Heterogeneous Online Learning for "Thing-Adaptive" Fog Computing in IoT

Tianyi Chen, *Student Member, IEEE*, Qing Ling, Yanning Shen, *Student Member, IEEE*, and Georgios B. Giannakis, *Fellow, IEEE*

***Abstract*—Internet of Things (IoT) is featured with its seamless connectivity of billions of smart devices, which offer different functionalities and serve various personalized tasks. To meet the task-specific requirements such as latency and privacy, the fog computing emerges to extend cloud computing services to the edge of the Internet backbone. This paper deals with *online fog computing* emerging in IoT, where the goal is to balance computation and communication at fog networks on-the-fly to minimize service latency. Due to heterogeneous devices and human participation in IoT, the online decisions here need to flexibly adapt to the temporally unpredictable user demands and availability of fog resources. By generalizing the classic online convex optimization (OCO) framework, the low-latency fog computing task is first formulated as an OCO problem involving both time-varying loss functions and time-varying constraints. These constraints are revealed after making decisions, and allow instantaneous violations yet they must be satisfied in the long term. Tailored for heterogeneous tasks in IoT, a "thing-adaptive" online saddle-point (TAOSP) scheme is developed, which automatically adjusts the stepsize to offer desirable *task-specific* learning rates. It is established that without prior knowledge of the time-varying parameters, TAOSP simultaneously yields near-optimality and feasibility, provided that the best dynamic solutions vary slowly over time. Numerical tests corroborate that our novel approach outperforms the state-of-the-art in minimizing network latency.**

***Index Terms*—Heterogeneous tasks, Internet of Things (IoT), mobile edge computing, online learning, saddle-point method.**

I. Introduction

INTERNET-OF-THINGS (IoT) envisions an intelligent network infrastructure offering task-specific services, such as those in smart home, healthcare, and smart cities [2]–[7]. One of the critical challenges in IoT is the pronounced *heterogeneity* due to a large number of devices and tasks.

Manuscript received January 3, 2018; revised May 30, 2018; accepted July 22, 2018. Date of publication July 26, 2018; date of current version January 16, 2019. This work was supported in part by the NSF under Grant 1509040, Grant 1508993, and Grant 1711471, in part by NSF China under Grant 61573331, and in part by NSF Anhui under Grant 1608085QF130. This paper was presented in part at the IEEE Asilomar Conference on Signals, Systems, and Computers, Pacific Grove, CA, USA, Oct. 29–Nov. 1, 2017 [1]. *(Corresponding author: Georgios B. Giannakis.)*

T. Chen, Y. Shen, and G. B. Giannakis are with the Department of Electrical and Computer Engineering and the Digital Technology Center, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: chen3827@umn.edu; shenx513@umn.edu; georgios@umn.edu).

Q. Ling is with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou 510006, China (e-mail: qingling@ieee.org).

Digital Object Identifier 10.1109/JIOT.2018.2860281

Device Heterogeneity: The computational and communication capacities of each device differ due to differences in hardware (e.g., CPU frequency), communication protocol (e.g., ZigBee and WiFi), and energy availability (e.g., battery level).

Task Heterogeneity: The tasks carried out on various devices can be considerably diverse, e.g., motion sensors monitor human behavior in a smart home [8], and cameras are responsible for recognizing vehicle plates in a parking garage.

All types of heterogeneity will lead to major differences in computation and communication latency of serving IoT tasks among individual IoT devices. Together with other unique features of IoT including *latency-sensitive*, and *unpredictable dynamics* due to human-in-the-loop, it all calls for innovations in network design and management for IoT [9].

To ensure desired user experience, IoT tasks nowadays are supported by a promising architecture termed *fog* that distributes computation, communication, and storage closer to the end IoT users, along the cloud-to-things continuum [10]. Regarding network design, network formation, and protocols to integrate cloud resources into the mobile fog networks have been extensively studied [11], [12]. From the fog management perspective, joint communication and computation approaches have been developed in [13] and [14]; latency-constrained extensions have been considered in [15]–[17]; and resource-aware quality-of-service management in [18]. However, the approaches in [13]–[18] are mainly for static offline settings, and their online variants have not been explored. Tailored for dynamic fog networks with time-varying user demands, a Lyapunov optimization-based approach is presented in [19], an MDP-based approach is advocated in [20], and an online approach with competitive ratio guarantee is reported in [21]; see [22] for a recent survey on related topics. Nevertheless, the assumption of stationarity[1] that is essential in stochastic optimization may not hold in practice due to human participation, and the precise information within a given time window leveraged in competitive analysis is also often unavailable. Therefore, online fog network management, which is robust to nonstationary dynamics and suitable for heterogenous IoT tasks, remains an uncharted territory [12], [22].

Targeting an efficient solution for the unique features present in IoT setups, we will employ online convex

[1]In this context, stationarity means that the time-varying quantities related to fog computing are drawn from a fixed probability distribution.

optimization (OCO), which is an emerging methodology for sequential tasks especially when the sequence of convex costs varies in an unknown and possibly adversarial manner [23]. Aiming to empower traditional fog management policies, most available OCO works benchmark algorithms with a *static regret*, which measures the difference of costs (also known as losses) between the online solution and the best static solution in hindsight [24], [25]. However, static regret is not a comprehensive performance metric in dynamic settings such as those encountered with fog computing in IoT [9].

Recent works extend the analysis of static to that of *dynamic regret* [26], [27], but they deal with time-invariant constraints that cannot be violated instantaneously. The *long-term* effect of such instantaneous violations was also studied in [28] and [29], where the focus is on static regret and time-invariant constraints. Unfortunately, the fog computing setups considered here require flexible adaptation of online decisions to the dynamic IoT user demands, and the availability of resources. In a generic network optimization setting, algorithms for OCO with time-varying constraints have been developed [30], [31], but they are not suitable for the heterogeneous IoT settings.

To account for the heterogeneous nature of IoT applications, and to meet their stringent latency requirements, this paper broadens the scope of OCO to the regime with time-varying objectives and constraints, and introduces a "thing-adaptive" online learning algorithm, which allows task-specific learning-rates and provides guarantees on optimality and feasibility.

Relative to prior art, the main contributions are as follows.

1) We formulate the fog computation offloading task emerging in IoT applications as a constrained OCO problem. The resultant online learning task generalizes the OCO framework with only adversarial costs in [23], [24], [26], and [27] to account also for possibly adversarial constraints (Sections II and III).
2) We develop a "thing-adaptive" online saddle-point (TAOSP) algorithm for this novel OCO setting, which incorporates an adaptive matrix stepsize to automatically adjust task-specific learning rates (one per coordinate or thing), and yields simultaneously sublinear dynamic regret and fit, provided that the best dynamic solutions vary slowly over time (Section IV).
3) We apply our novel TAOSP algorithm to fog computing, and compare it with popular alternatives that rely on stochastic gradient [32], and task-agnostic schemes [30]. Simulations demonstrate marked performance gain of TAOSP (Section V).

Notation: $(\cdot)^\top$ stands for vector and matrix transposition, and $\|\mathbf{x}\|$ denotes the ℓ_2-norm of a vector $\mathbf{x}$. Inequalities for vectors $\mathbf{x} > \mathbf{0}$, and the projection $[\mathbf{a}]^+ := \max\{\mathbf{a}, \mathbf{0}\}$ are entry-wise.

II. Online Fog Computation Offloading

In this section, we introduce the time-varying fog computing setup, and formulate its computation offloading problem for low-latency IoT service provisioning.

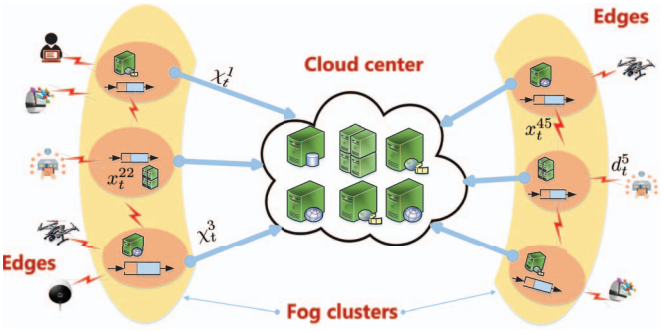


Fig. 1. Diagram for online fog computation offloading: the edges represent IoT devices in the edge layer, the fog clusters contain locally connected fog nodes, and the cloud center is the data center in the cloud layer.

A. Fog Computing Setup

Consider IoT tasks supported by a fog network with an edge layer, a fog layer, and a cloud layer [10], [21]. The edge layer contains *heterogeneous* low-power IoT sensors (e.g., wearable watches and temperature sensors). Due to their low-power design, IoT sensors have minimal on-device computational capability, and frequently offload their collected data to the nearby fog nodes (e.g., smartphones and high-tech routers) at the fog layer for further processing [33]. The communications between edges and fog nodes are typically through low-throughput but energy-efficient wireless connection, such as Bluetooth or ZigBee [12]. The fog layer consists of N nodes in the set $\mathcal{N} := \{1, \ldots, N\}$ with moderate processing capability. Part of the workload at the fog is collaboratively processed by the processors in smartphones or high-tech routers (also known as fog servers) to meet the stringent latency requirement; while the rest is offloaded to the remote data center in the cloud layer via high-throughput wireless or wireline connection [12]. In a related context, the fog nodes are also referred to IoT gateways, which bridge the wireless sensor networks with the Internet backbone [34].

Per time slot t, each fog node n collects streaming data requests d_t^n from all its nearby sensors. Once receiving d_t^n, the nth fog node has to make a decision over *three options*.

1) Offloading an amount χ_t^n to the remote cloud center.
2) Offloading an amount x_t^{nk} to its nearby fog node k for collaborative computing.
3) Processing an amount x_t^{nn} using the *in-situ* fog servers, subject to the availability of computational resources.

The optimization variable $\mathbf{x}_t$ consists of the cloud offloading, local offloading, and local processing amounts, namely, $\mathbf{x}_t := [\chi_t^1, \ldots, \chi_t^N, x_t^{11}, \ldots, x_t^{NN}]^\top$ (see also Fig. 1).

Assuming that each fog node has a local data queue to buffer unserved workloads, the instantaneously served workload (offloading plus processing) per node is not necessarily equal to the data arrival rate. Instead, a long-term constraint is imposed to ensure that the cumulative amount of served workloads is no less than the arrived amount at node n over a given time period of T slots; that is,

$$\sum_{t=1}^{T} g_t^n(\mathbf{x}_t) \leq 0, \quad \forall n \tag{1a}$$

$$g_t^n(\mathbf{x}_t) := d_t^n + \sum_{k\in\mathcal{N}_n^{\text{in}}} x_t^{kn} - \sum_{k\in\mathcal{N}_n^{\text{out}}} x_t^{nk} - \chi_t^n - x_t^{nn} \quad (1b)$$

where $\mathcal{N}_n^{\text{in}}$ and $\mathcal{N}_n^{\text{out}}$ represent the sets of fog nodes with in-coming links to node n, and those with out-going links from node n, respectively. The offloading limit of the communication link from fog node n to the remote cloud is $\bar{\chi}^n$, the maximum offloading capacity of link n-to-k is $\bar{x}^{nk}$, and the computation capability of fog node n is $\bar{x}^{nn}$. Due to different communication protocols and diverse processing cores used, the magnitudes of elements in $\{\bar{\chi}^n, \bar{x}^{nk}, \bar{x}^{nn}\}$ can vary considerably. Nevertheless, with $\bar{\mathbf{x}}$ collecting all the aforementioned known bounds, the feasible set is expressed as $\mathcal{X} := \{\mathbf{0} \le \mathbf{x}_t \le \bar{\mathbf{x}}\}$.

Also worth mentioning is that it can further incorporate other considerations in different settings as follows.

1) When the computing resources at the fog nodes are virtualized by means of virtual machines (VMs), only fog nodes with common VMs can perform collaborative computing [12]; and while only the offloading amount at each fog node was bounded by $\bar{x}^{nk}$, the total received amount for collaborative computing can be also constrained.
2) When the fog network serves heterogeneous tasks such as those in a smart building [35], the local offloading for collaborative computing can appear only between two fog nodes serving the same IoT task. For those fog nodes capable of performing multiple tasks, we can virtually split them into multiple single-task fog nodes.

Clearly, corresponding to all these practical considerations is a more involved polyhedral feasible set $\mathcal{X}$.

B. Toward Low-Latency Fog Computing

The figure of merit in deciphering the optimum $\mathbf{x}_t$ is the network delay of the online edge processing and offloading decisions. Specifically, as the computation delay is usually negligible for data centers with thousands of high-performance servers, the latency for cloud offloading amount χ_t^n is mainly due to the communication delay, which is modeled as a time-varying convex function $c_t^n(\chi_t^n)$ depending on the congestion level of the network during slot t. Likewise, the communication delay of the local offloading decision x_t^{nk} from node n to a nearby node k is denoted by $c_t^{nk}(x_t^{nk})$, and its magnitude is much lower than that of cloud offloading. In addition, latency of the edge processing amount x_t^{nn} comes from the computational delay due to its limited computation capability. The computational delay is represented as a time-varying function $h_t^n(x_t^{nn})$ capturing dynamics during the edge computing processes.

