A Distributed Primal-Dual Algorithm for Bandit Online Convex Optimization with Time-Varying Coupled Inequality Constraints

Xinlei Yi, Xiuxian Li, Tao Yang, Lihua Xie, Tianyou Chai, and Karl H. Johansson

Abstract—This paper considers distributed bandit online optimization with time-varying coupled inequality constraints. The global cost and the coupled constraint functions are the summations of local convex cost and constraint functions, respectively. The local cost and constraint functions are held privately and only at the end of each period the constraint functions are fully revealed, while only the values of cost functions at queried points are revealed, i.e., in a so called bandit manner. A distributed bandit online primal-dual algorithm with two queries for the cost functions per period is proposed. The performance of the algorithm is evaluated using its expected regret, i.e., the expected difference between the outcome of the algorithm and the optimal choice in hindsight, as well as its constraint violation. We show that \( O(T^c) \) expected regret and \( O(T^{1-c/2}) \) constraint violation are achieved by the proposed algorithm, where \( T \) is the total number of iterations and \( c \in [0.5, 1) \) is a user-defined trade-off parameter. Assuming Slater’s condition, we show that \( O(\sqrt{T}) \) expected regret and \( O(\sqrt{T}) \) constraint violation are achieved. The theoretical results are illustrated by numerical simulations.

I. INTRODUCTION

Online convex optimization is a promising methodology for modeling sequential tasks and can be traced back to the 1990s [1]–[4]. It has important applications in machine learning and control, see, e.g., [5]–[10]. Bandit online convex optimization is online convex optimization with so called bandit feedback meaning that in each period only the values of the cost functions at some points are revealed, rather than other information such as the gradient of the cost function. Gradient information may not be available in many applications, such as online source localization, online routing in data networks, and online advertisement placement in web search [7]. Essentially, bandit online convex optimization is a derivative-free method to solve convex optimization problems. Derivative-free methods have an evident advantage since computing a function value is much simpler than computing its gradient [11]. Early works of studying bandit online convex optimization include [12], [13], where the expected regret is used to measure the performance of the algorithms. The expected regret is the expected difference between the outcome of the algorithm and the optimal choice in hindsight.

A key step in bandit online convex optimization is to estimate the gradient of the cost function through querying the cost function. Various algorithms have been developed and can be divided into two categories depending on the number of queries. Algorithms with one query per period have been proposed in [13]–[22]. Algorithms with two or more queries per period have been proposed in [22]–[28] and the expected regret bounds can then be further reduced.

Aforementioned studies did not consider equality or inequality constraints. In the literature, there are only few papers considering bandit online convex optimization with equality or inequality constraints, although such constraints are common in applications. The authors of [29] studied online convex optimization with static inequality constraints and bandit feedback for constraints. They proposed an algorithm with two queries per period and achieved \( O(\sqrt{T}) \) and \( O(T^{3/4}) \) bounds on the expected regret and constraint violation, respectively. The authors of [30] studied online convex optimization with time-varying inequality constraints and bandit feedback for cost functions. Under Slater’s conditions, they proposed a class of algorithms with one or two queries per period.

Most existing bandit online convex optimization studies are in a centralized setting and only few papers considered distributed bandit online convex optimization. When cost functions are strongly convex, the authors of [31] proposed a consensus-based distributed bandit online algorithm with one query per period and obtained \( O(\sqrt{T \log(T)}) \) expected regret. When cost functions are quadratic, the authors of [32] proposed a consensus-based distributed bandit online algorithm with two queries per period and obtained \( O(\sqrt{T}) \) expected regret when there are set constraints.

