nabla g_i(x_{i,t})'(\nabla f_{i,t}(x_{i,t}) - \nabla f_{i,t}(\hat{y}_{i,t})) \hat{\lambda}_{i,t}](x_{i,t} - x_i)
$$

$$
+ \frac{1}{N} \sum_{i=1}^{N} [\nabla g_i(x_{i,t})'(\nabla f_{i,t}(\hat{y}_{i,t}) - \nabla f_{i,t}(\hat{y}_{i,t})) \hat{\lambda}_{i,t}](x_{i,t} - x_i)
$$

$$
+ \frac{1}{N} \sum_{i=1}^{N} [\nabla g_i(x_{i,t})'(\nabla f_{i,t}(\hat{y}_{i,t}) - \nabla f_{i,t}(\hat{y}_{i,t})) \hat{\lambda}_{i,t}](x_{i,t} - x_i)
$$

$$
\geq \sum_{i=1}^{N} \left( \frac{\|\nabla g_i(x_{i,t})\|}{\|\nabla f_{i,t}(x_{i,t})\|} \|\nabla f_{i,t}(x_{i,t})\| \right) (x_{i,t} - x_i) - 2C_x C_s L_g G_{xf} \sum_{i=1}^{N} \|\hat{y}_{i,t} - \bar{y}_{i,t}\|
$$

$$
- \frac{2}{N} C_x C_s L_g G_{xf} \sum_{i=1}^{N} \|\hat{\lambda}_{i,t} - \bar{\lambda}_i\|
$$

$$
\geq \mathcal{H}_f(x_{i,t}, \hat{\lambda}_i) - \mathcal{H}_f(x_{i,t}, \bar{\lambda}_i)
$$

$$
\geq \frac{1}{N} \sum_{i=1}^{N} \left( \frac{\|\nabla g_i(x_{i,t})\|}{\|\nabla f_{i,t}(x_{i,t})\|} \|\nabla f_{i,t}(x_{i,t})\| \right) (x_{i,t} - x_i) - 2C_x C_s L_g G_{xf} \sum_{i=1}^{N} \|\hat{y}_{i,t} - \bar{y}_{i,t}\|
$$

$$
\geq \mathcal{H}_f(x_{i,t}, \hat{\lambda}_i) - \mathcal{H}_f(x_{i,t}, \bar{\lambda}_i)
$$

where (55a) follows from the definition of $s_{i,t}$ in (20); (55b) follows from adding and subtracting terms accordingly; (55c) follows from the convexity of $\mathcal{H}_f(\cdot, \lambda)$. Combining (55a)-(55d) with (54), using $\|s_{i,t}\| \leq L_f + \frac{1}{N} L_g L_f C_X$ and rearranging terms, we obtain the desired result.

(b) From algorithm (21d), we have for any $\lambda \in \Lambda$ and $t \geq 1$:

$$
\sum_{i=1}^{N} \|\lambda_{i,t+1} - \lambda\|^2 = \sum_{i=1}^{N} \|\lambda_{i,t} - \lambda\|^2 + \frac{\alpha_t}{N} \sum_{i=1}^{N} \|\lambda_{i,t} - \lambda\|^2 + \frac{2\alpha_t}{N} \sum_{i=1}^{N} \|\lambda_{i,t} - \lambda\|^2.
$$

Similarly to (55a)-(55d), we have the following chain of relations for the last term on the right-hand side:

$$
\frac{1}{N} \sum_{i=1}^{N} \|\lambda_{i,t} - \lambda\|^2 = \frac{1}{N} \sum_{i=1}^{N} \left[ \|\lambda_{i,t} - \lambda\|^2 + \|\lambda_{i,t} - \lambda\|^2 \right]
$$

$$
= \frac{1}{N} \sum_{i=1}^{N} \left[ \|\lambda_{i,t} - \lambda\|^2 + \|\lambda_{i,t} - \lambda\|^2 \right] + \frac{1}{N} \sum_{i=1}^{N} \|\lambda_{i,t} - \lambda\|^2
$$

$$
\leq \mathcal{H}_i(x_{i,t}, \hat{\lambda}_i) - \mathcal{H}_i(x_{i,t}, \bar{\lambda}_i)
$$

$$
+ \frac{C_{xf}}{N} \sum_{i=1}^{N} \|\hat{y}_{i,t} - \bar{y}_{i,t}\| + 2C_x C_s L_g G_{xf} \sum_{i=1}^{N} \|\hat{y}_{i,t} - \bar{y}_{i,t}\|
$$

where (57a) follows from adding and subtracting terms accordingly; (57b) follows from the definition of $\mathcal{H}_i(\cdot, \lambda)$ together with the relation $N\hat{y}_t = \sum_{i=1}^{N} g_i(x_{i,t})$ from Lemma 1 Schwartz inequalities, and the boundedness Assumptions. Combining this relation with (56), using $\|\mathcal{H}_i(x_{i,t}, \hat{\lambda}_i) - \mathcal{H}_i(x_{i,t}, \bar{\lambda}_i)\| \leq C_{xf} \sum_{i=1}^{N} \|\hat{y}_{i,t} - \bar{y}_{i,t}\|^2 \leq \sum_{i=1}^{N} \|\hat{\lambda}_{i,t} - \lambda\|^2$ and rearranging terms, we obtain the desired result.