The overall performance of decision $\mathbf{x}_t$ is considered next.

Aggregate Delay: Per slot t, the aggregate network delay $f_t(\mathbf{x}_t)$ includes the computational delay at all fog nodes plus the communication delay at all links, namely,

$$f_t(\mathbf{x}_t) := \sum_{n\in\mathcal{N}} \Big(\underbrace{c_t^n(\chi_t^n) + \sum_{k\in\mathcal{N}_n^{\text{out}}} c_t^{nk}(x_t^{nk})}_{\text{communication}} + \underbrace{h_t^n\big(x_t^{nn}\big)}_{\text{computation}} \Big). \quad (2)$$

Note that through proper parallelization, communication, and computation tasks sometimes can be executed in parallel, and the actual delay experienced by users may depend on the level of such parallelization. As a result, the aggregate delay cannot accurately reflect the performance that directly affects user experience [21] in such cases, and the maximum delay discussed next is an alternative performance metric.

Maximum Delay: Per slot t, the worst-case network delay $f_t(\mathbf{x}_t)$ is the maximum of computational delay and the communication delay at all fog nodes, namely,

$$f_t(\mathbf{x}_t) := \sum_{n\in\mathcal{N}} \max_{k\in\mathcal{N}_n^{\text{out}}} \Big\{ \underbrace{c_t^n\big(\chi_t^n\big), c_t^{nk}\Big(x_t^{nk}\Big)}_{\text{communication}}, \underbrace{h_t^n\big(x_t^{nn}\big)}_{\text{computation}} \Big\}. \quad (3)$$

Alternatively, aggregate maximum delay can be also considered, which is the sum of the computation delay plus the maximum communication delay over all offloading links at each fog node.

Aiming to minimize the network delay (in either aggregate or worst-case sense) while serving all the IoT workloads in the long term, the optimal offloading strategy in this fog network is the solution of the following optimization problem:

$$\min_{\{\mathbf{x}_t\in\mathcal{X},\forall t\}} \sum_{t=1}^{T} f_t(\mathbf{x}_t) \quad \text{s.t.} \quad \sum_{t=1}^{T} g_t^n(\mathbf{x}_t)s \le 0 \quad \forall n. \quad (4)$$

For the optimization in (4), if the objective and the constraint functions are *known* ahead of the time and the horizon T is not prohibitively large, the fog computing decisions can be found by utilizing any off-the-shelf convex optimization solver. Not to mention the potentially high complexity of the offline solver, the crux is that communication and computation delays as well as user demands are usually unknown before allocating resources due to the unpredictable routing, network congestion, device malfunctions, and nowadays malicious attacks in IoT [2]. This motivates a fully *causal setting*, where the network delay $f_t(\mathbf{x}_t)$ and the data requests $\{d_t^n\}$ within slot t are not known when making the offloading and computing decision $\mathbf{x}_t$, but are revealed at the end of slot t after deciding $\mathbf{x}_t$.

Remark 1: For formulation, three remarks are in order.

1) While the communication delay is assumed to be a convex function of the offloaded amount of data, it can be nonconvex in general. Dealing with nonconvex delay functions is also of interest, and is in our future agenda.
2) The considered model only incorporates three network layers, but it can be readily extended to *multilayer* structures, where several intermediate fog layers are deployed between the IoT devices and the remote data centers.
3) Although (2) or (3) only captures the network delay effect, other relevant factors can be also incorporated in our OCO setting, e.g., throughput and energy consumption [12].

III. Online Convex Optimization for Fog Computing

In this section, we will formulate the fog computation offloading task as a constrained OCO problem, and provide

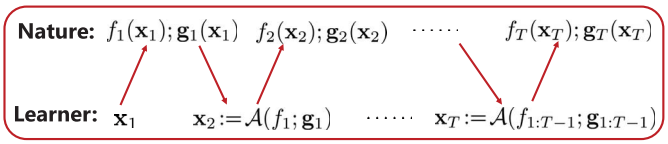


Fig. 2. Diagram of OCO with time-varying constraints.

pertinent performance metrics to evaluate algorithms in this setting.

A. OCO With Time-Varying Constraints

Targeting a customized solution to the challenging fog computing task (4), our idea is to leverage OCO tools to design algorithms with provable performance guarantees. However, most available OCO works do not allow instantaneous violations of constraints, which is not applicable to the fog computing setup. This prompts us to broaden the applicability of the classical OCO setting [23], [24] to the regime with dynamic regret and time-varying constraints.

To model the task, consider the fog computing problem as a repeated game between a learner and nature, as it appears in OCO [23]. With I denoting the dimension of $\mathbf{x}_t$, the learner $\mathcal{A}$ selects an action $\mathbf{x}_t$ from a known and fixed convex set $\mathcal{X} \subseteq \mathbb{R}^I$ per slot t, and then nature reveals not only a loss function $f_t(\cdot) : \mathbb{R}^I \to \mathbb{R}$ but also a time-varying constraint function $\mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}$, where $\mathbf{g}_t(\cdot) : \mathbb{R}^I \to \mathbb{R}^I$. Different from the known and fixed set $\mathcal{X}$, the constraint $\mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}$ can vary arbitrarily from slot to slot. Moreover, the fact that it is revealed after the learner $\mathcal{A}$ performs her/his decision makes it impossible to be satisfied at every time slot; see the setting in Fig. 2. Therefore, a more realistic goal in this context is to find a sequence of online solutions $\{\mathbf{x}_t \in \mathcal{X}\}$ that minimizes the aggregate loss, and ensures that the constraints $\{\mathbf{g}_t(\mathbf{x}_t) \leq \mathbf{0}\}$ are satisfied in the long term on average. Furthermore, the fact that f_t and $\mathbf{g}_t$ are revealed after the learner makes decisions, also accounts for possibly adversarial scenarios in IoT, e.g., malicious fog nodes and communication links present to strategically impede computation and communication [36].

Generalizing the OCO framework [23], [24] to accommodate such varying constraints, we consider the following problem:

$$\min_{\{\mathbf{x}_t \in \mathcal{X}, \forall t\}} \sum_{t=1}^{T} f_t(\mathbf{x}_t) \quad \text{s.t.} \quad \sum_{t=1}^{T} \mathbf{g}_t(\mathbf{x}_t) \leq \mathbf{0}. \tag{5}$$

Clearly, with $\mathbf{g}_t(\mathbf{x}_t)$ specified by (1) and $f_t(\mathbf{x}_t)$ as in (2) or (3), the problem (5) captures and further generalizes the fog computing problem in Section II. Taking a step further, the novel OCO framework can be also applied to tasks ranging from power control in wireless communication [32], to geographical load balancing in cloud networks [37].

Before our efficient algorithm development for thing-adaptive learning tasks and performance analysis to be carried out in Section IV, we first introduce suitable optimality and feasibility metrics.

B. Optimality and Feasibility Metrics

With regard to performance of online decisions, static regret is commonly adopted by OCO schemes to measure the difference between the aggregate loss of an OCO algorithm and that of the best fixed solution in hindsight [23], [24]. Generalizing the static regret to accommodate time-varying constraints in (5), we have $\text{Reg}_T^{\text{s}} := \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}^*)$, where the best static solution is $\mathbf{x}^* \in \arg\min_{\mathbf{x}\in\mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x})$, s.t. $\mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}$, $\forall t$. Though widely used in different OCO context, the *static regret* relies on a rather coarse benchmark, which may be less useful in dynamic settings. Quantitatively, the gap between the best static and the best dynamic solutions in terms of the aggregate loss can be as large as $\mathcal{O}(T)$ [26].

In response to the quest for appropriate benchmarks in the dynamic IoT setup with constraints, two metrics are considered: 1) *dynamic regret* and 2) *dynamic fit*. The notion of dynamic regret offers a competitive performance measure of online algorithms, given by

$$\text{Reg}_T^{\text{d}} := \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t\left(\mathbf{x}_t^*\right) \tag{6}$$

where the benchmark is now formed via a sequence of the best dynamic solutions $\{\mathbf{x}_t^*\}$ for the instantaneous cost minimization problem subject to the instantaneous constraint, namely,

$$\mathbf{x}_t^* \in \arg\min_{\mathbf{x}\in\mathcal{X}} f_t(\mathbf{x}) \qquad \text{s.t.} \qquad \mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}. \tag{7}$$

Clearly, the dynamic regret is always larger than the static regret, that is, $\text{Reg}_T^{\text{s}} \leq \text{Reg}_T^{\text{d}}$, since $\sum_{t=1}^{T} f_t(\mathbf{x}^*)$ is always no smaller than $\sum_{t=1}^{T} f_t(\mathbf{x}_t^*)$ given the definitions of $\mathbf{x}^*$ and $\mathbf{x}_t^*$. Hence, a sublinear dynamic regret implies a sublinear static regret, but not vice versa.

Regarding feasibility of decisions generated by an OCO algorithm, the *dynamic fit* is introduced to measure the accumulated violations of constraints, that is,

$$\text{Fit}_T^{\text{d}} := \left\| \left[\sum_{t=1}^{T} \mathbf{g}_t(\mathbf{x}_t) \right]^+ \right\|. \tag{8}$$

Note that the long-term constraint considered here implicitly assumes that the instantaneous constraint violations can be compensated by the later strictly feasible decisions, and thus allows adaptation of fog offloading and computation decisions to the unknown dynamics of IoT user demands.

With the optimality and feasibility metrics in hand, an ideal online algorithm will be the one that achieves both sublinear dynamic regret and sublinear dynamic fit. A sublinear dynamic regret implies "no-regret" relative to the clairvoyant dynamic solution on the long-term average; i.e., $\lim_{T\to\infty} \text{Reg}_T^{\text{d}}/T = 0$; a sublinear dynamic fit indicates that the online strategy is also feasible on average; i.e., $\lim_{T\to\infty} \text{Fit}_T^{\text{d}}/T = 0$. However, the sublinear performance is not achievable if the nature is allowed to behave arbitrarily at each and every slot, even when constraints are time-invariant [26]. Instead, we are after an online

strategy that generates a sequence $\{\mathbf{x}_t\}_{t=1}^T$ ensuring sublinear dynamic regret and fit, under the regularity condition on the nature's behavior.

IV. Thing-Adaptive Saddle-Point Method

In this section, a TAOSP method is developed, and its performance and feasibility are analyzed.

A. Algorithm Development

Consider now the per-slot problem (7), which contains the current objective $f_t(\mathbf{x})$, the current constraint $\mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}$, and a time-invariant feasible set $\mathcal{X}$. With $\boldsymbol{\lambda} \in \mathbb{R}_+^I$ denoting the Lagrange multiplier associated with the time-varying constraint, the online regularized Lagrangian of (7) is given by

$$\mathcal{L}_t(\mathbf{x}, \boldsymbol{\lambda}) := f_t(\mathbf{x}) + \boldsymbol{\lambda}^\top \mathbf{g}_t(\mathbf{x}) - \frac{\theta}{2}\|\boldsymbol{\lambda}\|^2 \tag{9}$$

where $\mathbf{x} \in \mathcal{X}$ remains implicit, and $\theta > 0$ is a preselected constant scaling the ℓ_2-norm that regularizes the constraint violations. The regularizer is tantamount to penalizing the constraint violations in the primal domain; namely,

$$\max_{\boldsymbol{\lambda} \geq \mathbf{0}} \boldsymbol{\lambda}^\top \mathbf{g}_t(\mathbf{x}) - \frac{\theta}{2}\|\boldsymbol{\lambda}\|^2 = \frac{1}{2\theta}\left\|\left[\mathbf{g}_t(\mathbf{x})\right]^+\right\|^2. \tag{10}$$

Based on $\mathcal{L}_t$ in (9), we will develop a novel TAOSP approach, which takes a task-specific gradient descent step in the primal domain followed by a dual ascent step per iteration. Specifically, given the primal iterate $\mathbf{x}_t$ and the dual iterate $\boldsymbol{\lambda}_t$ at slot t, the next decision $\mathbf{x}_{t+1}$ is generated by

$$\mathbf{x}_{t+1} \in \arg\min_{\mathbf{x} \in \mathcal{X}} \nabla_{\mathbf{x}}^\top \mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)(\mathbf{x} - \mathbf{x}_t) + \frac{1}{2\eta}\|\mathbf{x} - \mathbf{x}_t\|^2_{\mathbf{H}_t^{\frac{1}{2}}} \tag{11}$$

where $\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) = \nabla f_t(\mathbf{x}_t) + \nabla^\top \mathbf{g}_t(\mathbf{x}_t)\boldsymbol{\lambda}_t$ is the gradient of $\mathcal{L}_t(\mathbf{x}, \boldsymbol{\lambda}_t)$ with respect to $\mathbf{x}$ at $\mathbf{x} = \mathbf{x}_t$; η is a predefined constant; and the diagonal matrix $\mathbf{H}_t$ accumulates the diagonal entries of the outer product of the gradients as

$$\mathbf{H}_t := \delta \mathbf{I} + \sum_{\tau=1}^{t} \operatorname{diag}\left(\nabla_{\mathbf{x}}\mathcal{L}_\tau(\mathbf{x}_\tau, \boldsymbol{\lambda}_\tau)\nabla_{\mathbf{x}}^\top \mathcal{L}_\tau(\mathbf{x}_\tau, \boldsymbol{\lambda}_\tau)\right) \tag{12}$$

where $\operatorname{diag}(\mathbf{Y})$ is a diagonal matrix with the same diagonal entries of $\mathbf{Y}$, and $\delta > 0$ is a predefined constant. The minimization (11) admits the closed-form solution [25]

$$\mathbf{x}_{t+1} = \mathcal{P}_{\mathcal{X}}^{\mathbf{H}_t^{\frac{1}{2}}}\left(\mathbf{x}_t - \eta\, \mathbf{H}_t^{-\frac{1}{2}} \nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\right) \tag{13}$$

where $\mathcal{P}_{\mathcal{X}}^{\mathbf{H}_t^{1/2}}(\mathbf{y}) := \arg\min_{\mathbf{x} \in \mathcal{X}} (\mathbf{x}-\mathbf{y})^\top \mathbf{H}_t^{1/2}(\mathbf{x}-\mathbf{y})$. Intuitively, for the coordinates of $\mathbf{x}_t$ with large accumulated gradients, the associated stepsize will be scaled down, and for the ones with small accumulated gradients, their stepsizes will be enlarged relative to that of other coordinates.