This paper studies distributed bandit online convex optimization with time-varying coupled inequality constraints. The global cost and the coupled constraint functions are the summations of local convex cost and constraint functions, respectively. The local cost and constraint functions are held privately and only at the end of each period the constraint functions are fully revealed, while the values of the cost functions are revealed in a bandit manner. Specifically, per period each agent can sample the value of its local cost function at two points and can observe the value and the exact gradient of its local constraint function at one point. Compared to existing studies, the contributions of this paper are summarized as follows.
1) We develop a distributed bandit online primal-dual algorithm, where per period each agent uses two queries to estimate the gradient of its local cost function. The proposed algorithm uses different non-increasing stepsize sequences for the primal and dual updates and a non-increasing sequence of regularization parameters. Moreover, it also uses non-increasing shrinkage and exploration sequences in the gradient estimation model. These sequences give some freedom in the expected regret and constraint violation bounds, as they allow the trade-off between how fast these two bounds tend to zero. It should be highlighted that the total number of iterations is not used as a variable in the algorithm, which is different from most existing bandit online algorithms.

2) For the proposed algorithm, we show that $O(T^{1-c/2})$ expected regret and $O(T^{1-c/2})$ constraint violation are achieved, where $c \in [0.5, 1)$ is a user-defined trade-off parameter. Compared with the bandit algorithm in [25], which achieved $O(\sqrt{T})$ expected regret under static set constraints and centralized computations using the total number of iterations in the algorithm, we are relaxing all these assumptions.

3) Assuming Slater’s condition, we show that $O(\sqrt{T})$ expected regret and $O(\sqrt{T})$ constraint violation can be achieved. Although the two-query bandit algorithm in [30] also achieved the same expected regret and constraint violation bounds, it is a centralized algorithm and uses the total number of iterations. Moreover, a slightly stronger Slater’s condition was assumed in [30].

The rest of this paper is organized as follows. Section II introduces the preliminaries. Section III gives the problem formulation. Section IV provides a distributed bandit online algorithm and presents expected regret and constraint violation bounds. Section V gives numerical simulations. Finally, Section VI concludes the paper.

**Notations:** All inequalities and equalities are understood componentwise. $[n]$ represents the set $\{1, \ldots, n\}$ for any $n \in \mathbb{N}_+$. $\mathbb{R}^p$ stands for the unit sphere centered around the origin in $\mathbb{R}^p$ under Euclidean norm. $\text{col}(z_1, \ldots, z_k)$ represents the concatenated column vector of vectors $z_i \in \mathbb{R}^{m_i}, i \in [k]$. Given two scalar sequences $\{\alpha_t, t \in \mathbb{N}_+\}$ and $\{\beta_t > 0, t \in \mathbb{N}_+\}$, $\alpha_t = O(\beta_t)$ means that $\lim \sup_{t \to \infty} (\alpha_t/\beta_t)$ is bounded. For a set $\mathcal{K} \subseteq \mathbb{R}^p$, $\mathcal{P}_{\mathcal{K}}(\cdot)$ denote the projection operator, i.e., $\mathcal{P}_{\mathcal{K}}(x) = \arg \min_{y \in \mathcal{K}} \|x - y\|^2, \forall x \in \mathbb{R}^p$. For simplicity, $[\cdot]_+$ is used to denote $\mathcal{P}_{\mathcal{K}}(\cdot)$ when $\mathcal{K} = \mathbb{R}^p$.

**II. Preliminaries**

In this section, we present some definitions and assumptions related to graph theory and gradient approximation.

**A. Graph Theory**

Interactions between agents are modeled by a time-varying directed graph. Specifically, at time $t$, agents communicate with other agents according to a directed graph $G_t = (\mathcal{V}, \mathcal{E}_t)$, where $\mathcal{V} = [n]$ is the agent set and $\mathcal{E}_t \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set. Let $\mathcal{N}_{\text{in}}^t(G_t) = \{j \in [n] \mid (j, i) \in \mathcal{E}_t\}$ and $\mathcal{N}_{\text{out}}^t(G_t) = \{j \in [n] \mid (i, j) \in \mathcal{E}_t\}$ be the sets of in- and out-neighbors, respectively, of agent $i$ at time $t$. The mixing matrix $W_i \in \mathbb{R}^{n \times n}$ at time $t$ fulfills $[W_i]_{ij} > 0$ if $(i, j) \in \mathcal{E}_t$ or $i = j$, and $[W_i]_{ij} = 0$ otherwise.

