An Online Convex Optimization Approach to Proactive Network Resource Allocation

Tianyi Chen, *Student Member, IEEE*, Qing Ling, *Senior Member, IEEE*, and Georgios B. Giannakis ⓘ , *Fellow, IEEE*

Abstract—Existing approaches to online convex optimization make sequential one-slot-ahead decisions, which lead to (possibly adversarial) losses that drive subsequent decision iterates. Their performance is evaluated by the so-called *regret* that measures the difference of losses between the online solution and the *best yet fixed* overall solution in *hindsight*. The present paper deals with online convex optimization involving adversarial loss functions and adversarial constraints, where the constraints are revealed after making decisions, and can be tolerable to instantaneous violations but must be satisfied in the long term. Performance of an online algorithm in this setting is assessed by the difference of its losses relative to the *best dynamic* solution with one-slot-ahead information of the loss function and the constraint (that is here termed *dynamic regret*); and the accumulated amount of constraint violations (that is here termed *dynamic fit*). In this context, a modified online saddle-point (MOSP) scheme is developed, and proved to simultaneously yield *sublinear* dynamic regret and fit, provided that the accumulated variations of per-slot minimizers and constraints are *sublinearly* growing with time. MOSP is also applied to the dynamic network resource allocation task, and it is compared with the well-known stochastic dual gradient method. Numerical experiments demonstrate the performance gain of MOSP relative to the state of the art.

Index Terms—Constrained optimization, primal-dual method, online convex optimization, network resource allocation.

I. INTRODUCTION

ONLINE convex optimization (OCO) is an emerging methodology for sequential inference with well documented merits especially when the sequence of convex costs varies in an unknown and possibly adversarial manner [2]–[4]. Starting from the seminal papers [2] and [3], most of the

early works evaluate OCO algorithms with a *static regret*, which measures the difference of costs (a.k.a. losses) between the online solution and the overall best static solution in hindsight. If an algorithm incurs static regret that increases sub-linearly with time, then its performance loss averaged over an infinite time horizon goes to zero; see also [4], [5], and references therein.

However, static regret is not a comprehensive performance metric [6]. Take online parameter estimation as an example. When the true parameter varies over time, a static benchmark (time-invariant estimator) itself often performs poorly so that achieving sub-linear static regret is no longer attractive. Recent works [6]–[9] extend the analysis of static regret to that of *dynamic regret*, where the performance of an OCO algorithm is benchmarked by the best dynamic solution with a-priori information on the one-slot-ahead cost function. Sub-linear dynamic regret is proved to be possible, if the dynamic environment changes slow enough for the accumulated variation of either costs or per-slot minimizers to be sub-linearly increasing with respect to the time horizon. When the per-slot costs depend on previous decisions, the so-termed competitive difference can be employed as an alternative of the static regret [10], [11].

The aforementioned works [6]–[11] deal with dynamic costs focusing on problems with time-invariant constraints that must be strictly satisfied, but do not allow for instantaneous violations of the constraints. The *long-term* effect of such instantaneous violations was studied in [12], where an online algorithm with sub-linear static regret and sub-linear accumulated constraint violation was also developed. The regret bounds in [12] have been improved in the discrete time domain [13] and the continuous time domain [14], respectively. Decentralized optimization with consensus constraints, as a special case of having long-term but time-invariant constraints, has been studied in [15]–[17]. Nevertheless, [12]–[17] do not deal with OCO under time-varying adversarial constraints.

In this context, the present paper considers OCO with time-varying constraints that must be satisfied in the long term. Under this setting, the learner first takes an action without knowing a-priori either the adversarial cost or the time-varying constraint, which are revealed by the nature subsequently. Its performance is evaluated by: i) *dynamic regret* that is the optimality loss relative to a sequence of instantaneous minimizers with known costs and constraints; and, ii) *dynamic fit* that accumulates constraint violations incurred by the online learner due to the lack of knowledge about future constraints. We compare the OCO setting here with those of existing works in Table I.

We further introduce a modified online saddle-point (MOSP) method in this novel OCO framework, where the learner deals

TABLE I
A SUMMARY OF RELATED WORKS ON DISCRETE TIME OCO

Reference	Type of benchmark	Long-term constraint	Adversarial constraint
[2]	Static and dynamic	No	No
[3]–[5]	Static	No	No
[6]–[11]	Dynamic	No	No
[12], [13], [15], [16]	Static	Yes	No
[17]	Dynamic	Yes	No
This work	Dynamic	Yes	Yes

with time-varying costs as well as time-varying but long-term constraints. We analytically establish that MOSP simultaneously achieves sub-linear dynamic regret and fit, provided that the accumulated variations of both minimizers and constraints grow sub-linearly with time. This result provides valuable insights for OCO with long-term constraints: *When the dynamic environment comprising both costs and constraints does not change on average, and the order of variations is known, the online decisions provided by MOSP are as good as the best dynamic solution over a long time horizon.*

To demonstrate the impact of these results, we further apply the proposed MOSP approach to a dynamic network resource allocation task, where online management of resources is sought without knowing future network states. Existing algorithms include first- and second-order methods in the dual domain [18]–[23], which are tailored for time-invariant deterministic formulations. To capture the temporal variations of network resources, stochastic formulation of network resource allocation has been extensively pursued since the seminal work of [24]; see also the celebrated stochastic dual gradient method in [25], [26]. These stochastic approximation-based approaches assume that the time-varying costs are i.i.d. or generally samples from a stationary ergodic stochastic process [27], [28]. However, performance of most stochastic schemes is established in an asymptotic sense, considering the ensemble of per slot averages or infinite samples across time. Clearly, stationarity may not hold in practice, especially when the stochastic process involves human participation. Inheriting merits of the OCO framework, the proposed MOSP approach operates in a fully online mode with only information at previous time slots, and further admits finite-sample performance analysis under a sequence of deterministic, or even adversarial costs and constraints within a budget of temporal variation.

Relative to existing works, the main contributions of the present paper are summarized as follows.

c1) We generalize the standard OCO framework with only adversarial costs in [2]–[5] to account for both adversarial costs and constraints. Different from the regret analysis in [12]–[16], performance here is established relative to the best dynamic benchmark, via metrics that we term dynamic regret and fit.

c2) We develop a MOSP algorithm to tackle this novel OCO problem, and analytically establish that MOSP yields simultaneously sub-linear dynamic regret and fit, provided that the accumulated variations of per-slot minimizers and constraints are known to grow sub-linearly with time.

c3) Our novel approach is tailored for online resource allocation tasks, where MOSP is compared with the popular stochastic dual gradient approach. Relative to the latter, MOSP remains operational in a broader practical

setting without probabilistic assumptions. Numerical tests demonstrate the gain of MOSP over existing alternatives.

