International Journal of
Network Dynamics and Intelligence (IJNDI)

Article

Distributed Online Convex Optimization with Long-Term Constraints: No Cumulative Constraint Violation

Difeng Zhang [1], and Shaofu Yang [2,*]

[1] School of Mathematics, Southeast University, Nanjing 211189, China
[2] School of Computer Science and Engineering, Southeast University, Nanjing 211189, China
* Correspondence: sfyang@seu.edu.cn
Received: 13 September 2025; Revised: 3 November 2025; Accepted: 22 December 2025; Published: 11 March 2026

Abstract: This paper investigates the problem of distributed online optimization with long-term constraints (LTC), where constraints are not required to be satisfied at each time step but only over a longer horizon. We first propose a baseline algorithm based on online regularized Lagrangian functions, achieving a regret bound of $\mathcal{O}(T^{3/4})$. By refining the Lagrangian functions and optimization strategies, we further improve the regret bound to $\mathcal{O}(T^{2/3})$. Notably, our results enforce strict satisfaction of LTC over time, in contrast to most existing works that focus solely on achieving sublinear cumulative constraint violations. Moreover, the obtained performance matches the optimal guarantees achievable in the centralized setting. Finally, we demonstrate the effectiveness of our approach through a distributed online regularized linear regression problem.

Keywords: distributed optimization; online optimization; long-term constraints

1. Introduction

Online optimization and learning have recently emerged as popular research frameworks in optimization and machine learning, with applications in areas such as online display advertising [1], online allocation [2,3], and dictionary learning [4]. In essence, online optimization can be viewed as a repeated game between the learner and the environment. At each time step or iteration t, the learner selects a decision $\mathbf{x}_t$ from a given decision set or constraint set $\mathcal{X}$ and subsequently receives information about the loss function f_t after submitting the decision. The primary objective of online optimization is to minimize the cumulative loss function over time. A commonly used metric to evaluate algorithm performance is regret, that is

$$\mathrm{Reg}(T) = \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{u}_t), \tag{1}$$

which measures the difference between the cumulative loss incurred by the algorithm and that of an optimal baseline sequence $\mathbf{u}_t$. Depending on the choice of baseline, the regret can be classified as static regret or dynamic regret. In this paper, we focus on static regret, where the baseline $\mathbf{u}_t$ adopts the optimal static decision $\mathbf{x}^\star = \mathrm{argmin}_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} \sum_{i=1}^{N} f_{i,t}(\mathbf{x})$.

Significant progress has been made in the development of online optimization algorithms [5–9]. For online optimization problems, proposed algorithms typically employ a projection step to ensure constraint satisfaction by projecting the decision variable onto the feasible set. For instance, the online gradient descent (OGD) method introduced in [5] achieves an upper regret bound of $\mathcal{O}(\sqrt{T})$, where T denotes the time horizon. At each time step t, the learner or agent updates the decision variable through the following projection step:

$$\mathbf{x}_{t+1} = \mathcal{P}_{\mathcal{X}}\left(\mathbf{x}_t - \eta \nabla f_t(\mathbf{x}_t)\right), \tag{2}$$

where $\eta > 0$ is the step size, and $\mathcal{P}_{\mathcal{X}}(\cdot)$ means the projection onto the constraint set $\mathcal{X}$. However, when the constraint set $\mathcal{X}$ is general or complex, computing the projection step becomes equivalent to solving a challenging sub-optimization problem, leading to substantial computational burden and complexity. To address this issue, Ref. [10] proposed a simpler linear optimization step to replace the projection step, the idea is to linearize the cost function to simplify the projection step, but this leads to $\mathcal{O}(T^{3/4})$ regret whereas the optimal regret is $\mathcal{O}(\sqrt{T})$.

To mitigate the computational overhead imposed by the projection step, the concept of *long-term constraints* (LTC), first proposed in [11], has been widely adopted and investigated in the field of online optimization. Unlike conventional methods that require the constraint to be satisfied at each time step, LTC only demands that the constraint be satisfied over a long period. In other words, temporary constraint violations are allowed, but it is necessary to ensure that the overall *cumulative constraint violation* (CCV) which is defined as $\left[\sum_{t=1}^{T} g_s(\mathbf{x}_t)\right]_+$ grows at a sublinear rate with respect to the time horizon T. An online optimization problem with LTC can be formulated as follows: the agent needs to determine a sequence of decisions $\{\mathbf{x}_t\}_{t=1}^{T}$ to minimize the regret defined in (1) while ensuring that the average decision satisfies the constraint set, i.e., $\sum_{t=1}^{T} \mathbf{x}_t/T \in \mathcal{X}$. If the feasible set $\mathcal{X}$ is characterized by a set of inequality constraints, i.e., $\mathcal{X} := \{\mathbf{x} \in \mathbf{R}^d \mid g_s(\mathbf{x}) \leq 0,\ s = 1, \cdots, q\} \subseteq \mathbf{R}^d$, then LTC requires that $\sum_{t=1}^{T} g_s(\mathbf{x}_t) \leq 0,\ \forall s = 1, \ldots, q$. This also implies that the CCV should grow sublinearly [11,12]. This concept can be intuitively understood through simple mathematical derivation. In convex optimization problems, where the constraint functions $\{g_s(\mathbf{x})\}_{s=1}^{q}$ are convex, the following inequality holds:

$$g_s\left(\frac{\sum_{t=1}^{T} \mathbf{x}_t}{T}\right) \leq \sum_{t=1}^{T} g_s\left(\frac{\mathbf{x}_t}{T}\right) \leq \frac{\left[\sum_{t=1}^{T} g_s(\mathbf{x}_t)\right]_+}{T}. \tag{3}$$

The sublinear growth of $\left[\sum_{t=1}^{T} g_s(\mathbf{x}_t)\right]_+$ implies that $\lim_{T\to\infty} \frac{\left[\sum_{t=1}^{T} g_s(\mathbf{x}_t)\right]_+}{T} \to 0$, which leads to $g_s\left(\frac{\sum_{t=1}^{T} \mathbf{x}_t}{T}\right) \leq 0$, indicating that $\frac{\sum_{t=1}^{T} \mathbf{x}_t}{T} \in \mathcal{X}$. This problem has been extensively studied in centralized scenarios [9, 11–15]. The authors in [11] proposed an online primal-dual algorithm based on the online gradient descent method, achieving a regret of $\mathcal{O}(\sqrt{T})$ and a cumulative constraint violation of $\mathcal{O}(T^{3/4})$. In [9], the authors considered the strongly convex scenario and obtained a regret of $\mathcal{O}(\log T)$ and a CCV of $\mathcal{O}(\sqrt{T \log T})$. Moreover, Ref. [11] also investigated scenarios where the LTC is strictly satisfied, meaning that there is no cumulative violation: $\sum_{t=1}^{T} g_s(\mathbf{x}_t) \leq 0,\ \forall s = 1, \ldots, q$ and achieved a regret bound of $\mathcal{O}(T^{3/4})$. Jenatton et al. [13] improved the regret bound to $\mathcal{O}(T^{2/3})$ by employing adaptive algorithms. Additionally, in [14], a novel algorithm using the drift-plus-penalty technique was proposed, achieving a CCV of $\mathcal{O}(1)$ while maintaining a regret of $\mathcal{O}(\sqrt{T})$. Furthermore, the work in [15] extended these results by considering adversarial constraints where the constraint is unknown in advance and could be arbitrarily and adversarially chosen, providing a broader perspective on the problem.