The following standard assumption is made on the graph, see e.g., [33].

**Assumption 1.** For any $t \in \mathbb{N}_+$, the graph $G_t$ satisfies the following conditions: (i) There exists a constant $w \in (0, 1)$, such that $[W_i]_{ij} \geq w$ if $[W_i]_{ij} > 0$; (ii) The mixing matrix $W_i$ is doubly stochastic, i.e., $\sum_{i=1}^n [W_i]_{ij} = \sum_{j=1}^n [W_i]_{ij} = 1$, $\forall i, j \in [n]$; (ii) There exists an integer $\varepsilon > 0$ such that the graph $(\mathcal{V}, \cup_{t=0,\ldots,T} \mathcal{E}_t)$ is strongly connected.

**B. Gradient Approximation**

Let $f : \mathcal{K} \to \mathbb{R}$ be a function with $\mathcal{K} \subset \mathbb{R}^p$. We assume that $\mathcal{K}$ is bounded and has a nonempty interior. Without loss of generality, we assume that $\mathcal{K}$ contains the ball of radius $r(\mathcal{K})$ centered at the origin and is contained in the ball of radius $R(\mathcal{K})$, i.e., $r(\mathcal{K})B^p \subseteq \mathcal{K} \subseteq R(\mathcal{K})B^p$. We use the following gradient estimator

$$\hat{\nabla}_2 f(x) = \frac{D}{2}\left(f(x + du) - f(x)\right)u, \ \forall x \in (1 - \xi)\mathcal{K}, \quad (1)$$

where $u$ is a random vector which is uniformly distributed in the unit sphere $\mathbb{S}^p$, $\delta \in (0, r(\mathcal{K})\xi]$ is an exploration parameter, and $\xi \in (0, 1)$ is a shrinkage coefficient. The estimator (1) thus requires to sample the function at two points, so it is a two-query model. The intuition follows from directional derivatives [24]. The properties of $f$ can be found in [34].

**III. Problem Formulation**

Consider a network of $n$ agents indexed by $i \in [n]$. For each $i$, let the local decision set $X_i \subseteq \mathbb{R}^p$ be a closed convex set with $p_i$ being a positive integer. Let $\{f_{i,t} : X_i \to \mathbb{R}\}$ and $\{g_{i,t} : X_i \to \mathbb{R}^m\}$ be sequences of local convex cost and constraint functions over time $t = 1, 2, \ldots$, respectively, where $m_i$ is a positive integer. At each $t$, the network’s objective is to solve the constrained convex optimization problem

$$\min_{x_t \in \mathcal{X}} f_i(x_t)$$

s.t. $g_i(x_t) \leq 0_m, \quad t = 1, 2, \ldots \quad (2)$

where $\mathcal{X} = X_1 \times \cdots \times X_n \subseteq \mathbb{R}^p$ with $p = \sum_{i=1}^n p_i$ being the global decision set, $x_t = \text{col}(x_{1,t}, \ldots, x_{n,t})$ is the global decision variable, $f_i(x_t) = \sum_{i=1}^n f_{i,t}(x_{i,t})$ is the global cost function, and $g_i(x_t) = \sum_{i=1}^n g_{i,t}(x_{i,t})$ is the coupled constraint function. In order to guarantee that problem (2) is feasible, we assume that for any $T \in \mathbb{N}_+$, the set of all feasible sequences $\mathcal{X}_T = \{(x_{1,t}, \ldots, x_{n,t}) : x_i \in X_i, g_i(x_{i,t}) \leq 0_m, t \in [T]\}$ is non-empty. With this standing assumption, an optimal sequence to (2) always exists.