Notation: $\mathbb{E}$ denotes expectation, $\mathbb{P}$ stands for probability, $(\cdot)^{\top}$ stands for vector and matrix transposition, and $\|\mathbf{x}\|$ denotes the ℓ_2-norm of a vector $\mathbf{x}$. Inequalities for vectors, e.g., $\mathbf{x} > \mathbf{0}$, are defined entry-wise. The positive projection operator is defined as $[\mathbf{a}]^+ := \max\{\mathbf{a}, \mathbf{0}\}$, also entry-wise. The indicator function $\mathbb{1}_{\{A\}}$ takes value 1 when the event A happens, and 0 otherwise.

II. OCO WITH LONG-TERM TIME-VARYING CONSTRAINTS

In this section, we introduce the generic OCO formulation with long-term time-varying constraints, along with pertinent metrics to evaluate an OCO algorithm.

A. Problem Formulation

We begin with the classical OCO setting, where constraints are time-invariant and must be strictly satisfied. OCO can be viewed as a repeated game between a learner and nature [2]–[4]. Consider that time is discrete and indexed by t. Per slot t, a learner selects an action $\mathbf{x}_t$ from a convex set $\mathcal{X} \subseteq \mathbb{R}^I$, and subsequently nature chooses a (possibly adversarial) loss function $f_t(\cdot) : \mathbb{R}^I \to \mathbb{R}$ through which the learner incurs a loss $f_t(\mathbf{x}_t)$. The convex set $\mathcal{X}$ is a-priori known and fixed over the entire time horizon. Although this standard OCO setting is appealing to various applications such as online regression and classification [2]–[4], it does not account for potential variations of (possibly unknown) constraints, and does not deal with constraints that can possibly be satisfied in the long term rather than a slot-by-slot basis. Online optimization with time-varying and long-term constraints is motivated for applications such as navigation, tracking, localization, and resource allocation [14], [25], [26], [29], [30]. Taking resource allocation as an example, time-varying long-term constraints are usually imposed to tolerate instantaneous violations when available resources cannot satisfy user requests, and hence allow flexible adaptation of online decisions to temporal variations of resource availability.

To broaden the applicability of OCO to these scenarios, we consider that per slot t, a learner selects an action $\mathbf{x}_t$ from a known and fixed convex set $\mathcal{X} \subseteq \mathbb{R}^I$, and then nature reveals not only a loss function $f_t(\cdot) : \mathbb{R}^I \to \mathbb{R}$ but also a time-varying (possibly adversarial) penalty function $\mathbf{g}_t(\cdot) : \mathbb{R}^I \to \mathbb{R}^I$. This function leads to a time-varying constraint $\mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}$, which is driven by the unknown dynamics in various applications, e.g., on-demand data request arrivals in resource allocation. Different from the known and fixed set $\mathcal{X}$, the time-varying constraint $\mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}$ can vary arbitrarily or even adversarially from slot to slot. It is revealed after the learner makes her/his

decision, and is hence hard to be satisfied in every time slot. This is indeed a major difference when comparing to settings where the time-varying constraints are revealed before making decisions. Therefore, the goal in this context is to find a sequence of online solutions $\{\mathbf{x}_t \in \mathcal{X}\}$ that minimize 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. Specifically, we aim to solve the following online optimization problem

$$\min_{\{\mathbf{x}_t \in \mathcal{X}, \forall t\}} \sum_{t=1}^{T} f_t(\mathbf{x}_t) \tag{1a}$$

$$\text{s.t. } \sum_{t=1}^{T} \mathbf{g}_t(\mathbf{x}_t) \leq \mathbf{0} \tag{1b}$$

where T is the time horizon, $\mathbf{x}_t \in \mathbb{R}^I$ is the decision variable, f_t is the cost function, $\mathbf{g}_t := [g_t^1, \ldots, g_t^I]^\top$ denotes the constraint function with ith entry $g_t^i : \mathbb{R}^I \to \mathbb{R}$, and $\mathcal{X} \in \mathbb{R}^I$ is a convex set. The formulation (1) extends the standard OCO framework to accommodate adversarial time-varying constraints that must be satisfied in the long term. Complemented by algorithm development and performance analysis to be carried in the following sections, the main contribution of the present paper is incorporation of long-term and time-varying constraints to markedly broaden the scope of OCO.

B. Performance and Feasibility Metrics

Regarding performance of online decisions $\{\mathbf{x}_t\}_{t=1}^{T}$, static regret is adopted as a metric by standard OCO schemes, under time-invariant and strictly satisfied constraints. The static regret measures the difference between the online loss of an OCO algorithm and that of the best fixed solution in hindsight [2]–[4]. Extending the definition of static regret over T slots to accommodate time-varying constraints, it can be written as

$$\text{Reg}_T^{\text{s}} := \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}^*) \tag{2}$$

where the best static solution $\mathbf{x}^*$ is obtained as

$$\mathbf{x}^* \in \arg\min_{\mathbf{x} \in \mathcal{X}} \sum_{t=1}^{T} f_t(\mathbf{x}) \quad \text{s. to } \mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}, \forall t. \tag{3}$$

A desirable OCO algorithm in this case is the one yielding a sub-linear regret [12], [13], meaning $\text{Reg}_T^{\text{s}} = \mathbf{o}(T)$. Consequently, $\lim_{T \to \infty} \text{Reg}_T^{\text{s}}/T = 0$ implies that the algorithm is "on average" no-regret, or in other words, not worse asymptotically than the best fixed solution $\mathbf{x}^*$. Though widely used in various OCO applications, the aforementioned *static regret* has several limitations. For instance, it fails to capture the convergence of online decisions $\{\mathbf{x}_t\}$ relative to the fixed best solution $\mathbf{x}^*$, since small regrets can be also achieved by having $f_t(\mathbf{x}_t)$ oscillate around $f_t(\mathbf{x}^*)$ [14]. Even when the sub-linear static regret does imply $\mathbf{x}_t$ approaching $\mathbf{x}^*$, targeting a rather coarse benchmark may be less useful especially in dynamic settings. For instance, [6, Example 2] shows that the gap between the best static and the best dynamic benchmark can be as large as $\mathcal{O}(T)$. Furthermore, since the time-varying constraint $\mathbf{g}_t(\mathbf{x}_t) \leq \mathbf{0}$ is not observed before making a decision $\mathbf{x}_t$, its feasibility can not be checked instantaneously.