The online optimization problem with LTC can also be extended to the distributed framework. In the distributed online optimization (DOO) problem, at each time step t, the agents in the network submit their decision variables and receive feedback regarding their own local loss functions. Subsequently, they communicate with neighboring agents in the network to collaboratively address the global optimization problem. The primary objective of DOO is to minimize the cumulative loss across the entire network. In the case of DOO with LTC, the performance metrics of DOO algorithms include both regret and cumulative constraint violation. Similar to the centralized scenario, these metrics defined as:

$$\mathrm{Reg}(i, T) = \sum_{t=1}^{T}\sum_{i=1}^{N} f_{i,t}(\mathbf{x}_i(t)) - \sum_{t=1}^{T}\sum_{i=1}^{N} f_{i,t}(\mathbf{x}^\star), \tag{4}$$

$$\mathrm{CCV}(T) = \left[\sum_{t=1}^{T}\sum_{i=1}^{N} g_s(\mathbf{x}_i(t))\right]_+, \tag{5}$$

play a crucial role in evaluating the efficiency and robustness of the algorithms, where N is the number of the agents in the network and $\mathbf{x}^\star = \mathrm{argmin}_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T}\sum_{i=1}^{N} f_{i,t}(\mathbf{x})$ is the static optimal baseline. Achieving sublinear upper bounds for both regret and cumulative constraint violation is also a key goal, and researchers have actively worked to address this problem [16–19]. In [16], the authors extended the work in [11] to the distributed framework, introducing a stricter metric for constraint violation called cumulative absolute constraint violation (CACV), which is defined as $\sum_{t=1}^{T}\sum_{i=1}^{N} [g_s(\mathbf{x}_i(t))]_+$. They proposed a fully distributed algorithm where agents can independently update their own dual variables, and achieve a regret of $\mathcal{O}(\sqrt{T})$ and a CACV of $\mathcal{O}(T^{3/4})$, which matches the results found in the centralized framework [11,13]. In [17], the authors considered time-varying constraints and obtained regret and constraint violation bounds similar to those in [16].

From the above discussion, it is evident that there remains significant room for exploration in the DOO problem with LTC. Specifically, much of the existing research focuses on minimizing regret or exploring stricter metrics of constraint violation. However, there has been limited attention to scenarios where LTC is strictly satisfied, which means $\mathrm{CCV}(T) \leq 0$ in finite time instead of $\lim_{T\to\infty} \mathrm{CCV}(T) \to 0$. While the lower bound of regret has been extensively studied in the context of general online optimization [20], the lower bound of regret when LTC is strictly satisfied remains an unsolved problem. Motivated by this gap, we consider this issue and explore the regret upper bound in the current paper.

Indeed, many practical applications can be modeled as online optimization problems with long-term constraints. We provide a simple motivating example of online energy budgeting [21] to illustrate this.

$$\min \sum_{t=1}^{T} f_t(\mathbf{m}(t), \mathbf{x}(t), \boldsymbol{\lambda}(t)), \tag{6a}$$

$$s.t. \sum_{t=1}^{T} g(\mathbf{u}(t), \mathbf{m}(t)) \leq Z. \tag{6b}$$

where $\mathbf{m}$ is a column vector of the number of servers, $\mathbf{x}$ is the virtual machine resource allocation vector, $\boldsymbol{\lambda}$ is the workload vector of virtual machine, and $\mathbf{u}$ is the CPU utilization vector. The objective is to minimize the data center operational cost (6a), while satisfying the long term energy consumption target (6b). Note that the cost functions associated with electricity prices may change over time and could not be known in advance—this makes the online energy budgeting problem (6) is an online optimization problem. Furthermore, the long-term constraints (6b) must be satisfied strictly because the energy budget cap Z is limited. More details could be found in [21].

The following are our contributions.

(1) To the best of the authors' knowledge, this paper is the first to consider the regret when LTC is strictly satisfied in distributed online optimization, and we design gradient-based algorithms with fixed step size and adaptive step size, respectively, both of which ensure that the LTC is strictly satisfied and achieve sublinear regret upper bound.

(2) For the algorithms we designed in this paper, we present theorems demonstrating that LTC is strictly satisfied and achieve $\mathcal{O}(T^{3/4})$ regret under fixed step size and $\mathcal{O}(T^{2/3})$ regret under adaptive step size, which is consistent with the results in the centralized scenario [11,13].

The rest of the paper is organized as follows. Section 2 describes the problem setting and related assumptions, Section 3 proposes the algorithms and provides analysis, Section 4 is the numerical simulation, and Section 5 is the conclusion.

Notations: Let $\mathbf{R}^d$ represent d-dimensional Euclidean space. $\mathbf{S}^d$ means the unit sphere in $\mathbf{R}^d$. $\|\cdot\|$ denote the Euclidean norm and $[\mathbf{x}]_i$ denote the i-th entry of vector $\mathbf{x}$. Let $\mathbf{R}^q_+$ represent the nonnegative orthant in $\mathbf{R}^q$, i.e., $\mathbf{x}^q_+ = \{\mathbf{x} \in \mathbf{R}^q \mid [\mathbf{x}]_i \geq 0, i = 1, \ldots, q\}$. Denote $\mathcal{P}_{\mathcal{X}}(\mathbf{x})$ as the Euclidean projection of a vector $\mathbf{x}$ onto a convex set $\mathcal{X}$. Let $[\mathbf{A}]_{ij}$ denote the $(i,j)th$ entry of matrix $\mathbf{A}$. For a convex function f, let $\nabla f(\mathbf{x})$ denotes the gradient at the point $\mathbf{x}$.

2. Problem Formulation

2.1. Graph Theory

Consider a time-varying digraph sequence $\{\mathcal{G}_t\} = \{(\mathcal{N}, \mathcal{E}_t)\}$ with node set $\mathcal{N} = \{1, \ldots, N\}$ and edge set $\mathcal{E}_t \subset \mathcal{N} \times \mathcal{N}$ at time t. Agent i is able to transmit information to agent j at time t iff $(j, i) \in \mathcal{E}_t$. $\mathbf{A}(t) \in \mathbf{R}^{N\times N}$ is the weight matrix associated with $\mathcal{G}_t$.

As for time-varying communication graph, the following assumptions are commonly required (e.g., in [12,16,22]).

Assumption 1. *$\forall t \geq 1$, the weight matrix $\mathbf{A}(t)$ of $\mathcal{G}_t$ satisfied:*

(1) *$\mathbf{A}(t)$ is doubly stochastic, i.e., $\sum_{i=1}^{N}[\mathbf{A}(t)]_{ij} = \sum_{j=1}^{N}[\mathbf{A}(t)]_{ij} = 1$, $\forall i, j \in \mathcal{N}$.*

(2) *there exists a scalar $\xi > 0$ such that $[\mathbf{A}(t)]_{ii} \geq \xi$, $i = 1, \ldots, N$, and $[\mathbf{A}(t)]_{ij} \geq \xi$ if $(i, j) \in \mathcal{E}_t$ and $[\mathbf{A}(t)]_{ij} = 0$, otherwise.*

(3) *$\{\mathcal{G}_t\}$ is B-strongly connected. That is there exists an integer $B \geq 1$ such that the union graph $(\mathcal{N}, \cup_{t=0}^{B-1}\mathcal{E}_{k+t})$ is strongly connected for all $k \in \{0, 1, 2, \cdots\}$.*

2.2. DOO with LTC

We consider distributed online convex optimization problem over a time-varying network $\{\mathcal{G}_t\} = \{(\mathcal{N}, \mathcal{E}_t)\}$.

$$\min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T}\sum_{i=1}^{N} f_{i,t}(\mathbf{x}), \tag{7}$$

where $\{f_{i,t}\}_{t=1}^{T}$ is local loss functions of agents and $\mathcal{X} := \{\mathbf{x} \in \mathbf{R}^d \mid g_s(\mathbf{x}) \leq 0,\ s = 1, \cdots, q\} \subseteq \mathbf{R}^d$ is the global constraints set. We use regret (4) and LTC: $\sum_{t=1}^{T}\sum_{i=1}^{N} g_s(\mathbf{x})$ for algorithm performance metrics.