It is interesting and challenging to consider the problem (2) in a bandit online setting and propose distributed algorithms to solve it. For a distributed bandit online algorithm, at time $t$, each agent $i$ selects a decision $x_{i,t} \in \mathcal{X}_i$. After the selection, the agent can sample partial information of
its cost function $f_{i,t}$ and constraint function $g_{i,t}$ at some points. At the same moment, due to the lack of other agents’ cost and constraint function information, the agents exchange data (to be determined in the next section) with their neighbors over a time-varying directed graph. For bandit online algorithms, expected regret and constraint violation are commonly used as performance metrics. The expected regret is the expected accumulation over time of the loss difference between the decision determined by the algorithm and the static optimal decision if the sequences of cost and constraint functions are known in advance. Specifically, the efficacy of a decision sequence $x_T = (x_1, \ldots, x_T)$ is characterized by the expected regret

$$\text{Reg}(x_T) = E\left(\sum_{t=1}^{T} f_i(x_t)\right) - \sum_{t=1}^{T} f_i(x^*_T),$$

where $x^*_T \in \mathbb{X}$ such that $(x^*_T, \ldots, x^*_T) = \arg\min_{x_T \in \mathbb{X}} \sum_{t=1}^{T} f_i(x_t)$ with $x_T = \{x_1, \ldots, x_T\} \in \mathbb{X}$, $g(x) \leq 0$, $t = 1, \ldots, T$ is the set of feasible static sequences. In order to guarantee the existence of $x^*_T$, we assume that for any $T \in \mathbb{N}_+$, $x_T$ is non-empty. Moreover, the constraint violation measure is

$$\|\sum_{t=1}^{T} g_i(x_t)\|_\infty.$$ 

This definition implicitly allows constraint violations at some times to be compensated by strictly feasible decisions at other times. This is appropriate for constraints that have a cumulative nature such as energy budgets enforced through average power constraints.

In this paper, we consider the problem (2) in the bandit setting. Specifically, at each period each agent can sample the value of its local cost function at two points and can observe the value and the exact gradient of its local constraint function. The proposed distributed bandit online algorithm is given in pseudo-code as Algorithm 1. Note that each agent $i$ maintains three local sequences: the local primal decision variable sequence $\{x_{i,t}\} \subseteq \mathbb{X}_i$, the local dual variable sequence $\{\tilde{g}_{i,t}\} \subseteq \mathbb{R}^m$, and the estimator of the average of local dual variables $\{\tilde{g}_{i,t}\} \subseteq \mathbb{R}^m$. They are initialized by an arbitrary $x_{i,1} \in \mathbb{X}_i$ and $\tilde{g}_{i,1} = \tilde{g}_{i,1} = 0$, and updated recursively using the update rules (3a)-(3c). In (3b), $a_{i,t}$ is the updating direction information for the local primal variable defined as

$$a_{i,t} = \nabla_2 f_{i,t-1}(x_{i,t-1}) + \left(\nabla g_{i,t-1}(x_{i,t-1})\right)^\top \tilde{g}_{i,t}. \quad (4)$$

The intuition of the update rules (3a)-(3c) is as follows. The augmented Lagrangian function associated with the constrained optimization problem with cost function $f$ and constraint function $g$ is

$$A(x, \mu) = f(x) + \mu^\top g(x) - \frac{\beta}{2} \|\mu\|^2, \quad (5)$$

where $\mu \in \mathbb{R}^m$ is the Lagrange multiplier and $\beta > 0$ is the regularization parameter. $A(x, \mu)$ is a convex-concave function. A standard primal-dual algorithm to find its saddle point is

$$x_{k+1} = \mathcal{P}_\mathbb{X}(x_k - \alpha(\nabla f(x_k) + (\nabla g(x_k))^\top \mu_k)), \quad (6a)$$