The dual update takes the online gradient ascent form

$$\boldsymbol{\lambda}_{t+1} = \left[\boldsymbol{\lambda}_t + \mu \nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\right]^+ \tag{14}$$

where μ is a positive stepsize, and $\nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) = \mathbf{g}_t(\mathbf{x}_t) - \theta\boldsymbol{\lambda}_t$ is the gradient of $\mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda})$ with respect to $\boldsymbol{\lambda}$ at $\boldsymbol{\lambda} = \boldsymbol{\lambda}_t$. The choice of parameters $(\theta, \delta, \eta, \mu)$ that guarantees sublinear performance bounds will be discussed in Section IV-B.

Remark 2: We term (13) and (14) as an *adaptive* (or thing-adaptive) online saddle-point approach, because the primal update (13) can automatically adjust its *matrix stepsize* according to the steepness of the online Lagrangian along each direction, which is approximated by the magnitude of each gradient coordinate corresponding to one thing in IoT applications. The adaptive matrix stepsize can be regarded as an inexpensive approximation of the Hessian matrix used in the online Newton method [24], which has well-documented performance in, e.g., deep learning tasks [25]. Here, we leverage the adaptive matrix stepsize for OCO with long-term constraints. Using fog computing as a paradigm, we will show that our TAOSP algorithm markedly improves performance when the underlying IoT tasks are heterogeneous, meaning that the resultant gradients have *distinct* orders of magnitude over different coordinates.

B. Dynamic Regret and Fit Analysis

Before formally analyzing the dynamic regret and fit for TAOSP, we make the following assumptions.

A1: For every t, the functions $f_t(\mathbf{x})$ and $\mathbf{g}_t(\mathbf{x})$ are convex.

A2: Functions $f_t(\mathbf{x})$ and $\mathbf{g}_t(\mathbf{x})$ have bounded gradients; i.e., $\max\{|\nabla^i f_t(\mathbf{x})|, \|\nabla^i \mathbf{g}_t(\mathbf{x})\|_\infty\} \leq G_i, \forall \mathbf{x} \in \mathcal{X}$, where $\nabla^i f_t(\mathbf{x})$ and $\nabla^i \mathbf{g}_t(\mathbf{x})$ are with respect to the ith entry of $\mathbf{x}$, and $\sum_{i=1}^I G_i = G$.

A3: The radius of the convex feasible set $\mathcal{X}$ is bounded; i.e., $\|\mathbf{x} - \mathbf{y}\| \leq R,\ \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$.

Regarding these assumptions, A1 and A2 require the convexity and Lipschitz continuity of the objective and constraint functions, while A3 restricts the feasible set to be bounded. Note that A1–A3 are common in OCO with constraints [24], [28]. Next, we highlight the critical insights and the key lemmas leading to the final performance bounds, but defer the detailed derivations to Appendix A.

Under these assumptions, the regularized Lagrangian in (9) is convex with respect to the primal variable, and concave with respect to the dual variable, and thus it follows that [see (46a) and (46b)]:

$$\begin{aligned}\mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}) - \mathcal{L}_t(\mathbf{x}, \boldsymbol{\lambda}_t) \leq\ & (\mathbf{x}_t - \mathbf{x})^\top \nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) \\ & + (\boldsymbol{\lambda} - \boldsymbol{\lambda}_t)^\top \nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t).\end{aligned} \tag{15}$$

On the other hand, plugging $\mathbf{x} = \mathbf{x}_t^*$ into (15), and summing up over $t = 1, 2, \ldots, T$, it turns out that [see (51)–(58)]

$$\operatorname{Reg}_T^{\mathrm{d}} + \frac{1}{2\theta}\left(\operatorname{Fit}_T^{\mathrm{d}}\right)^2 \lessapprox \sum_{t=1}^{T}\left(\mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}) - \mathcal{L}_t\left(\mathbf{x}_t^*, \boldsymbol{\lambda}_t\right)\right) \tag{16}$$

where θ is the regularization coefficient, and "$\lessapprox$" means the inequality "$\leq$" holds under some technical conditions that will be specified in the following lemmas. Combining (15) with (16), if we can upper bound the summation in the RHS of (15) with a proper sublinear order of T, and appropriately choose θ, we can eventually obtain the desired dynamic regret and fit.

With the insights gained so far, we first derive a set of bounds on the RHS of (15).

Lemma 1: Suppose A1–A3 are satisfied, and consider the TAOSP recursion (13) and (14). For any $\mathbf{x} \in \mathcal{X}$, it holds that

$$(\mathbf{x}_t - \mathbf{x})^\top \nabla_{\mathbf{x}} \mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) \leq \frac{1}{2\eta}\|\mathbf{x} - \mathbf{x}_t\|^2_{\mathbf{H}_t^{\frac{1}{2}}} - \frac{1}{2\eta}\|\mathbf{x} - \mathbf{x}_{t+1}\|^2_{\mathbf{H}_t^{\frac{1}{2}}} + \frac{\eta}{2}\|\nabla_{\mathbf{x}} \mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\|^2_{\mathbf{H}_t^{-\frac{1}{2}}} \tag{17}$$

where η and $\mathbf{H}_t$ are defined in (13). The corresponding bound for the dual variables is

$$(\boldsymbol{\lambda} - \boldsymbol{\lambda}_t)^\top \nabla_{\boldsymbol{\lambda}} \mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) \leq \frac{1}{2\mu}\|\boldsymbol{\lambda} - \boldsymbol{\lambda}_t\|^2 - \frac{1}{2\mu}\|\boldsymbol{\lambda} - \boldsymbol{\lambda}_{t+1}\|^2 + \frac{\mu}{2}\|\nabla_{\boldsymbol{\lambda}} \mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\|^2. \tag{18}$$

Proof: See Appendix B. ∎

Lemma 1 reveals that the bounds for the RHS of (15) depend on the difference of two consecutive distances between the primal–dual iterates and a pair of fixed points, as well as the magnitudes of the primal–dual gradients. While the difference of two consecutive distances can be controlled by choosing primal and dual stepsizes, the magnitudes of gradients will be analyzed in the following lemma.

Lemma 2: Under the same conditions as those in Lemma 1, the gradients with respect to the primal variable can be bounded by

$$\frac{\eta}{2}\sum_{t=1}^{T}\|\nabla_{\mathbf{x}} \mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\|^2_{\mathbf{H}_t^{-\frac{1}{2}}} \leq \eta \sum_{i=1}^{I}\left\|\nabla^i_{\mathbf{x}} \mathcal{L}_{1:T}(\mathbf{x}_{1:T}, \boldsymbol{\lambda}_{1:T})\right\| \leq \eta G\sqrt{(I+1)T} + \eta G\sqrt{(I+1)\textstyle\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2} \tag{19}$$

where constant G is defined in A2, and the ith entry of the stacked gradients is $\nabla^i_{\mathbf{x}} \mathcal{L}_{1:T}(\mathbf{x}_{1:T}, \boldsymbol{\lambda}_{1:T}) := [\nabla^i_{\mathbf{x}} \mathcal{L}_1(\mathbf{x}_1, \boldsymbol{\lambda}_1), \ldots, \nabla^i_{\mathbf{x}} \mathcal{L}_T(\mathbf{x}_T, \boldsymbol{\lambda}_T)]^\top$. In addition, the magnitude of dual gradients can be bounded by

$$\frac{\mu}{2}\sum_{t=1}^{T}\|\nabla_{\boldsymbol{\lambda}} \mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\|^2 \leq \mu R^2 G^2 T + \mu\theta^2 \sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2 \tag{20}$$

where constant R is defined in A3.

Proof: See Appendix C. ∎

Using Lemmas 1 and 2, we can bound the dynamic regret and dynamic fit as follows.

Theorem 1: Under A1–A3, if we choose the stepsizes $\eta = R/\sqrt{2}$, $\mu = T^{-(5/8)}/(RG)$, and the parameters $\delta = \mathcal{O}(1)$ and $\theta = RGT^{-(1/8)}$, and further initialize the dual variable to satisfy $\|\boldsymbol{\lambda}_1\| = 4\sqrt{(I+1)}T^{(1/8)}$ with I denoting the number of constraints, the dynamic regret is bounded by

$$\mathrm{Reg}^{\mathrm{d}}_T \leq \epsilon_2 T^{\frac{5}{8}} + \epsilon_1 \mathbb{V}\big(\mathbf{x}^*_{1:T}\big) + \epsilon_0 = \mathcal{O}\Big(T^{\frac{7}{8}}\mathbb{V}\big(\mathbf{x}^*_{1:T}\big)\Big) \tag{21}$$

where the constants are $\epsilon_0 = \mathcal{O}(T^{(1/2)})$, $\epsilon_1 = \mathcal{O}(T^{(7/8)})$, and $\epsilon_2 = \mathcal{O}(T^{(1/4)})$; and, $\mathbb{V}(\mathbf{x}^*_{1:T})$ is the accumulated variation of the per-slot minimizers $\mathbf{x}^*_t$ defined as $\mathbb{V}(\mathbf{x}^*_{1:T}) := \sum_{t=1}^{T}\|\mathbf{x}^*_t - \mathbf{x}^*_{t-1}\|$. Accordingly, the dynamic fit of TAOSP is bounded by

$$\mathrm{Fit}^{\mathrm{d}}_T = \mathcal{O}\left(\max\left\{T^{\frac{15}{16}}, T^{\frac{7}{8}}\sqrt{\mathbb{V}\big(\mathbf{x}^*_{1:T}\big)}\right\}\right). \tag{22}$$

Proof: See Appendix D. ∎

Algorithm 1 TAOSP for Mobile-Edge Computation Offloading

1: **Initialize:** primal iterates $\{x^{nk}_0\}$ and $\{\chi^n_0\}$, dual iterate $\boldsymbol{\lambda}_0$, parameter θ, and proper stepsizes η and μ.
2: **for** $t = 1, 2 \ldots$ **do**
3: fog nodes perform offloading to the cloud and neighbor edges via (23a)-(23b) and locally process via (23c).
4: fog nodes observe the aggregate network delay and workload arrivals from IoT devices to update (23g).
5: **end for**

Theorem 1 asserts that TAOSP's dynamic regret and fit are upper bounded by some constants depending on the time horizon and the accumulated variation of per-slot minimizers. Specifically, without *a priori* knowledge of the accumulated per-slot minimizer variation, the dynamic regret and fit of TAOSP are sublinear provided that $\mathbb{V}(\mathbf{x}^*_{1:T}) = \mathbf{o}(T^{(1/8)})$. When the order of $\mathbb{V}(\mathbf{x}^*_{1:T})$ is known *a priori*, tighter regret and fit bounds can be effected by adjusting stepsizes accordingly (see [30]).

Remark 3: TAOSP improves upon the proposed algorithm in our precursor [30] in terms of *fewer* assumptions, *lower* computational complexity, and *task-specific* learning rates tailored for heterogeneous IoT setups. The desired algorithmic merits will be demonstrated next in the fog computing context.

V. Online Fog Computing Tests

In this section, we tackle the fog computing task within our novel OCO framework, and present numerical experiments.