It should be noted that the objective is to minimize the regret bound when LTC is strictly satisfied, $\sum_{t=1}^{T}\sum_{i=1}^{N} g_s(\mathbf{x}_i(t)) \leq 0$. Other existing researches on DOO-LTC do not strictly require that LTC be satisfied, only that CCV have a sublinear upper bound. From (3) we can see the LTC will be satisfied when $T \to \infty$ if we only require CCV have a sublinear upper bound. So if the constraint function is fixed and known, then it is meaningful to investigate whether and how we can ensure that LTC is strictly satisfied.

Below we give the standard assumptions about the loss and the constraint functions.

Assumption 2. *There exists a ball* $\mathcal{B} := \{\mathbf{x} \in \mathbf{R}^d \mid \|\mathbf{x}\| \leq R_{\mathcal{X}}\}$ *with* $0 < R_{\mathcal{X}} < \infty$ *such that* $\mathcal{X} \subseteq \mathcal{B}$.

Assumption 3. $f_{i,s}$ *and* g_l *are differentiable, convex, and the gradient is bounded within the ball* $\mathcal{B}$. *That is,* $\exists G > 0$, *s.t.* $\|\nabla f_{i,t}(\mathbf{x})\| \leq G, \|\nabla g_s(\mathbf{x})\| \leq G, \forall \mathbf{x} \in \mathcal{B}$.

Define $g(\mathbf{x}) = \max_{s\in\{1,\ldots,q\}} g_s(\mathbf{x})$. Next we present an assumption that will be used to ensure that LTC is strictly satisfied.

Assumption 4. *Define the convex set* $\tilde{\mathcal{X}} := \{\mathbf{x} \in \mathbf{R}^d \mid g(\mathbf{x}) + \gamma \leq 0\} \subseteq \mathbf{R}^d$ *as the subset of* $\mathcal{X}$ *where* $\gamma \geq 0$. *Assume that there exist a constant* $\sigma > 0$ *such that*

$$\min_{\mathbf{x}\in\mathbf{R}^d: g(\mathbf{x})+\gamma=0} \|\nabla g(\mathbf{x})\| \geq \sigma.$$

Assumption 4 is used in centralized framework which study the online optimization with LTC [11,13]. The objective of this assumption is to controls the gap between the optimal values over two domains $\tilde{\mathcal{X}}$ and $\mathcal{X}$. In specific, let $\mathbf{x}^\star$ and $\mathbf{x}_\gamma$ be the optimal decision variables of $\min_{g(\mathbf{x})\leq 0} \sum_{t=1}^T \sum_{i=1}^N f_{i,t}(\mathbf{x})$ and $\min_{g(\mathbf{x})+\gamma\leq 0} \sum_{t=1}^T \sum_{i=1}^N f_{i,t}(\mathbf{x})$, respectively, then we will get a upper bound of the gap.

Lemma 1. *If Assumption 4 is satisfied, we have*

$$\left|\sum_{t=1}^{T}\sum_{i=1}^{N} [f_{i,t}(\mathbf{x}^\star) - f_{i,t}(\mathbf{x}_\gamma)]\right| \leq \frac{GNT}{\sigma}\gamma \tag{8}$$

Proof. We can easily obtain (8) by using Theorem 7 in [11]. □

Lemma 1 indicates that we can appropriately narrow the boundary of the constraint set, and the error or change of the optimal values could be bounded by some parameters. And this will be manifested as a reduction in the frequency and degree of constraint violations when the algorithm is executed.

3. Main Results

In this section, we design a DOO algorithm for problem (7). First, we replace $\{g_s(\mathbf{x})\}_{s=1}^q$ with $g(\mathbf{x})$ to simplify the problem, if we guarantee no constraint violation for $g(\mathbf{x}) := \max\limits_{s\in\{1,\ldots,q\}} g_s(\mathbf{x})$, then obviously LTC can be satisfied for all g_s, $s = 1, \ldots, q$. Then we can know the prime optimization problem (7) is equivalent to the prime-dual problem

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\lambda\in\mathbf{R}_+} \sum_{t=1}^{T}\sum_{i=1}^{N} f_{i,t}(\mathbf{x}) + \lambda g(\mathbf{x}), \tag{9}$$

where λ is the dual variable.

Next, we introduce the concept of *online regularized Lagrangian function* from [11], which is also used in [13,16].

$$\mathcal{L}_{i,t}(\mathbf{x}, \lambda) := f_{i,t}(\mathbf{x}) + \lambda(g(\mathbf{x}) + \gamma) - \frac{\delta\eta}{2}\lambda^2, \tag{10}$$

where η is the step size, and $\gamma > 0$ and $\delta > 0$ will be determined later.

Optimizing the Lagrangian function could be equivalent to solving the following penalty function problem.

$$\min_{\mathbf{x}\in\mathcal{X}} \max_{\lambda\in\mathbf{R}_+} \mathcal{L}_{i,t} = \min_{\mathbf{x}\in\mathcal{X}} f_{i,t}(\mathbf{x}) + \frac{1}{2\delta\eta}\left[g(\mathbf{x}) + \gamma\right]^2, \tag{11}$$

where $\frac{1}{2\delta\eta}$ could also be called the penalty coefficient. The equality is obtained by first maximizing $\mathcal{L}_{i,t}$ with respect to the dual variable λ. Noting $\mathcal{L}_{i,t}$ is a quadratic function with respect to λ, so we can easily obtain the equality in (10).

We can see from (11) that in fact the constraint is $g(\mathbf{x}) + \gamma \leq 0$. Our main idea is to use a stricter constraint in the algorithm so that LTC will be strictly satisfied, although this stricter constraint may often be violated. It should be noted that a regularization term $\frac{\delta\eta}{2}\lambda^2$ is added to (10), the regularization term can bound the dual variable λ to prevent the gradient of $\mathcal{L}_{i,t}(\mathbf{x}, \lambda)$ from being too large with respect to $\mathbf{x}$, and thus affect the stability of the solution. In addition, the addition of a regularization term can make $\mathcal{L}_{i,t}$ strongly concave with respect to λ, further enable us to directly update the dual variable explicitly, which can be seen in the later algorithm. And finally, strong concavity will help us improve the regret upper bound in the Section 3.2.

In Algorithm 1, agents update the decision variable and dual variable through by gradient descent and gradient ascent, respectively. In (14), we project the intermediate variable $\mathbf{P}_i(t)$ into a Euclidean ball $\mathcal{B}$ instead of into the constraint set $\mathcal{X}$. It should be noted that although the projection notation is still used here, this step can be calculated by simple scaling and is not complicated. Since we are considering LTC, we do not need to strictly satisfy the constraints at each time step, so (14) is feasible.