In response to the quest for improved benchmarks in this dynamic setup, two metrics are considered here: *dynamic regret* and *dynamic fit*. The notion of dynamic regret (also termed tracking regret or adaptive regret) has been recently introduced in [6]–[9] to offer a competitive performance measure of OCO algorithms under time-invariant constraints. We adopt it in the setting of (1) by incorporating time-varying constraints

$$\text{Reg}_T^{\text{d}} := \sum_{t=1}^{T} f_t(\mathbf{x}_t) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^*) \tag{4}$$

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

$$\mathbf{x}_t^* \in \arg\min_{\mathbf{x} \in \mathcal{X}} f_t(\mathbf{x}) \quad \text{s.t. } \mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}. \tag{5}$$

Clearly, the dynamic regret is always larger than the static regret in (2), i.e., $\text{Reg}_T^{\text{s}} \leq \text{Reg}_T^{\text{d}}$, because $\sum_{t=1}^{T} f_t(\mathbf{x}^*)$ is always no smaller than $\sum_{t=1}^{T} f_t(\mathbf{x}_t^*)$ according to the definitions of $\mathbf{x}^*$ and $\mathbf{x}_t^*$. Hence, a sub-linear dynamic regret implies a sub-linear static regret, but not vice versa. The dynamic regret is suitable for cases where the goal is to track the time-varying solutions; e.g., AC power flow [31], and energy management policy [32].

To ensure feasibility of online decisions, the notion of *dynamic fit* is introduced to measure the accumulated violation of constraints; under time-invariant long-term constraints [12], [15] or under time-varying constraints [14]. It is defined as

$$\text{Fit}_T^{\text{d}} := \left\| \left[\sum_{t=1}^{T} \mathbf{g}_t(\mathbf{x}_t) \right]^+ \right\|. \tag{6}$$

Observe that the dynamic fit is zero if the accumulated violation $\sum_{t=1}^{T} \mathbf{g}_t(\mathbf{x}_t)$ is entry-wise less than zero. However, enforcing $\sum_{t=1}^{T} \mathbf{g}_t(\mathbf{x}_t) \leq \mathbf{0}$ is different from restricting $\mathbf{x}_t$ to meet $\mathbf{g}_t(\mathbf{x}_t) \leq \mathbf{0}$ in each and every slot. While the latter readily implies the former, the long-term (aggregate) constraint allows adaptation of online decisions to the environment dynamics; as a result, it is tolerable to have $\mathbf{g}_t(\mathbf{x}_t) \geq \mathbf{0}$ and $\mathbf{g}_{t+1}(\mathbf{x}_{t+1}) \leq \mathbf{0}$. Note that the definition of dynamic fit in (6) implicitly assumes that the instantaneous constraint violations can be compensated by the later strictly feasible decisions. When this is the case for resource allocation in power and cloud networks (see Section IV), extra modifications are required to account for other type of constraints, which go beyond the scope of the present paper.

An ideal algorithm in this broader OCO framework is the one that achieves both sub-linear dynamic regret and sub-linear dynamic fit. A sub-linear dynamic regret implies "no-regret" relative to the clairvoyant dynamic solution on the long-term average; i.e., $\lim_{T \to \infty} \text{Reg}_T^{\text{d}}/T = 0$; and a sub-linear dynamic fit indicates that the online strategy is also feasible on average; i.e., $\lim_{T \to \infty} \text{Fit}_T^{\text{d}}/T = 0$. Unfortunately, the sub-linear dynamic regret is not achievable in general, even under the special case of (1) where the time-varying constraint is absent [6]. For this reason, we aim at designing and analyzing an online strategy that generates a sequence $\{\mathbf{x}_t\}_{t=1}^{T}$ ensuring sub-linear dynamic regret and fit, under mild conditions that must be satisfied by the cost and constraint variations.

III. MODIFIED ONLINE SADDLE-POINT (MOSP) METHOD

In this section, a modified online saddle-point method is developed to solve (1), and its performance and feasibility are analyzed using the dynamic regret and fit metrics.

A. Algorithm Development

Consider now the per-slot problem (5), which contains the current objective $f_t(\mathbf{x})$, the current constraint $\mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}$, and a time-invariant constraint set $\mathcal{X}$. With $\boldsymbol{\lambda} \in \mathbb{R}_+^I$ denoting the Lagrange multiplier associated with the time-varying constraint, the online (partial) Lagrangian of (5) can be expressed as

$$\mathcal{L}_t(\mathbf{x}, \boldsymbol{\lambda}) := f_t(\mathbf{x}) + \boldsymbol{\lambda}^\top \mathbf{g}_t(\mathbf{x}) \tag{7}$$

where $\mathbf{x} \in \mathcal{X}$ remains implicit. For the online Lagrangian (7), we introduce a modified online saddle point (MOSP) approach, which takes a modified descent step in the primal domain, and a dual ascent step at each time slot t in a Gauss-Seidel manner. Specifically, given the previous primal iterate $\mathbf{x}_{t-1}$ and the current dual iterate $\boldsymbol{\lambda}_t$ at each slot t, the current decision $\mathbf{x}_t$ is the minimizer of the following optimization problem

$$\min_{\mathbf{x} \in \mathcal{X}} \nabla f_{t-1}(\mathbf{x}_{t-1})^\top (\mathbf{x} - \mathbf{x}_{t-1}) + \boldsymbol{\lambda}_t^\top \mathbf{g}_{t-1}(\mathbf{x}) + \frac{\|\mathbf{x} - \mathbf{x}_{t-1}\|^2}{2\alpha} \tag{8}$$

where α is a positive stepsize, and $\nabla f_{t-1}(\mathbf{x}_{t-1})$ is the gradient[1] of primal objective $f_{t-1}(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}_{t-1}$. After the current decision $\mathbf{x}_t$ is made, $f_t(\mathbf{x})$ and $\mathbf{g}_t(\mathbf{x})$ are observed, and the dual update takes the form

$$\boldsymbol{\lambda}_{t+1} = \left[\boldsymbol{\lambda}_t + \mu \nabla_{\boldsymbol{\lambda}} \mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t)\right]^+ = \left[\boldsymbol{\lambda}_t + \mu \mathbf{g}_t(\mathbf{x}_t)\right]^+ \tag{9}$$

where μ is also a positive stepsize, and $\nabla_{\boldsymbol{\lambda}} \mathcal{L}_t(\mathbf{x}_t, \boldsymbol{\lambda}_t) = \mathbf{g}_t(\mathbf{x}_t)$ is the gradient of online Lagrangian (7) with respect to (w.r.t.) $\boldsymbol{\lambda}$ at $\boldsymbol{\lambda} = \boldsymbol{\lambda}_t$. Clearly, updating $\boldsymbol{\lambda}_t$ and $\mathbf{x}_t$ at slot t only requires information of the cost and constraint at the previous slot.