Assumption 3. (Slater’s condition) There exists a constant $\varepsilon > 0$ and a vector $x_s \in \mathbb{X}$ such that $g_i(x_s) \leq -\varepsilon 1_m$, $\forall t \in \mathbb{N}_+$. Assumption 3 is slightly weaker than the Assumption (as3) used in [30] since the later requires $x_s \in (1-\xi)\mathbb{X}$ for a small $\xi > 0$.
Algorithm 1 Distributed Bandit Online Primal-Dual Descent

1: Input: non-increasing stepsize sequences \( \{\alpha_t\} \), \( \{\beta_t\} \), and \( \{\gamma_t\} \subseteq (0, 1] \); non-increasing shrinkage coefficients \( \{\xi_{i,t}\} \subseteq (0, 1) \), \( i \in [n] \); exploration parameters \( \{\delta_{i,t}\} \subseteq (0, r_i \xi_{i,t}) \), \( i \in [n] \).

2: Initialize: \( x_{i,1} \in (1 - \xi_{i,1}) X_i \) and \( q_{i,1} = 0_m \), \( \forall i \in [n] \).

3: for \( t = 2, \ldots, T \) do
4: for \( i \in [n] \) in parallel do
5: Select vector \( u_{i,t-1} \in \mathbb{S}^n \) independently and uniformly at random.
6: Sample \( f_{i,t-1}(x_{i,t-1} + \delta_{i,t-1} u_{i,t-1}) \) and \( f_{i,t-1}(x_{i,t-1}) \) and observe \( g_{i,t-1}(x_{i,t-1}) \) and \( \nabla g_{i,t-1}(x_{i,t-1}) \).
7: Receive \( [W_{i,t}]_{ij} q_{j,t-1} \) from \( j \in N_{i,t}^{\text{out}}(G_t) \).
8: Update
\[
q_{i,t} = \sum_{j=1}^{n} [W_{i,t}]_{ij} g_{j,t-1},
\]
\[
x_{i,t} = \frac{1}{(1 - \xi_{i,t})} x_{i,t-1} - \alpha_t a_{i,t}, \quad q_{i,t} = [\gamma_{i,t} + \gamma_t (g_{i,t-1}(x_{i,t-1}) - \beta_t q_{i,t})]_+, \quad (3a)
\]
9: Broadcast \( q_{i,t} \) to \( N_{i,t}^{\text{out}}(G_t) \).
10: end for
11: end for
12: Output: \( x_T \).

\[
\mu_{k+1} = \mu_k + \gamma (g(x_k) - \mu_k)\big|_+, \quad (6b)
\]

where \( \alpha > 0 \) and \( \gamma > 0 \) are the stepsizes used in the primal and dual updates, respectively. The update rules (3a)–(3c) are the distributed, online, and gradient-free extensions of (6a) and (6b). The term \(-\beta_t q_{i,t}\) in (3c) is derived from penalty term \(-\frac{\gamma}{2} \| x_t \|^2\) in the augmented Lagrangian function (5) and it has an important role to guarantee that the dual variable sequence is not growing too large as shown in the proof of Lemma 1 given in [34].

In Algorithm 1, the data exchanged between agents is the local dual variable rather than any information related to the local primal variable, so our algorithm is well suited to account for privacy requirements.

B. Expected Regret and Constraint Violation Bounds

The following lemma provides the expected regret and constraint violation bounds for the general case.

Lemma 1. Suppose Assumptions 1–2 hold. For any \( T \in \mathbb{N}_+ \), let \( x_T \) be the sequence generated by Algorithm 1. Then,

\[
\mathbb{E}\left[\sum_{t=1}^{T} \| g_t(x_t) \| + \sum_{t=1}^{T} \gamma_{t+1} + n \rho g_T^2 \sum_{t=1}^{T} \alpha_{t+1} + \frac{1}{2} \sum_{t=1}^{T} \sum_{i=1}^{n} \eta_t \mathbb{E}\left[\| q_{i,t} \|^2 \right] + \frac{2 \rho R_{\max}^2}{\alpha_{T+1}} \right],
\]