Algorithm 1 LTC-DOO with Fixed Step Size

Require: Total time horizon T, parameters $\eta, \delta \in \mathbf{R}$, $\mathbf{x}_i(1) \in \mathbf{R}^d$, $\lambda_i(1) \in \mathbf{R}_+$

for $t = 1, \ldots, T$ **do**

for $i = 1, \ldots, N$ **do**

1. Agent i submits $\mathbf{x}_i(t)$, then could receives the loss function $f_{i,t}$.
2. Agent i does a gradient descent

$$\mathbf{y}_i(t) = \mathbf{x}_i(t) - \eta\nabla_{\mathbf{x}}\mathcal{L}_{i,t}(\mathbf{x}_i(t), \lambda_i(t)). \tag{12}$$

3. Consensus protocol

$$\mathbf{P}_i(t) = \sum_{j=1}^{N}[\mathbf{A}(t)]_{ij}\mathbf{y}_j(t). \tag{13}$$

4. Agent i updates the decision variable as

$$\mathbf{x}_i(t+1) = \mathcal{P}_{\mathcal{B}}(\mathbf{P}_i(t)) \tag{14}$$

5. Agent i updates the dual variable as

$$\lambda_i(t+1) = \lambda_i(t) + \eta\nabla_{\lambda}\mathcal{L}_{i,t}(\mathbf{x}_i(t), \lambda_i(t)) \tag{15}$$

end for

end for

3.1. $\mathcal{O}(T^{3/4})$ Regret with no Constraint Violation

In this subsection, we show that using Algorithm 1 to solve problem (7), we can obtain $\mathcal{O}(T^{3/4})$ regret bound while strictly satisfying LTC. This is a preliminary result that matches the centralized case [11]. In the next subsection, we will further improve our results.

Similar to the analysis in [11,13], we first present some lemmas and analysis before giving the conclusion. For the sake of analysis, we make the following definition.

$$\begin{aligned} F &= \max_{t\in\{1,\ldots,T\}} \max_{i\in\{1,\ldots,N\}} \max_{\mathbf{x},\mathbf{y}\in\mathcal{X}} f_{i,t}(\mathbf{x}) - f_{i,t}(\mathbf{y}) \leq 2GR, \\ D &= \max_{s\in\{1,\ldots,q\}} \max_{\mathbf{x}\in\mathcal{B}} g_s(\mathbf{x}) \leq 2GR \end{aligned}$$

Lemma 2. *Suppose that Assumptions 1–3 hold, then we have the following upper bound associated with $\mathcal{L}_{i,t}$,*

$$\begin{aligned} \sum_{i=1}^{N}[\mathcal{L}_{i,t}(\mathbf{x}_i(t), \lambda) - \mathcal{L}_{i,t}(\mathbf{x}, \lambda_i(t))] &\leq N\eta\left(G^2 + \frac{3D^2}{2} + \frac{3\gamma^2}{2}\right) + \sum_{i=1}^{N}\left(\eta G^2 + \frac{3\eta^3\delta^2}{2}\right)\lambda_i^2(t) \\ &+ \frac{\sum_{i=1}^{N}\left[\|\lambda_i(t)-\lambda\|^2 - \|\lambda_i(t+1)-\lambda\|^2\right]}{2\eta} + \frac{\sum_{i=1}^{N}\left[\|\mathbf{x}_i(t)-\mathbf{x}\|^2 - \|\mathbf{x}_i(t+1)-\mathbf{x}\|^2\right]}{2\eta}, \end{aligned} \tag{16}$$

Lemma 3. *Suppose that Assumptions 1–3 hold, then we have the following error upper bound caused by the disagreement between agents,*

$$\sum_{t=1}^{T}\sum_{i=1}^{N}\|\mathbf{x}_i(t)-\mathbf{x}_j(t)\| \le 4NG^2 f_\xi \eta \sum_{t=1}^{T}\left(N+\sum_{i=1}^{N}\lambda_i^2(t)\right), \tag{17}$$

where $f_\xi = \frac{3N}{\psi^{2+1/B}(1-\psi^{1/B})}+4,\ \psi = 1-\frac{\xi}{4N^2}$.

Proof. Similar to ([16], [Lemma2]), we can directly obtain the results. □

In the following theorem, we present $\mathcal{O}(T^{3/4})$ regret with zero CCV.

Theorem 1. *Suppose that Assumptions 1–4 hold, let* $\delta = 4(4NG^3 f_\xi + G^2)$, $a = R\sqrt{G^2+3D^2+3b^2+8NG^3 f_\xi}$, $\eta = R^2/[a\sqrt{T}]$, $\gamma = bT^{-1/4}$, $b = 2\sqrt{F(a/R^2+\delta R^2/a)}$, *and* $\{\mathbf{x}_i(t)\}$ *is generated by Algorithm 1. When T is sufficiently large enough that* $FT \ge a\sqrt{T}$, *we have LTC* $\sum_{t=1}^{T}\sum_{i=1}^{N} g_s(\mathbf{x}_i(t)) \le 0$ *and the regret bound is*

$$\mathrm{Reg}(i,T) \le aN\sqrt{T} + \frac{GNT}{\sigma}\gamma \le \mathcal{O}(T^{3/4}).$$

Proof. Starting from the definition of regret, by adding $f_{j,t}(\mathbf{x}_j(t))$ and subtracting it, we can obtain

$$\begin{aligned}
\mathrm{Reg}(i,T) &= \sum_{t=1}^{T}\sum_{j=1}^{N} f_{j,t}(\mathbf{x}_i(t)) - \sum_{t=1}^{T}\sum_{j=1}^{N} f_{j,t}(\mathbf{x}^\star) = \sum_{t=1}^{T}\left(\sum_{j=1}^{N} f_{j,t}(\mathbf{x}_i(t)) - \sum_{j=1}^{N} f_{j,t}(\mathbf{x}^\star)\right)\\
&= \sum_{t=1}^{T}\sum_{j=1}^{N}(f_{j,t}(\mathbf{x}_i(t)) - f_{j,t}(\mathbf{x}_j(t)) + f_{j,t}(\mathbf{x}_j(t)) - f_{j,t}(\mathbf{x}^\star))\\
&\le \sum_{t=1}^{T}\sum_{j=1}^{N} G\|\mathbf{x}_i(t)-\mathbf{x}_j(t)\| + \sum_{t=1}^{T}\sum_{j=1}^{N}\Big[f_{j,t}(\mathbf{x}_j(t)) - f_{j,t}(\mathbf{x}^\star)\Big],
\end{aligned} \tag{18}$$

where the inequality is based on the G-Lipschitz continuity of $f_{i,t}$.

Substituting $\mathbf{x} = \mathbf{x}_\gamma$ into Lemma 2, and use the definition of $\mathcal{L}_{i,t}$ (10). We could derivate the following inequality,

$$\begin{aligned}
&\sum_{t=1}^{T}\sum_{i=1}^{N}\Big[f_{i,t}(\mathbf{x}_i(t)) - f_{i,t}(\mathbf{x}_\gamma)\Big] + \lambda\sum_{t=1}^{T}\sum_{i=1}^{N}(g(\mathbf{x}_i(t))+\gamma) - \frac{N\lambda^2}{2\eta} - \frac{NT\delta\eta}{2}\lambda^2\\
&\le \frac{NR^2}{2\eta} + NT\eta\left(G^2+\frac{3D^2}{2}+\frac{3\gamma^2}{2}\right) + \left(\frac{3\eta^3}{2}\delta^2 - \frac{\eta}{2}\delta + G^2\eta\right)\sum_{t=1}^{T}\sum_{i=1}^{N}\lambda_i^2(t).
\end{aligned} \tag{19}$$