Remark 1: The primal gradient step of the classical saddle-point approach in [12], [14], [15] is tantamount to minimizing a first-order approximation of $\mathcal{L}_{t-1}(\mathbf{x}, \boldsymbol{\lambda}_t)$ at $\mathbf{x} = \mathbf{x}_{t-1}$ plus a proximal term. We call the recursion (8) and (9) as a modified online saddle-point approach, since the primal update (8) is not an exact gradient step when the constraint $\mathbf{g}_t(\mathbf{x})$ is nonlinear w.r.t. $\mathbf{x}$. Similar to the primal update of OCO with long-term but time-invariant constraints in [13], the minimization in (8) penalizes the exact constraint violation $\mathbf{g}_t(\mathbf{x})$ instead of its first-order approximation, which improves control of constraint violations and facilitates performance analysis of MOSP. Nevertheless, when $\mathbf{g}_t(\mathbf{x})$ is linear, (8) and (9) reduce to the online saddle-point approach using the Gauss-Seidel update, which is different to those with the Jacobi one in [12], [14], [15].

Remark 2: When $\mathbf{g}_t(\mathbf{x})$ is linear or quadratic, the computational complexity of (8) is fairly low, and closed-form solutions are available. When $\mathbf{g}_t(\mathbf{x})$ is generally a convex function, penalizing the exact constraint in (8) comes with higher computational complexity than the saddle-point method. However, as (8) is strongly convex, iterative solvers can find the minimizer at linear convergence rate. Linearization techniques can be also

---

[1]One can replace the gradient by one of the sub-gradients when $f_t(\mathbf{x})$ is non-differentiable. The performance analysis still holds true for this case.

Algorithm 1: Modified Online Saddle-Point Method.

1: **Initialize:** primal iterate $\mathbf{x}_0$, dual iterate $\boldsymbol{\lambda}_1$, and proper stepsizes α and μ.
2: **for** $t = 1, 2 \ldots$ **do**
3:     Update primal variable $\mathbf{x}_t$ by solving (8).
4:     Observe the current cost $f_t(\mathbf{x})$ and constraint $\mathbf{g}_t(\mathbf{x})$.
5:     Update the dual variable $\boldsymbol{\lambda}_{t+1}$ via (9).
6: **end for**

---

incorporated to facilitate its implementation under fast dynamics, in which case the accuracy depends on the smoothness of $\mathbf{g}_t(\mathbf{x})$, and the variability of $\{\mathbf{x}_t\}$ (that can be e.g., controlled by the choice of stepsize α).

B. Performance Analysis

We proceed to show that for MOSP, the dynamic regret in (4) and the fit in (6) are both sub-linearly increasing, if the accumulated variations of per-slot minimizers and constraints are known to be sub-linearly growing. Before formally stating this result, we assume that the following conditions are satisfied.

Assumption 1: For every t, the cost function $f_t(\mathbf{x})$ and the time-varying constraint $\mathbf{g}_t(\mathbf{x})$ in (1) are convex.

Assumption 2: For every t, $f_t(\mathbf{x})$ has bounded gradient on $\mathcal{X}$; i.e., $\|\nabla f_t(\mathbf{x})\| \leq G$, $\forall \mathbf{x} \in \mathcal{X}$; and $\mathbf{g}_t(\mathbf{x})$ is bounded on $\mathcal{X}$; i.e., $\|\mathbf{g}_t(\mathbf{x})\| \leq M$, $\forall \mathbf{x} \in \mathcal{X}$.

Assumption 3: The radius of the convex feasible set $\mathcal{X}$ is bounded; i.e., $\|\mathbf{x} - \mathbf{y}\| \leq R$, $\forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$.

Assumption 4: There exists a constant $\epsilon > 0$, and an interior point $\tilde{\mathbf{x}}_t \in \mathcal{X}$ such that $\mathbf{g}_t(\tilde{\mathbf{x}}_t) \leq -\epsilon \mathbf{1}$, $\forall t$.

Assumption 5: The slack constant ϵ in Assumption 4 satisfies $\epsilon > \bar{V}(\mathbf{g})$, where the point-wise maximum variation of consecutive constraints is defined as

$$\bar{V}(\mathbf{g}) := \max_t \max_{\mathbf{x} \in \mathcal{X}} \left\| [\mathbf{g}_{t+1}(\mathbf{x}) - \mathbf{g}_t(\mathbf{x})]^+ \right\|. \tag{10}$$

Assumption 1 is necessary for regret analysis in the OCO setting. Assumption 2 bounds primal and dual gradients per slot, which is also typical in OCO [4], [7], [13], [15]. Assumption 3 restricts the action set to be bounded. Assumption 4 is Slater's condition, which guarantees the existence of a bounded optimal Lagrange multiplier [33]. Assumption 5 indicates that the slack constant ϵ is larger than the maximum variation of constraints. Although not always satisfied, it is a key assumption in our proof of the bounded dual iterate (the scaled fit). Equivalently, it requires $\min_{i,t} \max_{\mathbf{x} \in \mathcal{X}} [-g_t^i(\mathbf{x})]^+ > \max_t \max_{\mathbf{x} \in \mathcal{X}} \left\| [\mathbf{g}_{t+1}(\mathbf{x}) - \mathbf{g}_t(\mathbf{x})]^+ \right\|$, which is valid when the feasible region defined by $\mathbf{g}_t(\mathbf{x}) \leq \mathbf{0}$ is large enough, or, the trajectory of $\mathbf{g}_t(\mathbf{x})$ is smooth enough across time. Besides, Assumption 5 is analogous to the assumption of bounded multipliers in prior OCO works involving long-term constraints [15], [16]. One simple example for Assumption 5 to hold is that (cf. $I = 1$)

$$g_t(x) := 10x + \cos(\pi t), \text{ with } x \in \mathcal{X} := \{x| -2 \leq x \leq 2\} \tag{11}$$

where we have $\epsilon = \min_t \max_{x \in \mathcal{X}} [-g_t(x)]^+ = 19$, and $\bar{V}(\mathbf{g}) \leq 2$, so that $\epsilon > \bar{V}(\mathbf{g})$.