\[
(7a)
\]

where \( C_1 = \frac{3n(n+1)^2}{1 - \lambda}, \quad \lambda = \frac{(1 - w/2n^2)^{-1}}{1 - \eta}, \quad \eta_t = \frac{1}{\gamma_t} - \frac{1}{\gamma_{t+1}} + 4 G^2 \gamma_t, \quad R_{\max} = \max_{i \in [n]} \{ R_i \}, \quad \alpha_t = \frac{1}{t}, \quad \beta_t = \frac{4 G^2 \gamma_t + 1}{t}, \quad \gamma_t = \frac{1}{t^{1-c}},
\]

\[
(8)
\]

where \( c \in [0.5, 1) \) is a constant. Then,

\[
\mathbb{E}\left[\| \sum_{t=1}^{T} g_t(x_t) \| \right] \leq \frac{2 n \rho R_{\max}^2}{\alpha_{T+1}} + \frac{1}{2} \sum_{t=1}^{T} \sum_{i=1}^{n} \eta_t \mathbb{E}\left[\| q_{i,t} \|^2 \right] + \frac{2 \rho R_{\max}^2}{\alpha_{T+1}},
\]

\[
(9a)
\]
\[
\|\sum_{t=1}^{T} g_t(x_t)\| \leq \sqrt{C_3 T^{1-c/2}},
\]
(9b)

where

\[
C_2 = 4nG_f R_{\text{max}} + C_{2,1}, \quad C_{2,1} = 2n^2 F_g G_g R_{\text{max}} c(4G^2 + 1) + C_1
\]

\[
\]}

Remark 1. The parameter \(c\) in Theorem 1 is a user-defined parameter which enables the trade-off between the expected regret bound and the constraint violation bound. For example, setting \(c = 0.5\) gives \(\text{Reg}(x_T) = \mathcal{O}(\sqrt{T})\) and \(\|\sum_{t=1}^{T} g_t(x_t)\| = \mathcal{O}(T^{3/4})\). These two bounds are the same as the bounds achieved in [8], [9]. So in average sense, Algorithm 1 is as efficient as the algorithms proposed in [8], [9]. However, [8], [9] are in full-information setting and the algorithms proposed in them are centralized. Moreover, the constraint functions considered in [8] are time-invariant.

From (9b), we see that the constraint violation bound is strictly greater than \(\mathcal{O}(\sqrt{T})\) since \(c < 1\). In the following theorem we show that \(\mathcal{O}(\sqrt{T})\) bound on constraint violation can be achieved if Slater’s condition holds.

Theorem 2. Suppose Assumptions 1–3 hold. For any \(T \in \mathbb{N}_+\), let \(x_T\) be the sequence generated by Algorithm 1 with

\[
\begin{align*}
\alpha_t &= \frac{1}{\sqrt{t}}, \quad \beta_t = \frac{4G_f^2 + 1}{\sqrt{t}}, \\
\gamma_t &= \frac{1}{\sqrt{t}}, \\
\xi_{i,t} &= \frac{1}{t + 1}, \quad \delta_{i,t} = \frac{r_i}{t + 1}, \quad \forall t \in \mathbb{N}_+.
\end{align*}
\]

Then,

\[
\text{Reg}(x_T) \leq C_4 \sqrt{T},
\]
(11a)

\[
\|\sum_{t=1}^{T} g_t(x_t)\| \leq 4\sqrt{n} B_1 \sqrt{T},
\]
(11b)

where

\[
C_4 = 4G_f R_{\text{max}} + 2B_1 F_g G_g R_{\text{max}} + 2C_1 + 2np^2 G_f^2 + 8n R_{\text{max}}^2, \\
B_1 = \max \left\{ \frac{2nB_2}{F_g G_g R_{\text{max}}} \sqrt{2} \right\} + 2B_2, \\
B_2 = C_1 + 2p G_f R_{\text{max}} + 2n R_{\text{max}}^2 + \frac{4 C_3}{4G_g^2 + 1}
\]

are constants independent of \(T\).