A. TAOSP Solver for Fog Computing

TAOSP can be leveraged to solve (4) in an *online* fashion, with provable performance and feasibility guarantees. Specifically, the primal update (13) boils down to a simple closed-form gradient update amenable to decentralized implementation, which yields the cloud offloading amount as

$$\chi^n_t = \left[\chi^n_{t-1} - \eta\Big(H^{\chi^n}_{t-1}\Big)^{-\frac{1}{2}}\big(\nabla_{\chi^n} f_{t-1}(\mathbf{x}_{t-1}) - \lambda^n_{t-1}\big)\right]_0^{\bar{\chi}^n} \tag{23a}$$

the offloading amount from fog node n to k as

$$x^{nk}_t = \left[x^{nk}_{t-1} - \eta\Big(H^{x^{nk}}_{t-1}\Big)^{-\frac{1}{2}}\Big(\nabla_{x^{nk}} f_{t-1}(\mathbf{x}_{t-1}) - \lambda^n_{t-1} + \lambda^k_{t-1}\Big)\right]_0^{\bar{x}^{nk}} \tag{23b}$$

and the local processing decision at edge n as

$$x^{nn}_t = \left[x^{nn}_{t-1} - \eta\Big(H^{x^{nn}}_{t-1}\Big)^{-\frac{1}{2}}\big(\nabla_{x^{nn}} f_{t-1}(\mathbf{x}_{t-1}) - \lambda^n_{t-1}\big)\right]_0^{\bar{x}^{nn}} \tag{23c}$$

where the adaptive scaling coefficients are found as

$$H^{\chi^n}_{t-1} = \delta + \sum_{\tau=1}^{t-1}\big(\nabla_{\chi^n} f_\tau(\mathbf{x}_\tau) - \lambda^n_\tau\big)^2 \tag{23d}$$

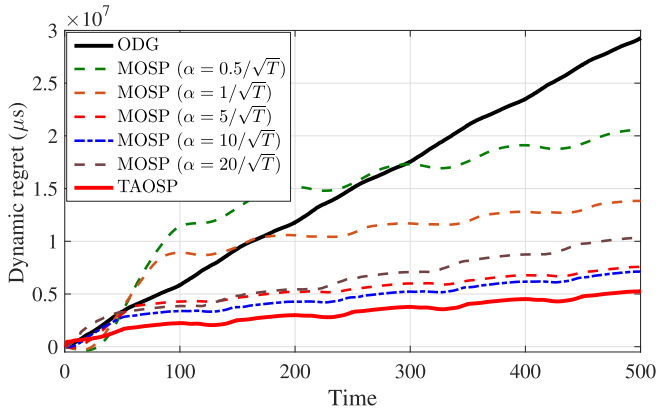


Fig. 3. Comparison of dynamic regret for fog computing tasks.

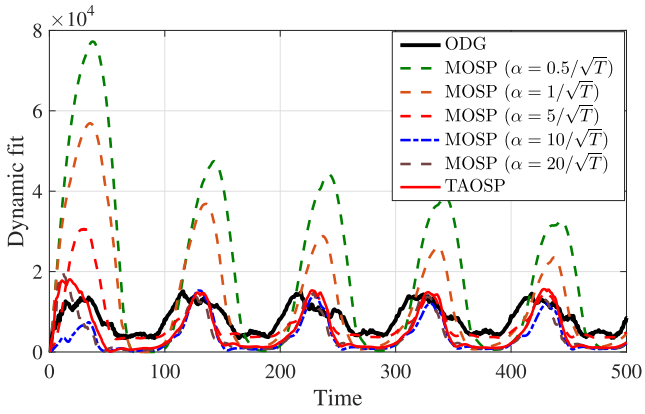


Fig. 4. Comparison of dynamic fit for fog computing tasks.

and

$$H_{t-1}^{x^{nn}} = \delta + \sum_{\tau=1}^{t-1} \left(\nabla_{x^{nn}} f_\tau(\mathbf{x}_\tau) - \lambda_\tau^n\right)^2 \tag{23e}$$

and likewise for

$$H_{t-1}^{x^{nk}} = \delta + \sum_{\tau=1}^{t-1} \left(\nabla_{x^{nk}} f_\tau(\mathbf{x}_\tau) - \lambda_\tau^n + \lambda_\tau^k\right)^2. \tag{23f}$$

These coefficients learn the magnitude of each coordinate, and adjust the learning rates associated with fog nodes on-the-fly. Depending on the specific delay functions (2) and (3), the involved gradients in (23a)–(23f) can be readily computed.

The dual variable update (14) at each fog node n reduces to

$$\lambda_t^n = \left[(1-\mu\theta)\lambda_{t-1}^n + \mu\left(d_{t-1}^n + \sum_{k\in\mathcal{N}_n^{\text{in}}} x_{t-1}^{kn} - \sum_{k\in\mathcal{N}_n^{\text{out}}} x_{t-1}^{nk} - \chi_{t-1}^n - x_{t-1}^{nn}\right)\right]^+ \tag{23g}$$

where μ and θ are chosen according to Theorem 1.

Intuitively, to guarantee long-term feasibility, the dual variable increases (increasing penalty) when there is instantaneous service residual (constraint violation), and decreases when over-serving occurs in the mobile-edge computing systems. TAOSP for online fog computing tasks is summarized in Algorithm 1.

B. Numerical Experiments

Consider the fog computing task in (4) with $N = 20$ fog nodes, and one cloud center. For both the aggregate delay function (2) and the maximum delay function (3), we consider their summands as follows. The communication delay of cloud offloading is $c_t^n(\chi_t^n) := p_t^n(\chi_t^n)^2 + q_t^n\chi_t^n$ (μs), where $p_t^n = \sin(\pi t/50) + v_t^n$ with v_t^n uniformly distributed over $[1, 3]$, and q_t^n uniformly distributed over $[1, 10]$; the communication delay of local offloading is $c_t^{nk}(x_t^{nk}) := l^{nk}x_t^{nk}$ (μs), and the local computation delay function is $h_t^n(x_t^{nn}) := l^{nn}(x_t^{nn})^2$ (μs), where the coefficients $\{l^{nk}\}$ are generated as follows. With the local communication limits $\bar{x}^{nk} = 10$, $n \in [1, 5]\bigcup[11, 20]$ and $\bar{x}^{nk} = 100$, $n \in [6, 10]$ as well as the fog computation limits $\bar{x}^{nn} = 100$, $n \in [1, 5]\bigcup[11, 20]$ and $\bar{x}^{nn} = 1000$, $n \in [6, 10]$, we set the delay coefficients as $l^{nk} = 50/\bar{x}^{nk}$ and $l^{nn} = 50/\bar{x}^{nn}$. These choices of coefficients ensure that the per-unit local offloading or computation delay is inversely proportional to the communication link or the fog server capacity. In addition, the edge-cloud offloading limits $\{\bar{\chi}^n\}$ are uniformly distributed over $[100, 200]$, and the data arrival rate d_t^n is generated according to $d_t^n = \sin(\pi t/50) + v_t^n$, with v_t^n uniformly distributed over $[45, 55]$. Here the scales of p_t^n, q_t^n, and d_t^n vary, mimicking the heterogeneity of IoT, while their periods follow the periodic patterns of human activities in IoT.

1) Benchmarks: TAOSP is benchmarked by the *nonadaptive* MOSP method in [30] with a fixed primal stepsize α, and the popular stochastic dual gradient approach in [32] and [37]. Since the stochastic gradient updates require noncausal knowledge of $f_t(\mathbf{x})$ and $\{d_t^n, \forall n\}$ to decide $\mathbf{x}_t$, we modify them in this OCO setting by using the information at slot $t-1$ instead. We refer to this method as online dual gradient (ODG). The parameters of all compared methods are tuned for the best performance. Simulated tests were averaged over 50 Monte Carlo realizations.

2) Dynamic Regret and Fit in Basic Setup: With the goal of minimizing the aggregate delay in (2), the dynamic regret [see (6)] is first compared for TAOSP, ODG, and MOSP under different stepsizes in Fig. 3. In this fog computing test, the regret is the difference between the delay of TAOSP and the minimal achievable delay [see (7)], and it is accumulated over hundreds of iterations and over all fog nodes. Clearly, the regret of TAOSP grows much slower than that of ODG. Although under different stepsizes the regret of MOSP has a growing rate similar to that of TAOSP, a constant gap can be still observed between their regrets. Regarding the dynamic fit [see (8)], Fig. 4 demonstrates that MOSP with larger stepsize η exhibits lower fit than that of ODG, and similar to the dynamic fit of TAOSP. In such a case however, TAOSP still enjoys lower regret than that of MOSP (see Fig. 3), thanks to its flexibility of using adaptive matrix stepsizes. Evidently, TAOSP performs the best in this simulated setting since it has a much smaller regret on minimizing network delay while its dynamic fit is smaller than that of ODG, and similar to that of MOSP with larger stepsizes.

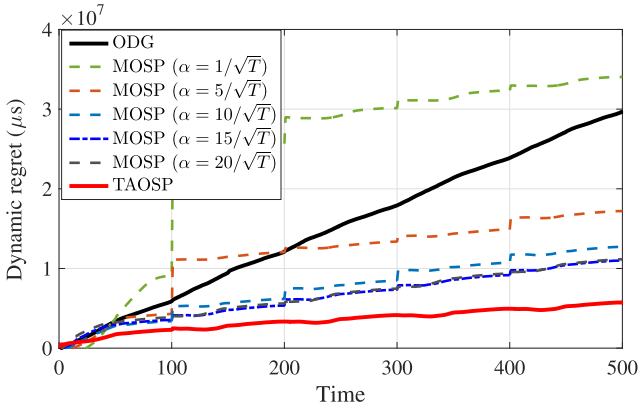


Fig. 5. Dynamic regret under two malicious fog nodes.

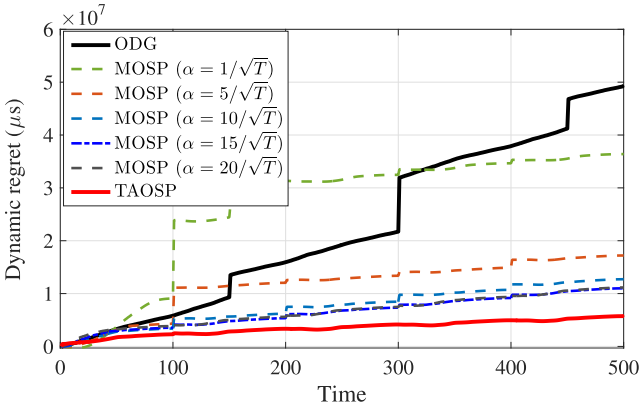


Fig. 6. Dynamic regret under three malicious fog nodes.

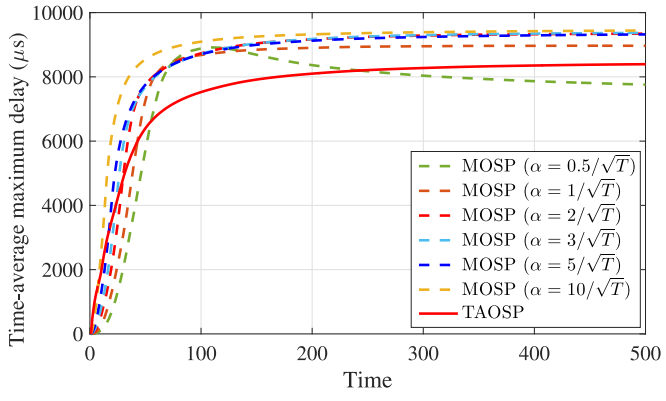


Fig. 7. Time-average maximum delay.

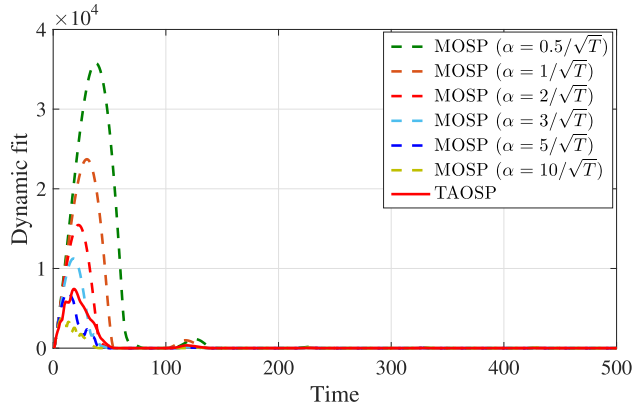


Fig. 8. Dynamic fit under maximum delay criterion.

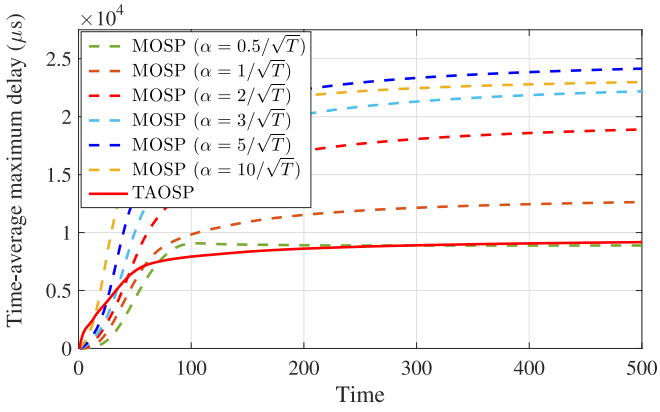


Fig. 9. Time-average maximum delay in time-varying networks.