Under these assumptions, we are on track to first provide an upper bound for the dynamic fit.

Theorem 1: Under Assumptions 1–5 and the dual variable initialization $\boldsymbol{\lambda}_1 = \mathbf{0}$, the dual iterate for the MOSP recursion (8)–(9) is bounded by

$$\|\boldsymbol{\lambda}_t\| \leq \|\bar{\boldsymbol{\lambda}}\| := \mu M + \frac{2GR + R^2/(2\alpha) + (\mu M^2)/2}{\epsilon - \bar{V}(\mathbf{g})}, \ \forall t \tag{12}$$

and the dynamic fit in (6) is upper-bounded by

$$\begin{aligned} \text{Fit}_T^{\text{d}} &\leq \frac{\|\boldsymbol{\lambda}_{T+1}\|}{\mu} \leq \frac{\|\bar{\boldsymbol{\lambda}}\|}{\mu} \\ &= M + \frac{2GR/\mu + R^2/(2\alpha\mu) + M^2/2}{\epsilon - \bar{V}(\mathbf{g})} \end{aligned} \tag{13}$$

where G, M, R, and ϵ are as in Assumptions 2–4.

Proof: See Appendix A. ∎

Theorem 1 asserts that under the condition on the time-varying constraints, $\|\boldsymbol{\lambda}_t\|$ is uniformly upper-bounded, and more importantly, its scaled version $\|\boldsymbol{\lambda}_{T+1}\|/\mu$ upper bounds the dynamic fit. Observe that with a fixed primal stepsize α, Fit_T^{d} is in the order of $\mathcal{O}(1/\mu)$, thus a larger dual stepsize essentially enables a better satisfaction of long-term constraints. In addition, a smaller $\bar{V}(\mathbf{g})$ leads to a smaller dynamic fit, which also makes sense intuitively.

In the next theorem, we further bound the dynamic regret.

Theorem 2: Under Assumptions 1–5 and the dual variable initialization $\boldsymbol{\lambda}_1 = \mathbf{0}$, the MOSP recursion (8)–(9) yields a dynamic regret

$$\begin{aligned} \text{Reg}_T^{\text{d}} &\leq \frac{R V(\{\mathbf{x}_t^*\}_{t=1}^T)}{\alpha} + \frac{\alpha G^2 T}{2} + \frac{\mu M^2 (T+1)}{2} + \frac{R^2}{2\alpha} \\ &\quad + \|\bar{\boldsymbol{\lambda}}\| V(\{\mathbf{g}_t\}_{t=1}^T) \end{aligned} \tag{14}$$

where $V(\{\mathbf{x}_t^*\}_{t=1}^T)$ is the accumulated variation of the per-slot minimizers $\mathbf{x}_t^*$ defined as

$$V(\{\mathbf{x}_t^*\}_{t=1}^T) := \sum_{t=1}^T \underbrace{\|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\|}_{V(\mathbf{x}_t^*)} \tag{15}$$

and $V(\{\mathbf{g}_t\}_{t=1}^T)$ is the accumulated variation of constraints

$$V(\{\mathbf{g}_t\}_{t=1}^T) := \sum_{t=1}^T \underbrace{\max_{\mathbf{x} \in \mathcal{X}} \left\| [\mathbf{g}_{t+1}(\mathbf{x}) - \mathbf{g}_t(\mathbf{x})]^+ \right\|}_{V(\mathbf{g}_t)}. \tag{16}$$

Proof: See Appendix B. ∎

Theorem 2 asserts that MOSP's dynamic regret is upper-bounded by a constant depending on the accumulated variations of per-slot minimizers and time-varying constraints as well as the primal and dual stepsizes. While the dynamic regret in the current form (14) is hard to grasp, the next corollary shall demonstrate that Reg_T^{d} can be very small.

Based on Theorems 1–2, we can readily arrive at the following corollary regarding the optimal stepsizes.

Corollary 1: Under the same assumptions of Theorems 1–2, if the primal and dual stepsizes are chosen such that

$$\alpha = \mu = \max\left\{ \sqrt{\frac{V(\{\mathbf{x}_t^*\}_{t=1}^T)}{T}}, \sqrt{\frac{V(\{\mathbf{g}_t\}_{t=1}^T)}{T}} \right\} \tag{17}$$

then the dynamic regret is upper-bounded by

$$\text{Reg}_T^{\text{d}} = \mathcal{O}\left(\max\left\{ \sqrt{V(\{\mathbf{x}_t^*\}_{t=1}^T)T}, \sqrt{V(\{\mathbf{g}_t\}_{t=1}^T)T} \right\} \right) \tag{18}$$

and the dynamic fit is upper-bounded by

$$\text{Fit}_T^{\text{d}} = \mathcal{O}\left(\max\left\{ \frac{T}{V(\{\mathbf{x}_t^*\}_{t=1}^T)}, \frac{T}{V(\{\mathbf{g}_t\}_{t=1}^T)} \right\} \right). \tag{19}$$

Proof: The corollary follows by plugging (12) into (14), and optimizing (13) and (14) over the primal-dual stepsizes. ∎

According to Theorems 1–2 and Corollary 1, two sets of stepsizes are discussed next.

S1) Stepsizes without knowledge of variations: If the primal and dual stepsizes are chosen such that $\alpha = \mu = \mathcal{O}(T^{-\frac{1}{3}})$, then the dynamic fit is upper-bounded by

$$\text{Fit}_T^{\text{d}} = \mathcal{O}(T^{\frac{2}{3}}) \tag{20a}$$

and the dynamic regret is bounded by

$$\text{Reg}_T^{\text{d}} = \mathcal{O}\left(\max\left\{ V(\{\mathbf{x}_t^*\}_{t=1}^T)T^{\frac{1}{3}}, V(\{\mathbf{g}_t\}_{t=1}^T)T^{\frac{1}{3}}, T^{\frac{2}{3}} \right\} \right). \tag{20b}$$

S2) Stepsizes with knowledge of variations: Assume that there exists a constant $\beta \in [0, 1)$ such that the temporal variations satisfy $V(\{\mathbf{x}_t^*\}_{t=1}^T) = \mathbf{o}(T^\beta)$ and $V(\{\mathbf{g}_t\}_{t=1}^T) = \mathbf{o}(T^\beta)$. Corollary 1 then implies that choosing the stepsizes as $\alpha = \mu = \mathcal{O}(T^{\frac{\beta-1}{2}})$ leads to the dynamic fit

$$\text{Fit}_T^{\text{d}} = \mathcal{O}(T^{1-\beta}) = \mathbf{o}(T) \tag{21a}$$

and the corresponding dynamic regret

$$\text{Reg}_T^{\text{d}} = \mathcal{O}\left(T^{\frac{\beta+1}{2}} \right) = \mathbf{o}(T). \tag{21b}$$