By noting $\lambda \in \mathbf{R}_+$, we have

$$\lambda\sum_{t=1}^{T}\sum_{i=1}^{N}(g(\mathbf{x}_i(t))+\gamma) - \frac{N\lambda^2}{2\eta} - \frac{NT\delta\eta}{2}\lambda^2 \le \frac{\Big[\sum_{t=1}^{T}\sum_{i=1}^{N}(g(\mathbf{x}_i(t))+\gamma)\Big]^2}{2\left(N/\eta + NT\delta\eta\right)}. \tag{20}$$

Substituting (17), (19), and (20) into (18), and using (8), we could obtain an upper bound associated with regret and CCV,

$$\begin{aligned}
\mathrm{Reg}(i,T) - \frac{GNT}{\sigma}\gamma + \frac{\Big[\sum_{t=1}^{T}\sum_{i=1}^{N}(g(\mathbf{x}_i(t))+\gamma)\Big]^2}{2\left(N/\eta + NT\delta\eta\right)} \le & 4N^2G^3 f_\xi T\eta + \frac{NR^2}{2\eta} + NT\eta\left(G^2+\frac{3D^2}{2}+\frac{3\gamma^2}{2}\right)\\
&+\left(\frac{3\eta^2}{2}\delta^2 - \frac{\delta}{2} + G^2 + 4NG^3 f_\xi\right)\eta\sum_{t=1}^{T}\sum_{i=1}^{N}\lambda_i^2(t),
\end{aligned} \tag{21}$$

where the last term on the right side could be negative by choosing a proper δ.

Noting that the third term on the left side is nonnegative, we can directly obtain that

$$\mathrm{Reg}(i,T) \le \mathcal{O}\left(T\eta + \frac{1}{\eta} + T\eta\gamma^2 + T\gamma\right) \le \mathcal{O}\left(T^{3/4}\right),$$

where $\eta = \mathcal{O}\left(T^{-1/2}\right)$ and $\gamma = \mathcal{O}\left(T^{-1/2}\right)$.

By noting the first term and second term on the left side of (21) is

$$\operatorname{Reg}(i,T)-\frac{GNT}{\sigma}\gamma=\sum_{t=1}^{T}\sum_{j=1}^{N}\Big[f_{j,t}(\mathbf{x}_i(t))-f_{j,t}(\mathbf{x}_\gamma)\Big]\geq -FNT, \tag{22}$$

we can easily validate that the LTC is strictly satisfied from (21)

$$\operatorname{CCV}(T)\leq N\sqrt{2\left(\frac{1}{\eta}+T\delta\eta\right)\left(a\sqrt{T}+\frac{GT}{\sigma}\gamma\right)}-NT\gamma\leq\mathcal{O}\left(T^{3/4}\right)-NT\gamma,$$

by choosing proper $\gamma=\mathcal{O}\left(T^{-1/4}\right)$, the right side could be negative.

$\square$

Remark 1. *From Theorem 1, we can see that if we set $\gamma=0$, then we can obtain $\mathcal{O}(\sqrt{T})$ regret and $\mathcal{O}(T^{3/4})$ CCV which match with the centralized framework [11, 13] and distributed framework [16]. This shows that there is a trade-off between regret and CCV, and it can also be seen from the analysis process.*

3.2. $\mathcal{O}(T^{2/3})$ Regret with no Constraint Violation

In this subsection, we improve Algorithm 1 inspired by [13]. The main idea is to note that the Lagrangian function (10) is strongly concave with respect to λ, and that the regular term coefficients in (10) could be dynamically adaptive rather than fixed.

Similar to (10), we first construct a modified Lagrangian function by using a time-varying parameter θ_t to replace $\delta\eta$.

$$\tilde{\mathcal{L}}_{i,t}(\mathbf{x},\lambda):=f_{i,t}(\mathbf{x})+\lambda(g(\mathbf{x})+\gamma)-\frac{\theta_t}{2}\lambda^2, \tag{23}$$

And similar to (11), we have

$$\min_{\mathbf{x}\in\mathcal{X}}\max_{\lambda\in\mathbf{R}_+}\tilde{\mathcal{L}}_{i,t}=\min_{\mathbf{x}\in\mathcal{X}}f_{i,t}(\mathbf{x})+\frac{1}{2\theta_t}\left[g(\mathbf{x})+\gamma\right]^2. \tag{24}$$

We can observe that solving problem (24) is equivalent to finding $\mathbf{x}_\gamma$ if θ_t is appropriate. Different with (11) and the other general penalty functions, the regularization/penalty coefficient θ_t in (24) is time-varying and its effect can be felt during the later analysis process.

Next, we also need to modify the Algorithm 1, we replace the fixed step size η in (12) and (15) with the time varying step size η_t and μ_t, respectively. That is (25) and (26).

With the above modification, we can use Algorithm 2 for $\tilde{\mathcal{L}}_{i,t}$ and get $\mathcal{O}(T^{2/3})$ regret with zero CCV.

Algorithm 2 LTC-DOO with Adaptive Step Size

Require: Total time horizon T, parameters $\{\eta_t,\mu_t,\theta_t\}_{t=1}^T$, $\mathbf{x}_i(1)\in\mathbf{R}^d$, $\lambda_i(1)\in\mathbf{R}_+$
for $t=1,\dots,T$ **do**
 for $i=1,\dots,N$ **do**
  1. Agent i submits $\mathbf{x}_i(t)$, then could receives $f_{i,t}$.
  2. Agent i does a gradient descent

$$\mathbf{y}_i(t)=\mathbf{x}_i(t)-\eta_t\nabla_{\mathbf{x}}\tilde{\mathcal{L}}_{i,t}(\mathbf{x}_i(t),\lambda_i(t)) \tag{25}$$

  3. Consensus protocol (13)
  4. Agent i updates the decision variable as (14)
  5. Agent i updates the dual variable as

$$\lambda_i(t+1)=\lambda_i(t)+\mu_t\nabla_\lambda\tilde{\mathcal{L}}_{i,t}(\mathbf{x}_i(t),\lambda_i(t)) \tag{26}$$

 end for
end for

Theorem 2. *Suppose that Assumptions 1–4 hold, let* $\theta_t = \frac{1}{t^\beta}$, $\eta_t = \frac{1}{t^\beta}$, $\mu_t = \frac{1}{\theta_t(t+1)}$, $\beta \in (0,1)$, *and* $\{\mathbf{x}_i(t)\}$ *is generated by Algorithm 2. There exist* c_0 *and* c_1 *such that setting* $\gamma = c_1(T^{-\beta/2})$, *we have LTC* $:= \sum_{t=1}^{T}\sum_{i=1}^{N} g_s(\mathbf{x}_i(t)) \le 0$ *for any* $T \ge c_0$ *and the regret bound is*

$$\mathrm{Reg}(i,T) \le \mathcal{O}(T^{\max\{\beta, 1-\beta/2\}}).$$

We can obtain $\mathcal{O}(T^{2/3})$ *regret when choosing* $\beta = 2/3$.