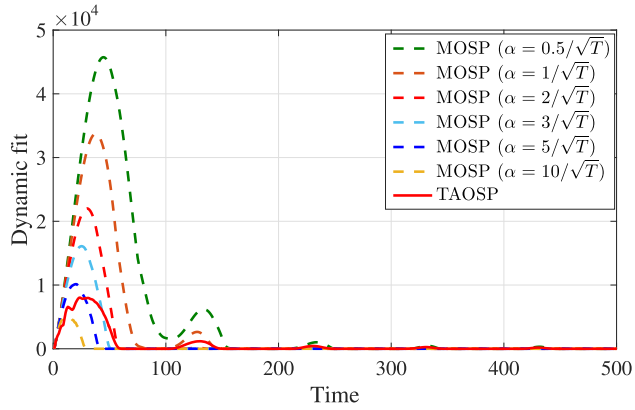


Fig. 10. Dynamic fit under maximum delay criterion in time-varying networks.

3) Effect of Cyber Attacks: The performance of TAOSP is further tested in the presence of cyber attacks, in which case malicious communication links strategically impede offloading from fog nodes and the cloud center, and lead to unexpected communication delays. In the first test, the unexpected communication delays are simulated by perturbing the coefficients in $c_t^n(\chi_t^n)$ to be $p_t^5 = 1500$ for $t \mod 100 = 1$, and $p_t^{10} = 500$ for $t \mod 50 = 1$, causing the dynamic regret shown in Fig. 5. Clearly, the performance gain of TAOSP is already observable. When more malicious communication links are involved, e.g., $p_t^{15} = 1500$ for $t \mod 150 = 1$, the performance gain of TAOSP becomes more pronounced; see Fig. 6. The curvatures of dynamic fit in these two cases are almost the same as that in Fig. 4, and thus they are omitted. The desired performance comes from its matrix learning rate that mitigates the effect of cyber attacks at one fog node on the other fog nodes.

4) Maximum Delay Criterion: TAOSP is further tested using the maximum delay criterion defined in (3); see the regret and fit in Figs. 7 and 8 for a static fog network, and in Figs. 9 and 10 for a time-varying one. For the time-varying fog network, the offloading limits between fog nodes switch between cases 1 and 2 in Fig. 11 at every other slot. Note that under this nonsmooth objective (3), OGD suffers from significantly high computational complexity due to the lack of a closed-form update, and thus it is not simulated here.

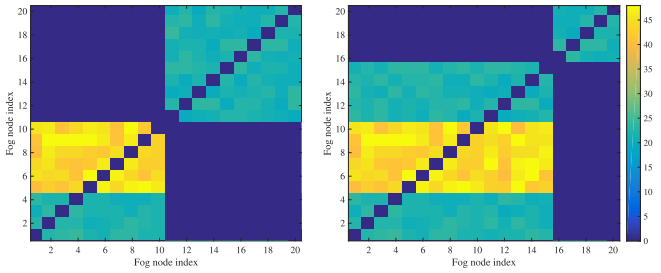


Fig. 11. Offloading limits $\{\bar{x}^{nk}\}$ among nearby fog nodes, where different color represents the different limit. (a) Case 1: fog nodes 1–10 belong to the same fog cluster and nodes 11–20 belong to the other one. (b) Case 2: nodes 1–15 belong to the same fog cluster and nodes 16–20 belong to the other one.

Aligned with its performance in previous experiments, Fig. 7 shows that the maximum delay of TAOSP is also lower than that of MOSP in most cases, and similar to MOSP with best tuned stepsize $\alpha = 0.5/\sqrt{T}$. However, Fig. 8 demonstrates that the dynamic fit of MOSP with the small stepsize is much larger than that of TAOSP in this setting. The performance gap between TAOSP and MOSP with small stepsize can be also observed in Figs. 9 and 10 with a time-varying fog network. It is clearly shown that TAOSP can still achieve competitive performance under the maximum delay criterion, in both static and varying networks.

VI. Conclusion

Fog computation offloading has been formulated as an online learning task with both adversarial costs and constraints. Different from existing OCO works, the focus is on a broader setting where part of the constraints is revealed after taking actions, they can be tolerable to instantaneous violations, but have to be satisfied on average. Accounting for the extreme heterogeneity of IoT applications, a thing-adaptive saddle-point approach termed TAOSP was introduced to learn the optimal fog-computing actions with task-specific learning rates. It has been shown that TAOSP simultaneously yields sublinear dynamic regret and fit, provided that the dynamic solutions vary slowly over time. The novel TAOSP algorithm and its dynamic regret analysis endow the fog computing tasks with efficient online implementation, as well as guaranteed IoT user experience in nonstationary dynamic environments.

To overcome the limitations of TAOSP, several future directions can be pursued. While the current model assumes that the relation between the amount of transmitted data versus the needed computation is uniform among all the tasks, its generalization to the task-specific case is of great importance. Such a generalization requires incorporating multiple long-term constraints, e.g., (1) per task and per node. Dealing with nonconvex delay functions is also in our future research agenda.

Appendix A
Supporting Lemmas

We first establish some key lemmas and propositions, and then present the proofs of Lemmas 1–2 and Theorem 1.

Lemma 3: For the TAOSP recursion (13) and (14), the differential squared Mahalanobis distance can be bounded by

$$\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\left\|\mathbf{x}_t-\mathbf{x}_{t-1}^*\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\leq 2R\sigma_{\max}\left(\mathbf{H}_t^{\frac{1}{2}}\right)\left\|\mathbf{x}_t^*-\mathbf{x}_{t-1}^*\right\| \tag{24}$$

where R is defined in A3, and $\sigma_{\max}(\mathbf{H}_t^{(1/2)})$ is the maximum eigenvalue of $\mathbf{H}_t^{(1/2)}$. The following distance is then bounded by:

$$\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_{t+1}^{\frac{1}{2}}}-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\leq R^2\mathrm{tr}\left(\mathbf{H}_{t+1}^{\frac{1}{2}}-\mathbf{H}_t^{\frac{1}{2}}\right). \tag{25}$$

Proof: For the first part of the lemma, it follows that:

$$\begin{aligned}&\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\left\|\mathbf{x}_t-\mathbf{x}_{t-1}^*\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\\&\quad=\left(\mathbf{x}_t^*-\mathbf{x}_t+\mathbf{x}_{t-1}^*-\mathbf{x}_t\right)^\top\mathbf{H}_t^{\frac{1}{2}}(\mathbf{x}_t^*-\mathbf{x}_{t-1}^*)\\&\quad\overset{(a)}{\leq}\left\|\mathbf{x}_t^*-\mathbf{x}_t+\mathbf{x}_{t-1}^*-\mathbf{x}_t\right\|\cdot\left\|\mathbf{H}_t^{\frac{1}{2}}(\mathbf{x}_t^*-\mathbf{x}_{t-1}^*)\right\|\\&\quad\overset{(b)}{\leq}2R\sigma_{\max}(\mathbf{H}_t^{\frac{1}{2}})\left\|\mathbf{x}_t^*-\mathbf{x}_{t-1}^*\right\|\end{aligned} \tag{26}$$

which (a) follows from Cauchy–Schwartz inequality, and (b) uses the definitions of R, and $\sigma_{\max}(\mathbf{H}_t^{(1/2)})$.

For the second part of the lemma, we have that

$$\begin{aligned}&\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_{t+1}^{\frac{1}{2}}}-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\\&\quad=\left(\mathbf{x}_t^*-\mathbf{x}_{t+1}\right)^\top\left(\mathbf{H}_{t+1}^{\frac{1}{2}}-\mathbf{H}_t^{\frac{1}{2}}\right)\left(\mathbf{x}_t^*-\mathbf{x}_{t+1}\right)\\&\quad\leq\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_\infty\mathrm{tr}\left(\mathbf{H}_{t+1}^{\frac{1}{2}}-\mathbf{H}_t^{\frac{1}{2}}\right)\\&\quad\leq\max_{t\leq T}\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_\infty\mathrm{tr}\left(\mathbf{H}_{t+1}^{\frac{1}{2}}-\mathbf{H}_t^{\frac{1}{2}}\right)\end{aligned} \tag{27}$$

which completes the proof as $\max_{t\leq T}\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\|_\infty\leq R$. ■

Lemma 4: For the TAOSP recursion (13) and (14), the accumulated squared Mahalanobis distance can be bounded by

$$\begin{aligned}\sum_{t=1}^T\left(\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\right)&\leq R^2G\sqrt{(I+1)T}\\&+R^2G\sqrt{(I+1)\textstyle\sum_{t=1}^T\|\boldsymbol{\lambda}_t\|^2}+2R\sigma_{\max}\left(\mathbf{H}_T^{\frac{1}{2}}\right)\mathbb{V}\left(\mathbf{x}_{1:T}^*\right)+\delta R^2\end{aligned} \tag{28}$$

where the constants G and R are defined in A2 and A3.

Proof: Adding and substracting $\|\mathbf{x}_t-\mathbf{x}_{t-1}^*\|^2_{\mathbf{H}_t^{(1/2)}}$ into the targeted term $\|\mathbf{x}_t^*-\mathbf{x}_t\|^2_{\mathbf{H}_t^{(1/2)}}-\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\|^2_{\mathbf{H}_t^{(1/2)}}$, we have that

$$\begin{aligned}&\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\\&\quad=\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\left\|\mathbf{x}_t-\mathbf{x}_{t-1}^*\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}+\left\|\mathbf{x}_t-\mathbf{x}_{t-1}^*\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\\&\qquad-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\\&\quad\overset{(a)}{\leq}2R\sigma_{\max}(\mathbf{H}_t^{\frac{1}{2}})\left\|\mathbf{x}_t^*-\mathbf{x}_{t-1}^*\right\|+\left\|\mathbf{x}_t-\mathbf{x}_{t-1}^*\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\\&\qquad-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\end{aligned} \tag{29}$$

where inequality (a) comes from (24) in Lemma 3, R is defined in A3, and $\sigma_{\max}(\mathbf{H}_t^{(1/2)})$ is the maximum eigenvalue of $\mathbf{H}_t^{(1/2)}$.

Summing up (29) over $t = 1, 2, \ldots, T$, we have that

$$\begin{aligned}
&\sum_{t=1}^{T}\left(\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\right)\\
&\overset{(b)}{\leq}\sum_{t=1}^{T-1}\left(\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_{t+1}^{\frac{1}{2}}}-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\right)\\
&\quad+2R\sigma_{\max}\left(\mathbf{H}_T^{\frac{1}{2}}\right)\mathbb{V}\left(\mathbf{x}_{1:T}^*\right)+\left\|\mathbf{x}_0^*-\mathbf{x}_1\right\|^2_{\mathbf{H}_1^{\frac{1}{2}}}\\
&\overset{(c)}{\leq}\sum_{t=1}^{T-1}R^2\mathrm{tr}\left(\mathbf{H}_{t+1}^{\frac{1}{2}}-\mathbf{H}_t^{\frac{1}{2}}\right)+2R\sigma_{\max}\left(\mathbf{H}_T^{\frac{1}{2}}\right)\mathbb{V}\left(\mathbf{x}_{1:T}^*\right)+\delta R^2\\
&\leq R^2\mathrm{tr}\left(\mathbf{H}_{T+1}^{\frac{1}{2}}-\mathbf{H}_1^{\frac{1}{2}}\right)+2R\sigma_{\max}\left(\mathbf{H}_T^{\frac{1}{2}}\right)\mathbb{V}\left(\mathbf{x}_{1:T}^*\right)+\delta R^2\\
&=R^2\sum_{i=1}^{I}\left\|\nabla_{\mathbf{x}}^i\mathcal{L}_{1:T}\right\|+2R\sigma_{\max}\left(\mathbf{H}_T^{\frac{1}{2}}\right)\mathbb{V}\left(\mathbf{x}_{1:T}^*\right)+\delta R^2 \qquad (30)
\end{aligned}$$

where (b) holds since $\sigma_{\max}(\mathbf{H}_t^{(1/2)})$ is nondecreasing due to the matrix update (12), and $\mathbb{V}(\mathbf{x}_{1:T}^*)$ is defined in Theorem 1; and (c) uses (25) in Lemma 3, $\mathbf{H}_1 = \delta\mathbf{I}$, and the definition of R.

Using (19) to further expand the RHS of (30), it follows that:

$$\begin{aligned}
&\sum_{t=1}^{T}\left(\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\right)\leq R^2G\sqrt{(I+1)T}\\
&\quad+R^2G\sqrt{(I+1)\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2}+2R\sigma_{\max}\left(\mathbf{H}_T^{\frac{1}{2}}\right)\mathbb{V}\left(\mathbf{x}_{1:T}^*\right)+\delta R^2
\end{aligned} \qquad (31)$$

from which we complete the proof. ∎

Proposition 1: If we choose $\mu = c_\mu T^{-(5/8)}$ and $\theta = c_\theta T^{-(1/8)}$ with constants $c_\mu > 0$ and $c_\theta > 0$, for a sufficiently large T, there exists $c_\lambda > (2c_0/c_\theta)$ such that for $\rho \geq c_\lambda T^{(1/8)}$, it holds that

$$\left(\mu\theta^2-\frac{\theta}{2}\right)\rho^2+c_0\rho\leq 0 \qquad (32)$$

where $c_0 > 0$ is a given constant.