In the case (S1), sub-linear regret and fit can be achieved given that $V(\{\mathbf{x}_t^*\}_{t=1}^T) = \mathbf{o}(T^{\frac{2}{3}})$ and $V(\{\mathbf{g}_t\}_{t=1}^T) = \mathbf{o}(T^{\frac{2}{3}})$. In the case (S2), the necessary conditions for the environment can be relaxed to $V(\{\mathbf{x}_t^*\}_{t=1}^T) = \mathbf{o}(T)$ and $V(\{\mathbf{g}_t\}_{t=1}^T) = \mathbf{o}(T)$, provided that a-priori knowledge of the environment is available. For example, when allocating resources to smart grids, the temporal variations of the best dynamic solutions and instantaneous constraints can be estimated using day-ahead forecasting of electricity loads and prices. Corollary 1 provides valuable insights for choosing optimal stepsizes in non-stationary settings. Specifically, adjusting stepsizes to match the variability of the environment is the key to achieving the optimal dynamic regret and fit. Intuitively, when the variation is fast (a larger β), slowly decaying stepsizes (thus larger stepsizes) can better track the potential changes; and vice versa.

It is instructive to give several cases where sub-linear accumulated variations emerge, so that the bounds in (21) hold.

C1) Intermittent switches: With $\mathbf{x}_t^* \neq \mathbf{x}_{t+1}^*$ or $\mathbf{g}_t \neq \mathbf{g}_{t+1}$ defining a switch, the number of switches is sub-linear over T; i.e., $\sum_{t=1}^T \mathbb{1}_{\{\mathbf{x}_t^* \neq \mathbf{x}_{t+1}^*\}} = T^\beta$, and $\sum_{t=1}^T \mathbb{1}_{\{\mathbf{g}_t \neq \mathbf{g}_{t+1}\}} = T^\beta$, $\forall \beta \in [0, 1)$. It then follows that $V(\{\mathbf{x}_t^*\}_{t=1}^T) = \mathcal{O}(T^\beta)$, and

$V(\{\mathbf{g}_t\}_{t=1}^T) = \mathcal{O}(T^\beta)$, since the one-slot variation of the minimizer and the constraint is bounded; see Assumptions 2–3.

C2) Decreasing variations: When the one-slot variations are decreasing over time such that $V(\mathbf{x}_t^*) = \mathcal{O}(t^{\beta-1})$ and $V(\mathbf{g}_t) = \mathcal{O}(t^{\beta-1})$, $\forall \beta \in [0,1)$, the accumulated variations of the per-slot minimizers and the consecutive constraints become $V(\{\mathbf{x}_t^*\}_{t=1}^T) = \mathcal{O}(T^\beta)$, and $V(\{\mathbf{g}_t\}_{t=1}^T) = \mathcal{O}(T^\beta)$.

Other cases do exist for which the accumulated variation is sub-linear, including the interplay between (C1) and (C2).

Remark 3: Theorems 1–2 and Corollary 1 are in the spirit of the recent works in [2], [6]–[9] and [17], where the regret bounds are established with respect to a dynamic benchmark in OCO without long-term time-varying constraints. Specifically, [6], [9] consider dynamic regret bounds for strongly-convex loss functions. For the general convex loss functions considered here, [2] reports the dynamic regret bound in the form of

$$\text{Reg}_T^{\text{d}} = \mathcal{O}\left(\sqrt{V(\{\mathbf{x}_t^*\}_{t=1}^T)T}\right) \quad (22)$$

and [6] states the bound in the form of

$$\text{Reg}_T^{\text{d}} = \mathcal{O}\left(V(\{f_t\}_{t=1}^T)^{\frac{1}{3}}T^{\frac{2}{3}}\right) \quad (23)$$

where the accumulated variation of loss functions is defined as $V(\{f_t\}_{t=1}^T) := \sum_{t=1}^T \max_{\mathbf{x}\in\mathcal{X}}\|f_{t+1}(\mathbf{x}) - f_t(\mathbf{x})\|$. The dynamic regret bound in [8] considers a hybrid version of (22) and (23), when the effect of dynamic models is further accounted for in the dynamic regret bounds of [7], [17]. When the functional variation $V(\{f_t\}_{t=1}^T)$ is not directly comparable to the variation of minimizers $V(\{\mathbf{x}_t^*\}_{t=1}^T)$, our regret bound in (18) immediately reduces to (22) in [2], by setting $\alpha = \sqrt{V(\{\mathbf{x}_t^*\}_{t=1}^T)/T}$. Note that [2], [6]–[9], [17] do not account for long-term and time-varying constraints, while the regret analysis is generalized here to the setting with long-term constraints. Interestingly though, in the considered setting, sub-linear dynamic regret and fit can be achieved when the environment consisting of the per-slot minimizer and the time-varying constraint *does not vary on average*, that is, $V(\{\mathbf{x}_t^*\}_{t=1}^T)$ and $V(\{\mathbf{g}_t\}_{t=1}^T)$ are sub-linearly increasing over T. Selecting the optimal stepsizes requires the knowledge of variations, and thus it is also promising to develop a parameter-free MOSP using the doubling trick [8], [34].

C. Beyond Dynamic Regret

Although the dynamic benchmark in (4) is more competitive than the static one in (2), it is worth noting that the sequence of the per-slot minimizer $\mathbf{x}_t^*$ in (5) is not the optimal solution to problem (1). Consider the offline optimal solutions to (1), i.e.,

$$\{\mathbf{x}_t^{\text{off}}\}_{t=1}^T \in \arg\min_{\{\mathbf{x}_t\in\mathcal{X},\forall t\}} \sum_{t=1}^T f_t(\mathbf{x}_t) \quad \text{s.t.} \quad \sum_{t=1}^T \mathbf{g}_t(\mathbf{x}_t) \leq \mathbf{0}. \quad (24)$$

Computing the per-slot minimizer $\mathbf{x}_t^*$ in (5) only requires one-slot-ahead information (namely, $f_t(\mathbf{x})$ and $g_t(\mathbf{x})$), while computing each $\mathbf{x}_t^{\text{off}}$ within $\{\mathbf{x}_t^{\text{off}}\}_{t=1}^T$ requires information over the entire time horizon (that is, $\{f_t(\mathbf{x})\}_{t=1}^T$ and $\{g_t(\mathbf{x})\}_{t=1}^T$). For this reason, we use the superscript "off" in $\{\mathbf{x}_t^{\text{off}}\}_{t=1}^T$ to emphasize that this solution comes from offline computation with information over T slots. Note that for the cases without

long-term constraints [6]–[9], the offline solutions $\{\mathbf{x}_t^{\text{off}}\}_{t=1}^T$ coincides with the sequence of per-slot minimizers $\{\mathbf{x}_t^*\}_{t=1}^T$.