Remark 2. From (9a) or (11a), we know that, no matter whether Slater’s condition holds or not, \(\mathcal{O}(\sqrt{T})\) expected regret is achieved by Algorithm 1. It was pointed out in [6] that \(\mathcal{O}(\sqrt{T})\) is a tight bound to regret for online convex optimization problems in full-information setting, so in average sense, Algorithm 1 is as efficient as the optimal online algorithms using full-information. The same expected regret bound and the same expected regret as well as constraint violation bounds were also achieved by the two-query bandit algorithm in [25] and [30], respectively. However, in [25] static set constraints (rather than time-varying inequality constraints) were considered and in [30] a slightly stronger Slater’s condition was assumed (see the discussion after Assumption 3 for details). Moreover, in [25], [30] the proposed algorithms are centralized (rather than distributed) and the total number of iterations needs to be known in advance to design the algorithm.

V. NUMERICAL SIMULATIONS

This section evaluates the performance of Algorithm 1 in solving the power generation example introduced in Section III-A. The local cost and constraint functions are denoted

\[
f_{i,t}(x_{i,t}) = x_{i,t}^T \Pi_{i,t} x_{i,t} + \langle \pi_{i,t}, x_{i,t} \rangle,
\]

\[
g_{i,t}(x_{i,t}) = x_{i,t}^T \Phi_{i,t} x_{i,t} + \langle \phi_{i,t}, x_{i,t} \rangle + c_{i,t},
\]

where \(\Pi_{i,t} \in \mathbb{R}^{p_i \times p_i}, \pi_{i,t} \in \mathbb{R}^{p_i}, \Phi_{i,t} \in \mathbb{R}^{p_i \times p_i}, \phi_{i,t} \in \mathbb{R}^{p_i},\) and \(c_{i,t} \in \mathbb{R}\). At each time \(t\), an undirected graph is used as the communication graph. Specifically, connections between vertices are random and the probability of two vertices being connected is \(p > 0\). Moreover, edges \((i, i + 1), i \in [n - 1]\) are added and \([W_i]_{ij} = 1/n\) if \((j, i) \in E_t\) and \([W_i]_{ij} = 1 - \sum_{j \in N_t(x_i)} [W_i]_{ij}\). The parameters are set as: \(n = 50, m = 1, p_i = 6, X_i = [-10, 10]^p,\) and \(p = 0.2\). Each element of \(\Pi_{i,t}, \pi_{i,t}, \Phi_{i,t}, \phi_{i,t},\) and \(c_{i,t}\) are drawn from the discrete uniform distribution in \([-5, 5], [0, 10], [-5, 5], [-5, 5],\) and \([-5, -1]\), respectively. Under above settings, Assumptions 1–2 hold.

Since there are no other distributed bandit online algorithms to solve the problem of online optimization with time-varying coupled inequality constraints, we compare our Algorithm 1 with the centralized two-query bandit algorithm in [30]. Figs. 1 and 2 show the evolutions of \(\text{Reg}(x_T)/T\) and \(\|\sum_{t=1}^{T} g_t(x_t)\|/T\), respectively. The average is taken over 100 realizations. From Figs. 1 and 2, we see that our proposed distributed algorithm achieves comparable results as the centralized algorithm proposed in [30].

VI. CONCLUSIONS

In this paper, we considered a distributed bandit online convex optimization problem with time-varying coupled inequality constraints. We proposed a distributed bandit online...
primal-dual algorithm to solve this problem, where a two-query procedure was used to estimate the gradients of the cost functions. We showed that sublinear expected regret and constraint violation can be achieved. We also showed that the results in this paper can be cast as non-trivial extensions of existing literature on online optimization and bandit feedback. An interesting future research directions is to consider an adaptive choice of the number of queries per period by different agents.

REFERENCES