Proof: Since $\rho > 0$, it suffices to show $(\mu\theta^2 - (\theta/2))\rho + c_0 \leq 0$. Choosing $\mu = c_\mu T^{-(5/8)}$ and $\theta = c_\theta T^{-(1/8)}$, we have

$$\left(\mu\theta^2-\frac{\theta}{2}\right)\rho+c_0=\left(c_\mu c_\theta^2T^{-\frac{7}{8}}-\frac{c_\theta}{2}T^{-\frac{1}{8}}\right)\rho+c_0 \qquad (33a)$$

$$\overset{(a)}{\leq}\left(c_\mu c_\theta^2T^{-\frac{7}{8}}-\frac{c_\theta}{2}T^{-\frac{1}{8}}\right)c_\lambda T^{\frac{1}{8}}+c_0\overset{(b)}{\leq}0 \qquad (33b)$$

where (a) holds whenever $T > (2c_\mu c_\theta)^{(4/3)}$ so that $c_\mu c_\theta^2 T^{-(7/8)} - (c_\theta/2)T^{-(1/8)} < 0$ and (33a) is nonincreasing with respect to $\rho \geq c_\lambda T^{(1/8)}$; and (b) holds when $c_\lambda > 2c_0/c_\theta$ and T satisfies that

$$T\geq\max\left\{\left(\frac{c_\mu c_\lambda c_\theta^2}{c_\theta c_\lambda/2-c_0}\right)^{\frac{4}{3}},(2c_\mu c_\theta)^{\frac{4}{3}}\right\}=\mathcal{O}(1) \qquad (34)$$

from which the proof is complete. ∎

Proposition 2: For the recursion (13) and (14), if the dual variable is initialized by $\|\boldsymbol{\lambda}_1\| = \mathcal{O}(T^{(1/8)})$ and the stepsize is chosen as $\mu = (1/RG)T^{-(5/8)}$, the maximum eigenvalue of the diagonal matrix $\mathbf{H}_t^{(1/2)}$ can be bounded by $\sigma_{\max}(\mathbf{H}_t^{(1/2)}) \leq \sigma(\mathbf{H}) := \mathcal{O}(T^{(7/8)})$.

Proof: The ith entry of the diagonal matrix $\mathbf{H}_t$ is given by

$$\begin{aligned}
\mathbf{H}_t^{ii}&=\delta+\sum_{\tau=1}^{t}\left(\nabla_{\mathbf{x}}^if_\tau(\mathbf{x}_\tau)+\sum_{j=1}^{I}\lambda_\tau^j\nabla_{\mathbf{x}}^ig_\tau^j(\mathbf{x}_\tau)\right)^2\\
&\overset{(a)}{\leq}\delta+\sum_{\tau=1}^{t}(I+1)\left(G_i^2+\sum_{j=1}^{I}\left(\lambda_\tau^j\right)^2G_i^2\right)\\
&=\delta+(I+1)G_i^2t+(I+1)G_i^2\sum_{\tau=1}^{t}\|\boldsymbol{\lambda}_\tau\|^2 \qquad (35)
\end{aligned}$$

where (a) uses $(a_1 + \cdots + a_n)^2 \leq n(a_1^2 + \cdots + a_n^2)$ and the Lipschitz condition in (A2).

From (35), it follows that $\sigma_{\max}(\mathbf{H}_t^{(1/2)}) \leq \max_i(\mathbf{H}_T^{ii})^{(1/2)}$ since $\mathbf{H}_t^{ii}$ is nondecreasing over t, and we have:

$$\begin{aligned}
\max_i\left(\mathbf{H}_T^{ii}\right)^{\frac{1}{2}}&\leq\left(\delta+(I+1)G^2T+(I+1)G^2\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2\right)^{\frac{1}{2}}\\
&=\mathcal{O}\left(\sqrt{T}\|\bar{\boldsymbol{\lambda}}\|\right) \qquad (36)
\end{aligned}$$

where we simply used $G_i^2 \leq G^2$, and $\|\bar{\boldsymbol{\lambda}}\| := \max_t \|\boldsymbol{\lambda}_t\|$.

For $\boldsymbol{\lambda}_{t+1} = [\boldsymbol{\lambda}_t + \mu\mathbf{g}_t(\mathbf{x}_t) - \mu\theta\boldsymbol{\lambda}_t]^+$, since $\mu\theta\boldsymbol{\lambda}_t \geq \mathbf{0}$, it can be shown using induction that the sequence $\{\|\boldsymbol{\lambda}_t\|\}$ is upper bounded by the sequence $\{\|\hat{\boldsymbol{\lambda}}_t\|\}$ generated by the recursion $\hat{\boldsymbol{\lambda}}_{t+1} = [\hat{\boldsymbol{\lambda}}_t + \mu\mathbf{g}_t(\mathbf{x}_t)]^+$ with $\hat{\boldsymbol{\lambda}}_1 = \boldsymbol{\lambda}_1$, which gives rise to

$$\begin{aligned}
\|\bar{\boldsymbol{\lambda}}\|&\leq\|\hat{\boldsymbol{\lambda}}_T-\hat{\boldsymbol{\lambda}}_{T-1}\|+\ldots+\|\hat{\boldsymbol{\lambda}}_2-\boldsymbol{\lambda}_1\|+\|\boldsymbol{\lambda}_1\|\\
&\leq\sum_{t=1}^{T-1}\mu\|\mathbf{g}_t(\mathbf{x}_t)\|+\|\boldsymbol{\lambda}_1\|\overset{(b)}{\leq}T\mu RG+\|\boldsymbol{\lambda}_1\|\\
&=T^{\frac{3}{8}}+c_\lambda T^{\frac{1}{8}}=\mathcal{O}\left(T^{\frac{3}{8}}\right) \qquad (37)
\end{aligned}$$

where (b) uses the definitions of G and R in A2 and A3, and in (c), the parameters are chosen as $\mu = [1/(RG)]T^{-(5/8)}$ and $\|\boldsymbol{\lambda}_1\| = c_\lambda T^{(1/8)}$. Plugging (37) into (36), the proof is then complete. ∎

Note that while the order of $\|\boldsymbol{\lambda}_1\|$ in Proposition 2 is not unique to ensure $\sigma(\mathbf{H}) = \mathbf{o}(T)$, the bound derived in (52) ensures that it can neither be too big nor too small.

APPENDIX B
PROOF OF LEMMA 1

Recall that the iterate $\mathbf{x}_{t+1}$ is the optimal solution to the problem (11), thus the optimality condition implies that [38]

$$(\mathbf{x}-\mathbf{x}_{t+1})^\top\left(\eta\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)+\mathbf{H}_t^{\frac{1}{2}}(\mathbf{x}_{t+1}-\mathbf{x}_t)\right)\geq 0\quad\forall\mathbf{x}\in\mathcal{X}. \qquad (38)$$

With (38) in hand, we can thus upper bound the following:

$$\begin{aligned}
&\eta(\mathbf{x}_t-\mathbf{x})^\top\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\\
&\quad=\eta(\mathbf{x}_{t+1}-\mathbf{x})^\top\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)+\eta(\mathbf{x}_t-\mathbf{x}_{t+1})^\top\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\\
&\quad=(\mathbf{x}-\mathbf{x}_{t+1})^\top\left(\mathbf{H}_t^{\frac{1}{2}}(\mathbf{x}_t-\mathbf{x}_{t+1})-\eta\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\right)\\
&\qquad+(\mathbf{x}-\mathbf{x}_{t+1})^\top\mathbf{H}_t^{\frac{1}{2}}(\mathbf{x}_{t+1}-\mathbf{x}_t)+\eta(\mathbf{x}_t-\mathbf{x}_{t+1})^\top\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\\
&\quad\overset{(a)}{\leq}(\mathbf{x}-\mathbf{x}_{t+1})^\top\mathbf{H}_t^{\frac{1}{2}}(\mathbf{x}_{t+1}-\mathbf{x}_t)+\eta(\mathbf{x}_t-\mathbf{x}_{t+1})^\top\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)
\end{aligned}\tag{39}$$

where (a) follows from (38). We can further expand the first term in the RHS of (39) by

$$\begin{aligned}
(\mathbf{x}-\mathbf{x}_{t+1})^\top\mathbf{H}_t^{\frac{1}{2}}(\mathbf{x}_{t+1}-\mathbf{x}_t)=&\frac{1}{2}\|\mathbf{x}-\mathbf{x}_t\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\frac{1}{2}\|\mathbf{x}_{t+1}-\mathbf{x}_t\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\\
&-\frac{1}{2}\|\mathbf{x}-\mathbf{x}_{t+1}\|^2_{\mathbf{H}_t^{\frac{1}{2}}}.
\end{aligned}\tag{40}$$

For the second term in the RHS of (39), we have

$$\begin{aligned}
\eta(\mathbf{x}_t-\mathbf{x}_{t+1})^\top\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)&\overset{(b)}{\leq}\eta\|\mathbf{x}_t-\mathbf{x}_{t+1}\|_{\mathbf{H}_t^{\frac{1}{2}}}\|\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|_{\mathbf{H}_t^{-\frac{1}{2}}}\\
&\overset{(c)}{\leq}\frac{1}{2}\|\mathbf{x}_t-\mathbf{x}_{t+1}\|^2_{\mathbf{H}_t^{\frac{1}{2}}}+\frac{\eta^2}{2}\|\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2_{\mathbf{H}_t^{-\frac{1}{2}}}
\end{aligned}\tag{41}$$

where (b) uses the Cauchy–Schwartz inequality, and (c) is due to Young's inequality.

Plugging (40) and (41) into (39) leads to (17), which completes the first part of the proof.

Likewise, using the dual update (14), we have

$$\begin{aligned}
\|\boldsymbol{\lambda}-\boldsymbol{\lambda}_{t+1}\|^2&=\left\|\boldsymbol{\lambda}-[\boldsymbol{\lambda}_t+\mu\nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)]^+\right\|^2\\
&\overset{(d)}{\leq}\|\boldsymbol{\lambda}-\boldsymbol{\lambda}_t\|^2-2\mu(\boldsymbol{\lambda}-\boldsymbol{\lambda}_t)^\top\nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\\
&\quad+\mu^2\|\nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2
\end{aligned}\tag{42}$$

where (d) uses the nonexpansive property of the projection operator, and we can conclude (18) by rearranging terms.

Appendix C
Proof of Lemma 2

Using the result in [25, Lemma 4], it follows that:

$$\frac{\eta}{2}\sum_{t=1}^{T}\|\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2_{\mathbf{H}_t^{-\frac{1}{2}}}\leq\eta\sum_{i=1}^{I}\|\nabla^i_{\mathbf{x}}\mathcal{L}_{1:T}(\mathbf{x}_{1:T},\boldsymbol{\lambda}_{1:T})\|.\tag{43}$$

Therefore, the gradient with respect to primal variable can be bounded by

$$\begin{aligned}
\frac{\eta}{2}\sum_{t=1}^{T}\|\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2_{\mathbf{H}_t^{-\frac{1}{2}}}&\leq\eta\sum_{i=1}^{I}\|\nabla^i_{\mathbf{x}}\mathcal{L}_{1:T}(\mathbf{x}_{1:T},\boldsymbol{\lambda}_{1:T})\|\\
&\leq\eta\sum_{i=1}^{I}\sqrt{\textstyle\sum_{t=1}^{T}\left(\nabla^i_{\mathbf{x}}f_t(\mathbf{x}_t)+\sum_{j=1}^{I}\lambda_t^j\nabla^i_{\mathbf{x}}g_t^j(\mathbf{x}_t)\right)^2}\\
&\overset{(a)}{\leq}\eta\sum_{i=1}^{I}\sqrt{\textstyle\sum_{t=1}^{T}(I+1)\left(\left(\nabla^i_{\mathbf{x}}f_t(\mathbf{x}_t)\right)^2+\sum_{j=1}^{I}\left(\lambda_t^j\nabla^i_{\mathbf{x}}g_t^j(\mathbf{x}_t)\right)^2\right)}\\
&\leq\eta\sum_{i=1}^{I}\sqrt{\textstyle\sum_{t=1}^{T}(I+1)\left(G_i^2+\sum_{j=1}^{I}\left(\lambda_t^j\right)^2G_i^2\right)}\\
&\overset{(b)}{\leq}\eta\sum_{i=1}^{I}G_i\sqrt{(I+1)T}+\eta\sum_{i=1}^{I}G_i\sqrt{(I+1)\textstyle\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2}\\
&\leq\eta G\sqrt{(I+1)T}+\eta G\sqrt{(I+1)\textstyle\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2}
\end{aligned}\tag{44}$$

where (a) uses the inequality $(a_1+\ldots+a_n)^2\leq n(a_1^2+\ldots+a_n^2)$, (b) follows from $\|\boldsymbol{\lambda}_t\|^2=\sum_{j=1}^{I}(\lambda_t^j)^2$ and $\sqrt{a_1+a_2}\leq\sqrt{a_1}+\sqrt{a_2}$, and the constant is defined as $G\coloneqq\sum_{i=1}^{I}G_i$. And for the online gradient of Lagrangian with respect to dual variable is

$$\begin{aligned}
\frac{\mu}{2}\sum_{t=1}^{T}\|\nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2&\leq\frac{\mu}{2}\sum_{t=1}^{T}\sum_{i=1}^{I}\left(g_t^i(\mathbf{x}_t)-\theta\lambda_t^i\right)^2\\
&\overset{(c)}{\leq}\mu\sum_{t=1}^{T}\sum_{i=1}^{I}\left(\left(g_t^i(\mathbf{x}_t)\right)^2+\left(\theta\lambda_t^i\right)^2\right)\\
&\leq\mu R^2G^2T+\mu\theta^2\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2
\end{aligned}\tag{45}$$

where (c) again uses the inequality $(a_1+a_2)^2\leq 2(a_1^2+a_2^2)$.