Proof. We first deal with the first term on the right side of (18). Similar to (3), we can directly obtain

$$\sum_{t=1}^{T}\sum_{i=1}^{N}\|\mathbf{x}_i(t)-\mathbf{x}_j(t)\| \le 4NG^2 f_\xi\left(N\sum_{t=1}^{T}\eta_t + \sum_{t=1}^{T}\sum_{i=1}^{N}\eta_t\lambda_i^2(t)\right), \tag{27}$$

Next, we deal with the second term on the right side of (18). Similar to Lemma 2, by using the strong concavity of $\tilde{\mathcal{L}}_{i,t}$ with respect to λ and the θ_t-strong convexity with respect to $\mathbf{x}$, we have

$$\begin{aligned}\sum_{t=1}^{T}\sum_{i=1}^{N}\left[\tilde{\mathcal{L}}_{i,t}(\mathbf{x}_i(t),\lambda) - \tilde{\mathcal{L}}_{i,t}(\mathbf{x},\lambda_i(t))\right] \le& \sum_{t=1}^{T}\frac{\eta_t}{2}\sum_{i=1}^{N}\|\nabla_{\mathbf{x}}\tilde{\mathcal{L}}_{i,t}(\mathbf{x}_i(t),\lambda_i(t))\|^2 + \sum_{t=1}^{T}\frac{\mu_t}{2}\sum_{i=1}^{N}\|\nabla_{\lambda}\tilde{\mathcal{L}}_{i,t}(\mathbf{x}_i(t),\lambda_i(t))\|^2\\ &+\sum_{t=1}^{T}\frac{\sum_{i=1}^{N}\left[\|\lambda_i(t)-\lambda\|^2 - \|\lambda_i(t+1)-\lambda\|^2\right]}{2\mu_t} - \sum_{t=1}^{T}\frac{\theta_t}{2}\sum_{i=1}^{N}\|\lambda_i(t)-\lambda\|^2\\ &+\sum_{t=1}^{T}\frac{\sum_{i=1}^{N}\left[\|\mathbf{x}_i(t)-\mathbf{x}\|^2 - \|\mathbf{x}_i(t+1)-\mathbf{x}\|^2\right]}{2\eta_t} - \frac{\sigma}{2}\sum_{t=1}^{T}\sum_{i=1}^{N}\|\mathbf{x}_i(t)-\mathbf{x}\|^2.\end{aligned} \tag{28}$$

Substituting the definition of $\tilde{\mathcal{L}}_{i,t}$ (23) into the left side of (27), and setting $\mathbf{x} = \mathbf{x}_\gamma$, $\mathbf{x}_i(1) = \lambda_i(1) = 0$, we could derivate the following inequality,

$$\begin{aligned}&\sum_{t=1}^{T}\sum_{i=1}^{N}\left[f_{i,t}(\mathbf{x}_i(t)) - f_{i,t}(\mathbf{x}_\gamma)\right] + \lambda\sum_{t=1}^{T}\sum_{i=1}^{N}(g(\mathbf{x}_i(t))+\gamma) - \frac{\lambda^2}{2}\left(\sum_{t=1}^{T}\theta_t + \frac{1}{\mu_1} - \theta_1\right)\\ &\le \sum_{t=1}^{T}\left(\eta_t - \frac{\theta_t}{2} + \mu_t\theta_t^2\right)\sum_{i=1}^{N}\lambda_i^2(t) + \sum_{t=1}^{T}\sum_{i=1}^{N}\frac{\|\mathbf{x}_i(t)-\mathbf{x}_\gamma\|^2}{2}\left[\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} - \sigma\right]\\ &\quad+\sum_{t=1}^{T}\sum_{i=1}^{N}\frac{\|\lambda_i(t)-\lambda\|^2}{2}\left[\frac{1}{\mu_t} - \frac{1}{\mu_{t-1}} - \theta_t\right] + \sum_{i=1}^{N}\|\mathbf{x}_\gamma\|^2\left(\frac{1}{2\eta_1} - \frac{\sigma}{2}\right) + NG^2\sum_{t=1}^{T}\eta_t + N(D^2+\gamma^2)\sum_{t=1}^{T}\mu_t.\end{aligned} \tag{29}$$

By noting $\lambda \in \mathbf{R}_+$, we have

$$\lambda\sum_{t=1}^{T}\sum_{i=1}^{N}(g(\mathbf{x}_i(t))+\gamma) - \frac{\lambda^2}{2}\left(\sum_{t=1}^{T}\theta_t + \frac{1}{\mu_1} - \theta_1\right) \le \frac{\left[\sum_{t=1}^{T}\sum_{i=1}^{N}(g(\mathbf{x}_i(t))+\gamma)\right]_+^2}{2N\left(\sum_{t=1}^{T}\theta_t + \frac{1}{\mu_1} - \theta_1\right)}. \tag{30}$$

Substituting (27), (29), (30) into (18), and using (8), we could obtain an upper bound associated with regret and CCV,

$$\begin{aligned}&\mathrm{Reg}(i,T) - \frac{GNT}{\sigma}\gamma + \frac{\left[\sum_{t=1}^{T}\sum_{i=1}^{N}(g(\mathbf{x}_i(t))+\gamma)\right]_+^2}{2N\left(\sum_{t=1}^{T}\theta_t + \frac{1}{\mu_1} - \theta_1\right)} \le 4N^2G^3 f_\xi\sum_{t=1}^{T}\eta_t + \frac{NR^2}{2\eta} + N(D^2+\gamma^2)\sum_{t=1}^{T}\mu_t\\ &+\sum_{t=1}^{T}\sum_{i=1}^{N}\frac{\|\mathbf{x}_i(t)-\mathbf{x}_\gamma\|^2}{2}\left[\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} - \sigma\right] + \sum_{t=1}^{T}\sum_{i=1}^{N}\frac{\|\lambda_i(t)-\lambda\|^2}{2}\left[\frac{1}{\mu_t} - \frac{1}{\mu_{t-1}} - \theta_t\right] + \left(\frac{1}{2\eta_1} - \frac{\sigma}{2}\right)\sum_{i=1}^{N}\|\mathbf{x}_\gamma\|^2\\ &+\sum_{t=1}^{T}\sum_{i=1}^{N}\left(\eta_t - \frac{\theta_t}{2} + \mu_t\theta_t^2 + 4Nf_\xi G^3\eta_t\right)\lambda_i^2(t)\\ &\le \mathcal{O}\left(\sum_{t=1}^{T}\eta_t + \sum_{t=1}^{T}\mu_t + \sum_{t=1}^{T}\left[\frac{1}{\eta_t} - \frac{1}{\eta_{t-1}} - \sigma\right]\right) \le \mathcal{O}\left(T^{\max\{\beta,1-\beta/2\}}\right),\end{aligned} \tag{31}$$

where the last two inequalities are based on the parameters set in the theorem. And we can obtain $\mathrm{Reg}(i,T) \le \mathcal{O}(T^{2/3})$ by setting $\beta = 2/3$.