Regarding feasibility, $\{\mathbf{x}_t^{\text{off}}\}_{t=1}^T$ exactly satisfies the long-term constraint (1b), while the solution of MOSP satisfies (1b) on average under mild conditions (cf. Corollary 1). For optimality, the cost of the online decisions $\{\mathbf{x}_t\}_{t=1}^T$ attained by MOSP is further benchmarked by the offline solutions $\{\mathbf{x}_t^{\text{off}}\}_{t=1}^T$. To this end, define MOSP's *optimality gap* as

$$\text{OptGap}_T^{\text{off}} := \sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}_t^{\text{off}}). \quad (25\text{a})$$

Intuitively, if $\{\mathbf{x}_t^{\text{off}}\}_{t=1}^T$ are close to $\{\mathbf{x}_t^*\}_{t=1}^T$, the dynamic regret Reg_T^{d} is able to provide an accurate performance measure in the sense of $\text{OptGap}_T^{\text{off}}$. Specifically, one can decompose the optimality gap as

$$\text{OptGap}_T^{\text{off}} = \underbrace{\sum_{t=1}^T f_t(\mathbf{x}_t) - \sum_{t=1}^T f_t(\mathbf{x}_t^*)}_{\mathcal{U}_1}$$
$$+ \underbrace{\sum_{t=1}^T f_t(\mathbf{x}_t^*) - \sum_{t=1}^T f_t(\mathbf{x}_t^{\text{off}})}_{\mathcal{U}_2} \quad (25\text{b})$$

where $\mathcal{U}_1$ corresponds to the dynamic regret Reg_T^{d} in (4) capturing the regret relative to the sequence of per-slot minimizers with one-slot-ahead information, and $\mathcal{U}_2$ is the difference between the performance of per-slot minimizers and the offline optimal solutions. Although the second term appears difficult to quantify, we will show next that $\mathcal{U}_2$ is driven by the accumulated variation of the dual functions associated with (5).

To this end, consider the dual function of the instantaneous primal problem (5), which can be expressed by minimizing the online Lagrangian in (7) at time t, namely [33]

$$\mathcal{D}_t(\boldsymbol{\lambda}) := \min_{\mathbf{x}\in\mathcal{X}} \mathcal{L}_t(\mathbf{x},\boldsymbol{\lambda}) = \min_{\mathbf{x}\in\mathcal{X}} f_t(\mathbf{x}) + \boldsymbol{\lambda}^\top \mathbf{g}_t(\mathbf{x}). \quad (26)$$

Likewise, the dual function of (1) over the entire horizon is

$$\mathcal{D}(\boldsymbol{\lambda}) := \min_{\{\mathbf{x}_t\in\mathcal{X},\forall t\}} \sum_{t=1}^T \mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda})$$
$$\overset{(a)}{=} \sum_{t=1}^T \min_{\mathbf{x}_t\in\mathcal{X}} \mathcal{L}_t(\mathbf{x}_t,\boldsymbol{\lambda}) \overset{(b)}{=} \sum_{t=1}^T \mathcal{D}_t(\boldsymbol{\lambda}) \quad (27)$$

where equality (a) holds since the minimization is separable across the summand at time t, and equality (b) is due to the definition of the per-slot dual function in (26). As the primal problems (1) and (5) are both convex, Slater's condition in Assumption 4 implies that strong duality holds. Accordingly, $\mathcal{U}_2$ in (25b) can be written as

$$\sum_{t=1}^T f_t(\mathbf{x}_t^*) - \sum_{t=1}^T f_t(\mathbf{x}_t^{\text{off}}) = \sum_{t=1}^T \max_{\boldsymbol{\lambda}_t\geq\mathbf{0}} \mathcal{D}_t(\boldsymbol{\lambda}_t)$$
$$- \max_{\boldsymbol{\lambda}\geq\mathbf{0}} \sum_{t=1}^T \mathcal{D}_t(\boldsymbol{\lambda}) \quad (28)$$

which is the difference between the dual objective of the static best solution, i.e., $\boldsymbol{\lambda}^* \in \arg\max_{\boldsymbol{\lambda} \geq \mathbf{0}} \sum_{t=1}^{T} \mathcal{D}_t(\boldsymbol{\lambda})$, and that of the per-slot best solution for (26), i.e., $\boldsymbol{\lambda}_t^* \in \arg\max_{\boldsymbol{\lambda}_t \geq \mathbf{0}} \mathcal{D}_t(\boldsymbol{\lambda}_t)$. Leveraging this special property of the dual problem, we next establish that $\mathcal{U}_2$ can be bounded by the variation of the dual function, thus providing an estimate of the optimality gap (25).

Proposition 1: Define the variation of the dual function (26) from time t to $t+1$ as

$$V(\mathcal{D}_t) := \max_{\boldsymbol{\lambda} \geq \mathbf{0}} \|\mathcal{D}_{t+1}(\boldsymbol{\lambda}) - \mathcal{D}_t(\boldsymbol{\lambda})\| \qquad (29)$$

and the total variation over the time horizon T as $V(\{\mathcal{D}_t\}_{t=1}^T) := \sum_{t=1}^{T} V(\mathcal{D}_t)$. Then the cost difference between the best offline solution and the best dynamic solution satisfies

$$\sum_{t=1}^{T} f_t(\mathbf{x}_t^*) - \sum_{t=1}^{T} f_t(\mathbf{x}_t^{\text{off}}) \leq 2TV(\{\mathcal{D}_t\}_{t=1}^T) \qquad (30)$$

where $\mathbf{x}_t^*$ is the minimizer of the instantaneous problem (5), and $\mathbf{x}_t^{\text{off}}$ solves (1) with all future information available. Combined with (25b), it readily follows that

$$\text{OptGap}_T^{\text{off}} \leq \text{Reg}_T^{\text{d}} + 2TV(\{\mathcal{D}_t\}_{t=1}^T) \qquad (31)$$

where Reg_T^{d} is defined in (4), and $\text{OptGap}_T^{\text{off}}$ in (25).