Appendix D
Proof of Theorem 1

Given $\boldsymbol{\lambda}_t$, the convexity of $\mathcal{L}_t(\mathbf{x},\boldsymbol{\lambda}_t)$ implies

$$\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)-\mathcal{L}_t(\mathbf{x},\boldsymbol{\lambda}_t)\leq(\mathbf{x}_t-\mathbf{x})^\top\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\tag{46a}$$

and likewise, the concavity of $\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda})$ with respect to $\boldsymbol{\lambda}$ leads to

$$\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda})-\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\leq(\boldsymbol{\lambda}-\boldsymbol{\lambda}_t)^\top\nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t).\tag{46b}$$

Combining (46a) and (46b) leads to (15).

Plugging (17) and (18) in Lemma 1 into (15), and setting $\mathbf{x}=\mathbf{x}_t^*$ defined in (7), we arrive at

$$\begin{aligned}
\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda})-\mathcal{L}_t\left(\mathbf{x}_t^*,\boldsymbol{\lambda}_t\right)\leq&\frac{1}{2\eta}\left(\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\right)\\
&+\frac{\eta}{2}\|\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2_{\mathbf{H}_t^{-\frac{1}{2}}}\\
&+\frac{1}{2\mu}\left(\|\boldsymbol{\lambda}-\boldsymbol{\lambda}_t\|^2-\|\boldsymbol{\lambda}-\boldsymbol{\lambda}_{t+1}\|^2\right)\\
&+\frac{\mu}{2}\|\nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2.
\end{aligned}\tag{47}$$

Summing up (47) over $t=1,2,\ldots,T$, we find

$$\begin{aligned}
&\sum_{t=1}^{T}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda})-\mathcal{L}_t\left(\mathbf{x}_t^*,\boldsymbol{\lambda}_t\right)\\
&\quad\overset{(a)}{\leq}\frac{1}{2\eta}\sum_{t=1}^{T}\left(\left\|\mathbf{x}_t^*-\mathbf{x}_t\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}-\left\|\mathbf{x}_t^*-\mathbf{x}_{t+1}\right\|^2_{\mathbf{H}_t^{\frac{1}{2}}}\right)\\
&\qquad+\frac{1}{2\mu}\|\boldsymbol{\lambda}-\boldsymbol{\lambda}_1\|^2+\frac{\mu}{2}\sum_{t=1}^{T}\|\nabla_{\boldsymbol{\lambda}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|^2\\
&\qquad+\frac{\eta}{2}\sum_{t=1}^{T}\|\nabla_{\mathbf{x}}\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}_t)\|_{\mathbf{H}_t^{-\frac{1}{2}}}
\end{aligned}\tag{48}$$

where (a) uses the non-negativity of $\|\boldsymbol{\lambda}-\boldsymbol{\lambda}_{T+1}\|^2$.

Note that the three terms in the RHS of (48) have been bounded in Lemmas 1, 2, and 4, respectively. Hence, simply plugging (19), (20), and (28) into (48), we arrive at

$$\begin{aligned}\sum_{t=1}^{T}&\big(\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda})-\mathcal{L}_t\big(\mathbf{x}_t^*,\boldsymbol{\lambda}_t\big)\big)\\ &\le \frac{R}{\eta}\sigma_{\max}\Big(\mathbf{H}_T^{\frac{1}{2}}\Big)\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)+\mu R^2G^2T+\frac{\delta R^2}{2\eta}\\ &\quad+\left(\frac{R^2}{2\eta}+\eta\right)G\sqrt{(I+1)\textstyle\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2}+\mu\theta^2\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2\\ &\quad+\left(\frac{R^2}{2\eta}+\eta\right)G\sqrt{(I+1)T}+\frac{1}{2\mu}\|\boldsymbol{\lambda}\|^2+\frac{1}{2\mu}\|\boldsymbol{\lambda}_1\|^2\end{aligned}\tag{49}$$

where we used that $[1/(2\mu)]\|\boldsymbol{\lambda}-\boldsymbol{\lambda}_1\|^2 \le [1/(2\mu)]\|\boldsymbol{\lambda}\|^2 + [1/(2\mu)]\|\boldsymbol{\lambda}_1\|^2$.

On the other hand, also note that the definition of the online Lagrangian in (9) gives rise to

$$\begin{aligned}\sum_{t=1}^{T}&\big(\mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda})-\mathcal{L}_t\big(\mathbf{x}_t^*,\boldsymbol{\lambda}_t\big)\big)=\sum_{t=1}^{T}\big(f_t(\mathbf{x}_t)-f_t\big(\mathbf{x}_t^*\big)\big)\\ &\quad+\sum_{t=1}^{T}\boldsymbol{\lambda}^\top\mathbf{g}_t(\mathbf{x}_t)-\sum_{t=1}^{T}\boldsymbol{\lambda}_t^\top\mathbf{g}_t(\mathbf{x}_t^*)-\frac{\theta T}{2}\|\boldsymbol{\lambda}\|^2+\sum_{t=1}^{T}\frac{\theta}{2}\|\boldsymbol{\lambda}_t\|^2\\ &\overset{(b)}{\ge}\sum_{t=1}^{T}\big(f_t(\mathbf{x}_t)-f_t\big(\mathbf{x}_t^*\big)\big)+\sum_{t=1}^{T}\boldsymbol{\lambda}^\top\mathbf{g}_t(\mathbf{x}_t)-\frac{\theta T}{2}\|\boldsymbol{\lambda}\|^2\\ &\quad+\sum_{t=1}^{T}\frac{\theta}{2}\|\boldsymbol{\lambda}_t\|^2\end{aligned}\tag{50}$$

where (b) follows since the minimizer $\mathbf{x}_t^*$ defined in (7) is a feasible solution, i.e., $\mathbf{g}_t(\mathbf{x}_t^*) \le \mathbf{0}$ thus $\sum_{t=1}^{T}\boldsymbol{\lambda}_t^\top\mathbf{g}_t(\mathbf{x}_t^*) \le 0$. Combining (49) and (50), we have

$$\begin{aligned}\sum_{t=1}^{T}&\big(f_t(\mathbf{x}_t)-f_t\big(\mathbf{x}_t^*\big)\big)+\sum_{t=1}^{T}\boldsymbol{\lambda}^\top\mathbf{g}_t(\mathbf{x}_t)-\left(\frac{\theta T}{2}+\frac{1}{2\mu}\right)\|\boldsymbol{\lambda}\|^2\\ &\le\frac{R}{\eta}\sigma_{\max}(\mathbf{H}_T^{\frac{1}{2}})\mathbb{V}(\mathbf{x}_{1:T}^*)+\mu R^2G^2T+\frac{\delta R^2}{2\eta}\\ &\quad+\left(\mu\theta^2-\frac{\theta}{2}\right)\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2\\ &\quad+\left(\frac{R^2}{2\eta}+\eta\right)G\sqrt{(I+1)\textstyle\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2}\\ &\quad+\left(\frac{R^2}{2\eta}+\eta\right)G\sqrt{(I+1)T}+\frac{\|\boldsymbol{\lambda}_1\|^2}{2\mu}.\end{aligned}\tag{51}$$

By selecting $\eta = R/\sqrt{2}$, $\mu = [1/(RG)]T^{-(5/8)}$, and $\theta = RGT^{-(1/8)}$, and initializing $\boldsymbol{\lambda}_1$ such that $\|\boldsymbol{\lambda}_1\| = 4\sqrt{(I+1)}T^{(1/8)}$, one can easily verify that the conditions in Proposition 1 are satisfied with $\rho = \sqrt{\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2} \ge \|\boldsymbol{\lambda}_1\|$, which implies that

$$\left(\mu\theta^2-\frac{\theta}{2}\right)\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2+\left(\frac{R^2}{2\eta}+\eta\right)G\sqrt{(I+1)\textstyle\sum_{t=1}^{T}\|\boldsymbol{\lambda}_t\|^2}\le 0$$

together with (51) leading to

$$\begin{aligned}\sum_{t=1}^{T}&\big(f_t(\mathbf{x}_t)-f_t\big(\mathbf{x}_t^*\big)\big)+\sum_{t=1}^{T}\boldsymbol{\lambda}^\top\mathbf{g}_t(\mathbf{x}_t)-\Big(T^{\frac{7}{8}}+T^{\frac{5}{8}}\Big)\frac{RG}{2}\|\boldsymbol{\lambda}\|^2\\ &\le\sqrt{2}\sigma_{\max}\Big(\mathbf{H}_T^{\frac{1}{2}}\Big)\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)+\frac{\delta R}{\sqrt{2}}+\frac{RG}{2}\|\boldsymbol{\lambda}_1\|^2T^{\frac{5}{8}}\\ &\quad+RGT^{\frac{3}{8}}+\sqrt{2}RG\sqrt{(I+1)T}\\ &\overset{(c)}{\le}\sqrt{2}\sigma(\mathbf{H})\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)+\frac{RG}{2}\|\boldsymbol{\lambda}_1\|^2T^{\frac{5}{8}}+\epsilon_0\end{aligned}\tag{52}$$

where (c) uses Proposition 2, and the constant ϵ_0 is defined as $\epsilon_0 \coloneqq [(\delta R)/\sqrt{2}] + RGT^{(3/8)} + \sqrt{2}RG\sqrt{(I+1)T} = \mathcal{O}(\sqrt{T})$.

Notice that the RHS of (51) is irrelevant to the choice of $\boldsymbol{\lambda}$, thus we can maximize its LHS over $\boldsymbol{\lambda}$, given by

$$\sum_{t=1}^{T}\boldsymbol{\lambda}^\top\mathbf{g}_t(\mathbf{x}_t)-\Big(T^{\frac{7}{8}}+T^{\frac{5}{8}}\Big)\frac{RG}{2}\|\boldsymbol{\lambda}\|^2.\tag{53}$$

Using $\boldsymbol{\lambda} = [(\sum_{t=1}^{T}\mathbf{g}_t(\mathbf{x}_t))/((T^{7/8}+T^{5/8})RG)]$ in the RHS of (52), it follows that:

$$\begin{aligned}\sum_{t=1}^{T}&\big(f_t(\mathbf{x}_t)-f_t\big(\mathbf{x}_t^*\big)\big)+\sum_{t=1}^{T}\boldsymbol{\lambda}^\top\mathbf{g}_t(\mathbf{x}_t)-\Big(T^{\frac{7}{8}}+T^{\frac{5}{8}}\Big)\frac{RG}{2}\|\boldsymbol{\lambda}\|^2\\ &=\sum_{t=1}^{T}\big(f_t(\mathbf{x}_t)-f_t\big(\mathbf{x}_t^*\big)\big)+\frac{\|\sum_{t=1}^{T}\mathbf{g}_t(\mathbf{x}_t)\|^2}{2\big(T^{7/8}+T^{5/8}\big)RG}\\ &\overset{(d)}{\le}\sqrt{2}\sigma(\mathbf{H})\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)+\frac{RG}{2}\|\boldsymbol{\lambda}_1\|^2T^{\frac{5}{8}}+\epsilon_0\end{aligned}\tag{54}$$

where (d) follows from the LHS of (52).

To obtain the dynamic regret bound defined in (6), observe that $\|\sum_{t=1}^{T}\mathbf{g}_t(\mathbf{x}_t)\|^2 \ge 0$, then it follows from (54) that:

$$\mathrm{Reg}_T^{\mathrm{d}} \le \sqrt{2}\sigma(\mathbf{H})\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)+\frac{RG}{2}\|\boldsymbol{\lambda}_1\|^2T^{\frac{5}{8}}+\epsilon_0\tag{55}$$

where $\sigma(\mathbf{H}) = \mathcal{O}(T^{(7/8)})$, $\|\boldsymbol{\lambda}_1\|^2 = \mathcal{O}(T^{(1/4)})$, and $\epsilon_0 = \mathcal{O}(T^{*(1/2)})$. With short-hand notations $\epsilon_1 \coloneqq \sqrt{2}\sigma(\mathbf{H}) = \mathcal{O}(T^{(7/8)})$ and $\epsilon_2 \coloneqq [(RG)/2]\|\boldsymbol{\lambda}_1\|^2 = \mathcal{O}(T^{(1/4)})$, we can rewrite (55) as

$$\mathrm{Reg}_T^{\mathrm{d}} \le \epsilon_2T^{\frac{5}{8}}+\epsilon_1\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)+\epsilon_0 = \mathcal{O}\Big(T^{\frac{7}{8}}\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)\Big).\tag{56}$$

On the other hand, the mean-value theorem implies that there exists $\hat{\mathbf{x}}$ such that $f_t(\mathbf{x}_t)-f_t(\mathbf{x}_t^*) = (\mathbf{x}_t-\mathbf{x}_t^*)^\top\nabla f_t(\hat{\mathbf{x}}) \ge -\|\mathbf{x}_t-\mathbf{x}_t^*\|\|\nabla f_t(\hat{\mathbf{x}})\| \ge -RG$. Therefore, we have

$$\frac{\|\sum_{t=1}^{T}\mathbf{g}_t(\mathbf{x}_t)\|^2}{2\big(T^{7/8}+T^{5/8}\big)RG} \le \epsilon_2T^{\frac{5}{8}}+\epsilon_1\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)+\epsilon_0+RGT.\tag{57}$$

Rearranging terms in (57), we can conclude that

$$\begin{aligned}\left\|\sum_{t=1}^{T}\mathbf{g}_t(\mathbf{x}_t)\right\| &\le 2\sqrt{RG}T^{\frac{7}{16}}\sqrt{\epsilon_2T^{\frac{5}{8}}+\epsilon_1\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)+\epsilon_0+RGT}\\ &=\mathcal{O}\left(\max\left\{T^{\frac{15}{16}},T^{\frac{7}{8}}\sqrt{\mathbb{V}\big(\mathbf{x}_{1:T}^*\big)}\right\}\right)\end{aligned}\tag{58}$$

from which the proof is complete.