Combine (22) and (31), we obtain that

$$\frac{\left[\sum_{t=1}^{T}\sum_{i=1}^{N}\left(g(\mathbf{x}_i(t))+\gamma\right)\right]_+^2}{2N\left(\sum_{t=1}^{T}\theta_t+\frac{1}{\mu_1}-\theta_1\right)}\leq FNT+o(T)\leq\mathcal{O}(T).$$

Further we have $CCV\leq\sqrt{\mathcal{O}(T\sum_{t=1}^{T}\theta_t)}-\gamma NT$. By choosing proper $\gamma=c_1(T^{-\beta/2})$, we can guarantee that the LTC is strictly satisfied. □

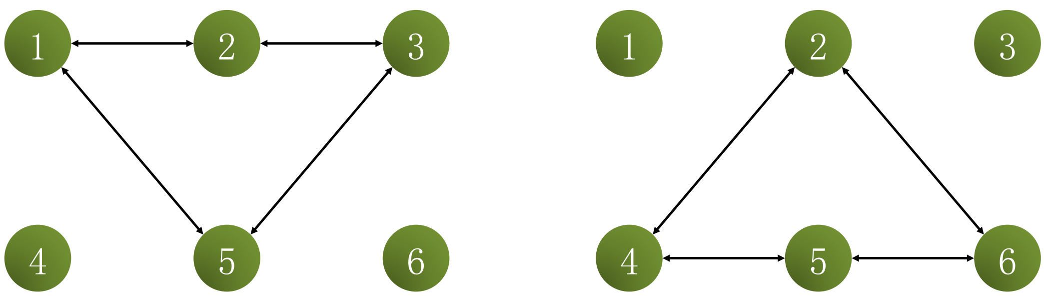
Remark 2. *To the best of our knowledge, among existing works on online optimization that strictly enforce LTC, the optimal regret bound is $\mathcal{O}(T^{3/4})$ for fixed-step algorithms [11] and $\mathcal{O}(T^{2/3})$ for time-varying step algorithms [13]. However, these results are derived in a centralized framework, whereas our work considers a distributed setting. Notably, this paper is the first to study online optimization with strictly satisfied LTC in the distributed framework.*

4. Numerical Simulations

We consider the time-varying communication graphs depicted in Figure 1. A distributed online regularized linear regression problem is formulated as follows:

$$\begin{aligned}&\min\sum_{t=1}^{T}\sum_{i=1}^{N}\frac{1}{2}\left(\mathbf{k}_i(t)^\top\mathbf{x}-b(t)\right)^2\\&\text{s.t. } g_s(\mathbf{x})=L-[\mathbf{x}]_s\leq 0,\\&\qquad g_{d+s}(\mathbf{x})=[\mathbf{x}]_s-U\leq 0,\ s=1,\ldots,d\end{aligned}$$

where $\mathbf{k}_i(t)\in\mathbf{R}^d$ and $b(t)\in\mathbf{R}$ are the local parameters that only revealed to agent i at time t. Each component of $\mathbf{k}_i(t)$ follows the uniform distribution over $[-1,1]$, $b(t)=\mathbf{k}_i(t)^\top\bar{\mathbf{x}}+\kappa(t)$, where $[\bar{\mathbf{x}}]_i=\begin{cases}1, & 1\leq i\leq\lfloor d/2\rfloor\\ 0, & \text{otherwise}\end{cases}$ and $\kappa(t)\sim\mathcal{N}(0,1)$ is the noise that follows the standard normal distribution. We set $d=5,\ T=2000,\ U=0.5,\ L=0$ and $g(\mathbf{x})=\max_{s\in\{1,\ldots,2d\}}g_s(\mathbf{x})$.



(a) $t=0,2,\cdots$ (b) $t=1,3,\cdots$

Figure 1. Time-varying 2-strongly connected communication graph varies sequentially among (**a**) and (**b**).

We validate Algorithm 1 (fixed step size) and Algorithm 2 (adaptive step size), respectively, and the result of regret is shown in Figure 2. It can be seen that both algorithms achieve the sublinear regret, and the improved version obtains a better regret, which also conforms to the results of Theorem 1 and Theorem 2. Figure 3 shows the comparison of LTC. It can be seen that LTC is strictly satisfied in both algorithms, and Algorithm 2 has better performance. These figures also show that the actual effect of the algorithm is consistent with the conclusions of the theorems.

In addition, we consider different values of γ by setting $\beta=1/2,2/3,1$, respectively. The corresponding regrets and LTCs for different γ are shown in Figure 4. Note that the settings $\gamma=0$ and $\gamma=cT^{-1}$ are excluded by our theorems, as they cannot guarantee strict satisfaction of LTC, which is also evident in Figure 4. From the curve with label $\gamma=0$, which can also be regarded as the classical online algorithm without the shrinkage parameter γ (such as [13,16]), although the result is slightly better, unfortunately, it cannot strictly satisfy LTC. From Figure 4, we can know that there is a trade-off between regret and LTC. Strictly satisfying LTC requires sacrificing regret's convergence rate. Furthermore, a more stringent shrinkage constraint set (larger γ) will lead to better LTC guarantees (LTC being strictly satisfied faster and better), but a worse regret bound.

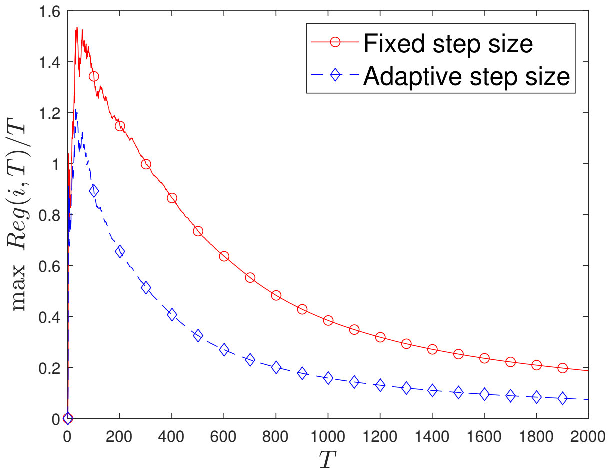
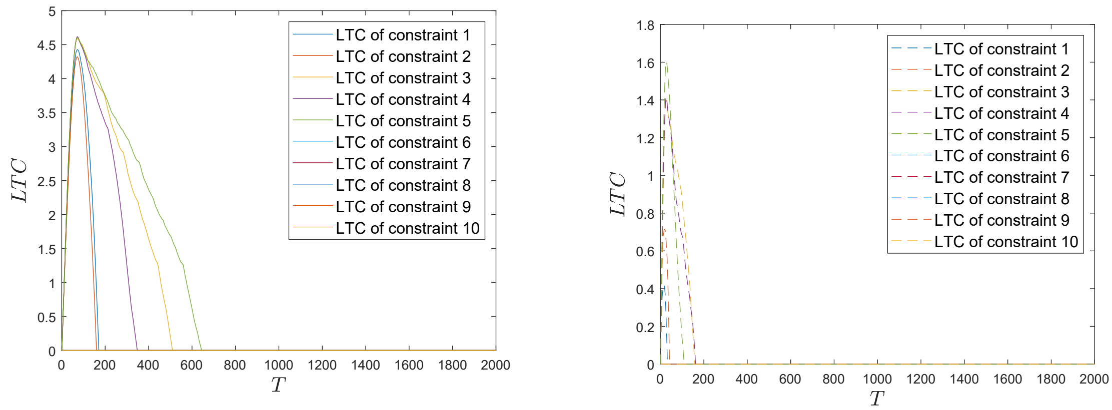
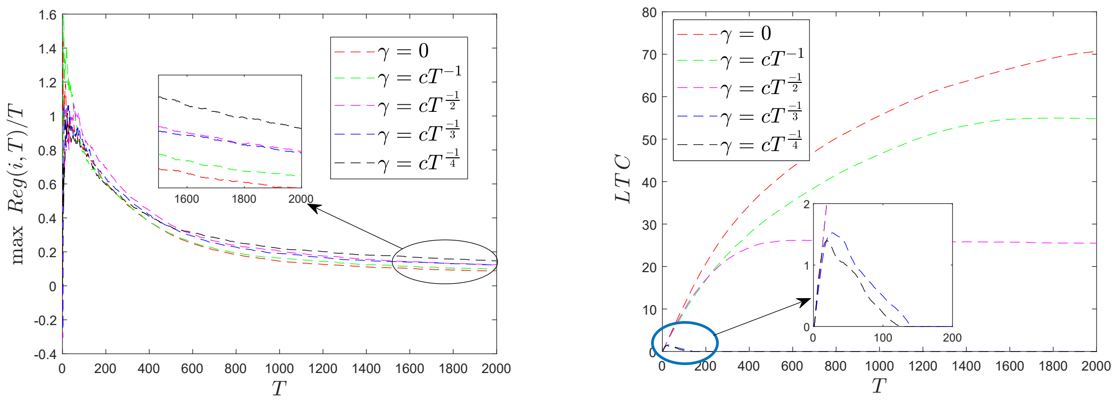


Figure 2. Regret of Algorithm 1 and its improved version.