REFERENCES

[1] T. Chen, Y. Shen, Q. Ling, and G. B. Giannakis, "Online learning for 'thing-adaptive' fog computing in IoT," in *Proc. Asilomar Conf.*, Pacific Grove, CA, USA, Oct. 2017, pp. 664–668.

[2] L. Atzori, A. Iera, and G. Morabito, "The Internet of Things: A survey," *Comput. Netw.*, vol. 54, no. 15, pp. 2787–2805, Oct. 2010.

[3] J. A. Stankovic, "Research directions for the Internet of Things," *IEEE Internet Things J.*, vol. 1, no. 1, pp. 3–9, Feb. 2014.

[4] "The Internet of Things: Extend the cloud to where the things are," San Jose, CA, USA, Cisco, White Paper, 2016.

[5] X. Cheng, L. Fang, L. Yang, and S. Cui, "Mobile big data: The fuel for data-driven wireless," *IEEE Internet Things J.*, vol. 4, no. 5, pp. 1489–1516, Oct. 2017.

[6] X. Cheng, L. Fang, X. Hong, and L. Yang, "Exploiting mobile big data: Sources, features, and applications," *IEEE Netw.*, vol. 31, no. 1, pp. 72–79, Jan./Feb. 2017.

[7] T. Chen, S. Barbarossa, X. Wang, G. B. Giannakis, and Z.-L. Zhang, "Learning and management for Internet-of-Things: Accounting for adaptivity and scalability," *Proc. IEEE*, submitted for publication.

[8] D. Lymberopoulos, A. Bamis, and A. Savvides, "Extracting spatiotemporal human activity patterns in assisted living using a home sensor network," *Universal Access Inf. Soc.*, vol. 10, no. 2, pp. 125–138, 2011.

[9] S. Barbarossa, S. Sardellitti, and P. Di Lorenzo, "Communicating while computing: Distributed mobile cloud computing over 5G heterogeneous networks," *IEEE Signal Process. Mag.*, vol. 31, no. 6, pp. 45–55, Nov. 2014.

[10] M. Chiang and T. Zhang, "Fog and IoT: An overview of research opportunities," *IEEE Internet Things J.*, vol. 3, no. 6, pp. 854–864, Dec. 2016.

[11] M. Satyanarayanan *et al.*, "An open ecosystem for mobile-cloud convergence," *IEEE Commun. Mag.*, vol. 53, no. 3, pp. 63–70, Mar. 2015.

[12] P. Mach and Z. Becvar, "Mobile edge computing: A survey on architecture and computation offloading," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 3, pp. 1628–1656, 3rd Quart., 2017.

[13] S. Sardellitti, G. Scutari, and S. Barbarossa, "Joint optimization of radio and computational resources for multicell mobile-edge computing," *IEEE Trans. Signal Inf. Process. Over Netw.*, vol. 1, no. 2, pp. 89–103, Jun. 2015.

[14] F. Wang, J. Xu, X. Wang, and S. Cui, "Joint offloading and computing optimization in wireless powered mobile-edge computing systems," *IEEE Trans. Wireless Commun.*, vol. 17, no. 3, pp. 1784–1797, Mar. 2018.

[15] O. Muñoz, A. Pascual-Iserte, and J. Vidal, "Optimization of radio and computational resources for energy efficiency in latency-constrained application offloading," *IEEE Trans. Veh. Technol.*, vol. 64, no. 10, pp. 4738–4755, Oct. 2015.

[16] X. Cao, F. Wang, J. Xu, R. Zhang, and S. Cui, "Joint computation and communication cooperation for mobile edge computing," in *Proc. IEEE WiOpt*, Shanghai, China, May 2018.

[17] L. Yang, J. Cao, H. Cheng, and Y. Ji, "Multi-user computation partitioning for latency sensitive mobile cloud applications," *IEEE Trans. Comput.*, vol. 64, no. 8, pp. 2253–2266, Aug. 2015.

[18] F. Samie *et al.*, "Distributed QoS management for Internet of Things under resource constraints," in *Proc. Int. Conf. Hardw. Softw. Codesign Syst. Synth.*, Pittsburgh, PA, USA, Oct. 2016, pp. 1–10.

[19] Y. Mao, J. Zhang, and K. B. Letaief, "Dynamic computation offloading for mobile-edge computing with energy harvesting devices," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 12, pp. 3590–3605, Dec. 2016.

[20] J. Liu, A. Eryilmaz, N. B. Shroff, and E. S. Bentley, "Heavy-ball: A new approach to tame delay and convergence in wireless network optimization," in *Proc. IEEE INFOCOM*, San Francisco, CA, USA, Apr. 2016, pp. 1–9.

[21] G. Lee, W. Saad, and M. Bennis, "An online secretary framework for fog network formation with minimal latency," in *IEEE Int. Conf. Commun.*, Paris, France, May 2017.

[22] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 4, pp. 2322–2358, 4th Quart., 2017.

[23] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. Int. Conf. Mach. Learn.*, Washington, DC, USA, Aug. 2003, pp. 928–935.

[24] E. Hazan, A. Agarwal, and S. Kale, "Logarithmic regret algorithms for online convex optimization," *Mach. Learn.*, vol. 69, nos. 2–3, pp. 169–192, Dec. 2007.

[25] J. Duchi, E. Hazan, and Y. Singer, "Adaptive subgradient methods for online learning and stochastic optimization," *J. Mach. Learn. Res.*, vol. 12, pp. 2121–2159, Jul. 2011.

[26] O. Besbes, Y. Gur, and A. Zeevi, "Non-stationary stochastic optimization," *Oper. Res.*, vol. 63, no. 5, pp. 1227–1244, Sep. 2015.

[27] E. C. Hall and R. M. Willett, "Online convex optimization in dynamic environments," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 4, pp. 647–662, Jun. 2015.

[28] M. Mahdavi, R. Jin, and T. Yang, "Trading regret for efficiency: Online convex optimization with long term constraints," *J. Mach. Learn. Res.*, vol. 13, pp. 2503–2528, Sep. 2012.

[29] R. Jenatton, J. C. Huang, and C. Archambeau, "Adaptive algorithms for online convex optimization with long-term constraints," in *Proc. Int. Conf. Mach. Learn.*, New York, NY, USA, Jun. 2016, pp. 402–411.

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

[31] H. Yu, M. Neely, and X. Wei, "Online convex optimization with stochastic constraints," in *Proc. Adv. Neural Inf. Process. Syst.*, Long Beach, CA, USA, Dec. 2017, pp. 1427–1437.

[32] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," in *Synthesis Lectures on Communication Networks*, vol. 3. San Rafael, CA, USA: Morgan & Claypool, 2010, pp. 1–211.

[33] H. Huang, Q. Ling, W. Shi, and J. Wang, "Collaborative resource allocation over a hybrid cloud center and edge server network," *J. Comput. Math.*, vol. 35, no. 4, pp. 421–436, 2017.

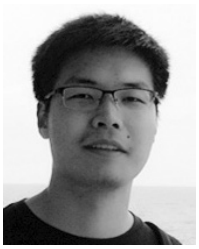
[34] Q. Zhu, R. Wang, Q. Chen, Y. Liu, and W. Qin, "IoT gateway: Bridging wireless sensor networks into Internet of Things," in *Proc. Int. Conf. Embedded Ubiquitous Comput.*, Hong Kong, Dec. 2010, pp. 347–352.

[35] (Dec. 2017). *8 Sensors to Help You Create a Smart Home*. [Online]. Available: www.ibm.com/blogs/internet-of-things/sensors-smart-home/

[36] S. Sicari, A. Rizzardi, L. A. Grieco, and A. Coen-Porisini, "Security, privacy and trust in Internet of Things: The road ahead," *Comput. Netw.*, vol. 76, pp. 146–164, Jan. 2015.

[37] T. Chen, Q. Ling, and G. B. Giannakis, "Learn-and-adapt stochastic dual gradients for network resource allocation," *IEEE Trans. Control Netw. Syst.*, to be published, doi: 10.1109/TCNS.2017.2774043.

[38] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York, NY, USA: Cambridge Univ. Press, 2004.



Tianyi Chen (S'14) received the B.Eng. degree (Highest Hons.) in communication science and engineering from Fudan University, Shanghai, China, in 2014, and the M.Sc. degree in electrical and computer engineering (ECE) from the University of Minnesota (UMN), Minneapolis, MN, USA, in 2016, where he is currently pursuing the Ph.D. degree.

His current research interests include online learning and stochastic optimization with applications to distributed machine learning and Internet-of-Things.

Mr. Chen was a Best Student Paper Award finalist of the Asilomar Conference on Signals, Systems, and Computers, the National Scholarship from China in 2013, the UMN ECE Department Fellowship in 2014, and the UMN Doctoral Dissertation Fellowship in 2017.

Qing Ling received the B.E. degree in automation and Ph.D. degree in control theory and control engineering from the University of Science and Technology of China, Hefei, China, in 2001 and 2006, respectively.

He was a Post-Doctoral Research Fellow with the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI, USA, from 2006 to 2009 and an Associate Professor with the Department of Automation, University of Science and Technology of China, from 2009 to 2017. He is currently a Professor with the School of Data and Computer Science, Sun Yat-sen University, Guangzhou, China. His current research interest includes decentralized network optimization and its applications.

Dr. Ling was a recipient of the 2017 IEEE Signal Processing Society Young Author Best Paper Award as a Supervisor and the 2017 International Consortium of Chinese Mathematicians Distinguished Paper Award. He is an Associate Editor of the IEEE TRANSACTIONS ON NETWORK AND SERVICE MANAGEMENT and IEEE SIGNAL PROCESSING LETTERS.

Yanning Shen (S'13) received the B.Sc. and M.Sc. degrees in electrical engineering from the University of Electronic Science and Technology of China, Hefei, China, in 2011 and 2014, respectively. She is currently pursuing the Ph.D. degree at the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA.

Her current research interest includes signal processing on graphs, network science, and machine learning.

Ms. Shen was a Best Student Paper Award finalist of the 2017 IEEE International Workshop on Computational Advances in Multisensor Adaptive Processing. She was selected to participate in the 2017 Rising Stars in EECS Workshop at Stanford University and was a recipient of the UMN Doctoral Dissertation Fellowship in 2018.

Georgios B. Giannakis (F'97) received the Diploma degree in electrical engineering from the National Technical University of Athens, Athens, Greece, in 1981, and the M.Sc. degree in electrical engineering, M.Sc. degree in mathematics, and Ph.D. degree in electrical engineering from the University of Southern California, Los Angeles, CA, USA, in 1983, 1986, and 1986, respectively.

He was with the University of Virginia, Charlottesville, VA, USA, from 1987 to 1998 and since 1999, he has been a Professor with the University of Minnesota, Minneapolis, MN, USA, where he holds an Endowed Chair in wireless telecommunications, a University of Minnesota McKnight Presidential Chair in Electrical and Computer Engineering, and serves as the Director of the Digital Technology Center. He has co-invented 30 patents. He has authored or co-authored over 400 journal papers, 700 conference papers, 25 book chapters, 2 edited books, and 2 research monographs (H-index 132). His current research interests include communications, networking, statistical learning, data science, Internet of Things, and network science with applications to social, brain, and power networks with renewables.

Dr. Giannakis was a co-recipient of nine Best Journal Paper Awards from the IEEE Signal Processing (SP) Society and the IEEE Communications Society, including the G. Marconi Prize Paper Award in Wireless Communications, the Technical Achievement Awards from the SP Society in 2000 and EURASIP in 2005, the Young Faculty Teaching Award, the G. W. Taylor Award for Distinguished Research from the University of Minnesota, and the IEEE Fourier Technical Field Award (inaugural recipient in 2015). He is a Fellow of EURASIP and has served the IEEE in a number of posts, including that of a Distinguished Lecturer for the IEEE SP Society.