(a) Algorithm 1 (b) Algorithm 2

Figure 3. CCV of Algorithms 1 and 2.



(a) Regret (b) LTC

Figure 4. Regret and LTC of different γ.

5. Conclusions

This paper is the first to study a special online optimization problem with LTC in the distributed framework, where LTC needs to be strictly satisfied rather than just sublinear convergence. Based on the gradient algorithm and regularized Lagrangian function under the centralized framework, we design the fixed step size DOO-LTC algorithm and the improved adaptive step size algorithm. While LTC is strictly satisfied, we obtain the $\mathcal{O}(T^{3/4})$ and $\mathcal{O}(T^{2/3})$ regret upper bound, respectively, which matches the optimal result in the centralized framework [11,13]. Finally, our conclusions are verified in the numerical simulation. In the future, we will focus on designing communication strategies to reduce the communication burdens and then enhance the scalability and robustness of the algorithm in large-scale networks. In addition, the doubly stochastic property of the weight matrix is considered to be relaxed as future work.

Author Contributions: D.Z.: Writing—original draft, Methodology; S.Y.: Writing—review and editing, Supervision, Methodology, Conceptualization, Funding acquisition.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant Nos. 62173087 and 62176056 and the Fundamental Research Funds for the Central Universities, in part by the Alexander von Humboldt Foundation of Germany, and in part by Young Elite Scientists Sponsorship Program by CAST, 2021QNRC001.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

Use of AI and AI-Assisted Technologies: No AI tools were utilized for this paper.

References

1. Avi, G.; Catherine, T. Online Display Advertising: Targeting and Obtrusiveness. *Mark. Sci.* **2011**, *30*, 389–404.
2. Balseiro, S.R.; Lu, H.; Mirrokni, V. The Best of Many Worlds: Dual Mirror Descent for Online Allocation Problems. *Oper. Res.* **2023**, *71*, 101–119.
3. Chen, T.; Ling, Q.; Giannakis, G.B. An Online Convex Optimization Approach to Proactive Network Resource Allocation. *IEEE Trans. Signal Process.* **2017**, *65*, 6350–6364.
4. Mairal, J.; Bach, F.; Ponce, J.; et al. Online Dictionary Learning for Sparse Coding. In Proceedings of the 26th International Conference on Machine Learning (ICML), Montreal, QC, Canada, 14–18 June 2009; pp. 689–696.
5. Zinkevich, M. Online Convex Programming and Generalized Infinitesimal Gradient Ascent. In Proceedings of the 20th International Conference on Machine Learning (ICML), Washington, DC, USA, 21–24 August 2003; pp. 928–936.
6. Flaxman, A.D.; Kalai, A.T.; McMahan, H.B. Online Convex Optimization in the Bandit Setting: Gradient Descent without a Gradient. In Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), Vancouver, BC, Canada, 23–25 January 2005; pp. 385–394.
7. Agarwal, A.; Dekel, O.; Xiao, L. Optimal Algorithms for Online Convex Optimization with Multi-Point Bandit Feedback. In Proceedings of the 23rd Annual Conference on Learning Theory (COLT), Haifa, Israel, 27–29 June 2010; pp. 28–40.
8. Koppel, A.; Jakubiec, F.Y.; Ribeiro, A. A Saddle Point Algorithm for Networked Online Convex Optimization. *IEEE Trans. Signal Process.* **2015**, *63*, 5149–5164.
9. Yuan, J.; Lamperski, A. Online Convex Optimization for Cumulative Constraints. In Proceedings of the 32nd Neural Information Processing Systems (NeurIPS), Montreal, QC, Canada, 3–8 December 2018; pp. 6140–6149.
10. Zhang, W.; Zhao, P.; Zhu, W.; et al. Projection-Free Distributed Online Learning in Networks. In Proceedings of the 34th International Conference on Machine Learning (ICML), Sydney, NSW, Australia, 6–11 August 2017; Volume 70, pp. 4054–4062.
11. Mahdavi, M.; Jin, R.; Yang, T. Trading Regret for Efficiency: Online Convex Optimization with Long Term Constraints. *J. Mach. Learn. Res.* **2012**, *13*, 2503–2528.
12. Yi, X.; Li, X.; Yang, T.; et al. Regret and Cumulative Constraint Violation Analysis for Online Convex Optimization with Long Term Constraints. In Proceedings of the 38th International Conference on Machine Learning (ICML), online, 18–24 July 2021; Volume 139, pp. 11998–12008.
13. Jenatton, R.; Huang, J.C.; Archambeau, C. Adaptive Algorithms for Online Convex Optimization with Long-Term Constraints. In Proceedings of the 33rd International Conference on Machine Learning (ICML), New York, NY, USA, 19–24 June 2016; Volume 48, pp. 402–411.
14. Yu, H.; Neely, M.J. A Low Complexity Algorithm with O($\sqrt{T}$) Regret and O(1) Constraint Violations for Online Convex Optimization with Long Term Constraints. *J. Mach. Learn. Res.* **2020**, *21*, 1–24.
15. Guo, H.; Liu, X.; Wei, H.; et al. Online Convex Optimization with Hard Constraints: Towards the Best of Two Worlds and Beyond. In Proceedings of the 36th International Conference on Neural Information Processing Systems, New Orleans, LA, USA, 28 November–9 December 2022; Volume 35, pp. 36426–36439.
16. Yuan, D.; Proutiere, A.; Shi, G. Distributed Online Optimization with Long-Term Constraints. *IEEE Trans. Autom. Control* **2022**, *67*, 1089–1104.
17. Yi, X.; Li, X.; Yang, T.; et al. Distributed Bandit Online Convex Optimization with Time-Varying Coupled Inequality Constraints. *IEEE Trans. Autom. Control* **2021**, *66*, 4620–4635.
18. Li, X.; Yi, X.; Xie, L. Distributed Online Optimization for Multi-Agent Networks with Coupled Inequality Constraints. *IEEE Trans. Autom. Control* **2021**, *66*, 3575–3591.
19. Yuan, D.; Ho, D.W.C.; Jiang, G.P. An Adaptive Primal-Dual Subgradient Algorithm for Online Distributed Constrained Optimization. *IEEE Trans. Cybern.* **2018**, *48*, 3045–3055.
20. Abernethy, J.D.; Bartlett, P.L.; Rakhlin, A.; et al. Optimal Strategies and Minimax Lower Bounds for Online Convex Games. In Proceedings of the 21st Annual Conference on Learning Theory (COLT), Helsinki, Finland, 9–12 July 2008.

21. Islam, M.A.; Ren, S.; Quan, G.; et al. An Adaptive Primal-Dual Subgradient Algorithm for Online Distributed Constrained Optimization. *IEEE Trans. Cybern.* **2018**, *48*, 3045–3055.
22. Xiong, M.; Zhang, B.; Ho, D.W.C.; et al. Event-Triggered Distributed Stochastic Mirror Descent for Convex Optimization. *IEEE Trans. Neural Netw. Learn. Syst.* **2023**, *34*, 6480–6491.