Proof: Instead of going to the primal domain, we upper bound $\mathcal{U}_2$ via the dual representation in (28). Letting $\tilde{t}$ denote any slot in $\mathcal{T} := \{1, \dots, T\}$, we have

$$\sum_{t \in \mathcal{T}} \max_{\boldsymbol{\lambda} \geq \mathbf{0}} \mathcal{D}_t(\boldsymbol{\lambda}) - \max_{\boldsymbol{\lambda} \geq \mathbf{0}} \sum_{t \in \mathcal{T}} \mathcal{D}_t(\boldsymbol{\lambda}) \qquad (32)$$
$$\leq \sum_{t \in \mathcal{T}} \left(\mathcal{D}_t(\boldsymbol{\lambda}_t^*) - \mathcal{D}_t(\boldsymbol{\lambda}_{\tilde{t}}^*) \right) \leq T \max_{t \in \mathcal{T}} \left\{ \mathcal{D}_t(\boldsymbol{\lambda}_t^*) - \mathcal{D}_t(\boldsymbol{\lambda}_{\tilde{t}}^*) \right\}.$$

The first inequality comes from the definition $\boldsymbol{\lambda}_t^* \in \arg\max_{\boldsymbol{\lambda} \geq \mathbf{0}} \mathcal{D}_t(\boldsymbol{\lambda})$. Note that if $\max_{t \in \mathcal{T}} \{\mathcal{D}_t(\boldsymbol{\lambda}_t^*) - \mathcal{D}_t(\boldsymbol{\lambda}_{\tilde{t}}^*)\} \leq 2V(\{\mathcal{D}_t\}_{t=1}^T)$, the proposition readily follows from (32). We will prove this inequality by contradiction. Assume there exists a slot $t_0 \in \mathcal{T}$ such that $\mathcal{D}_{t_0}(\boldsymbol{\lambda}_{t_0}^*) - \mathcal{D}_{t_0}(\boldsymbol{\lambda}_{\tilde{t}}^*) > 2V(\{\mathcal{D}_t\}_{t=1}^T)$, which implies that

$$\mathcal{D}_{\tilde{t}}(\boldsymbol{\lambda}_{\tilde{t}}^*) \overset{(a)}{\leq} \mathcal{D}_{t_0}(\boldsymbol{\lambda}_{\tilde{t}}^*) + V(\{\mathcal{D}_t\}_{t=1}^T) \overset{(b)}{<} \mathcal{D}_{t_0}(\boldsymbol{\lambda}_{t_0}^*) - V(\{\mathcal{D}_t\}_{t=1}^T)$$
$$\overset{(c)}{\leq} \mathcal{D}_{\tilde{t}}(\boldsymbol{\lambda}_{t_0}^*), \ \forall \tilde{t} \in \mathcal{T} \qquad (33)$$

where inequalities (a) and (c) come from the fact that $V(\{\mathcal{D}_t\}_{t=1}^T)$ is the accumulated variation over T slots, and hence $\max_{t_1, t_2 \in \mathcal{T}} \|\mathcal{D}_{t_1}(\boldsymbol{\lambda}) - \mathcal{D}_{t_2}(\boldsymbol{\lambda})\| \leq V(\{\mathcal{D}_t\}_{t=1}^T)$, while (b) is due to the hypothesis above. Note that $\mathcal{D}_{\tilde{t}}(\boldsymbol{\lambda}_{\tilde{t}}^*) < \mathcal{D}_{\tilde{t}}(\boldsymbol{\lambda}_{t_0}^*)$ in (33) contradicts the fact that $\boldsymbol{\lambda}_{\tilde{t}}^*$ is the maximizer of $\mathcal{D}_{\tilde{t}}(\boldsymbol{\lambda})$. Therefore, we have $\mathcal{D}_t(\boldsymbol{\lambda}_t^*) - \mathcal{D}_t(\boldsymbol{\lambda}_{\tilde{t}}^*) \leq 2V(\{\mathcal{D}_t\}_{t=1}^T)$, which completes the proof. ∎

The following remark provides an approach to improving the bound in Proposition 1.

Remark 4: Although the optimality gap in (31) appears to be at least linear w.r.t. T, one can use the "restarting" trick for dual variables, similar to that for primal variables in the unconstrained case; see e.g., [6]. Specifically, if the total variation $V(\{\mathcal{D}_t\}_{t=1}^T)$ is known a-priori, one can divide the entire

time horizon $\mathcal{T} := \{1, \dots, T\}$ into $\lceil T/\Delta_T \rceil$ sub-horizons (each with $\Delta_T = \mathbf{o}(T/V(\{\mathcal{D}_t\}_{t=1}^T))$ slots), and restart the dual iterate $\boldsymbol{\lambda}$ at the beginning of each sub-horizon. By assuming that $V(\{\mathcal{D}_t\}_{t=1}^T)$ is sub-linear w.r.t. T, one can guarantee that $\Delta_T \geq 1$ always exists. In this case, the optimality gap in (31) can be improved by

$$\text{OptGap}_T^{\text{off}} \leq \lceil T/\Delta_T \rceil \text{Reg}_{\Delta_T}^{\text{d}} + 2\Delta_T V(\{\mathcal{D}_t\}_{t=1}^T) \qquad (34a)$$

and the dynamic fit is the summation over each sub-horizon

$$\text{Fit}_T^{\text{d}} \leq \lceil T/\Delta_T \rceil \text{Fit}_{\Delta_T}^{\text{d}}. \qquad (34b)$$

To this end, if the regularity conditions of the environment in (21) are satisfied, one can properly set the primal-dual stepsizes to guarantee the sub-linear regret and fit on each sub-horizon. Correspondingly, the optimality gap and the dynamic fit in (34) are also both sub-linearly growing with time. Interested readers are referred to [6] for details of this restarting trick, which are omitted here due to space limitation.

IV. APPLICATION TO NETWORK RESOURCE ALLOCATION

In this section, we solve the network resource allocation problem within the OCO framework, and present numerical experiments to demonstrate the merits of our MOSP solver.

A. Online Network Resource Allocation

Consider the resource allocation problem over a cloud network [29], which is represented by a directed graph $\mathcal{G} = (\mathcal{I}, \mathcal{E})$ with node set $\mathcal{I}$ and edge set $\mathcal{E}$, where $|\mathcal{I}| = I$ and $|\mathcal{E}| = E$. Nodes considered here include mapping nodes collected in the set $\mathcal{J} = \{1, \dots, J\}$, and data centers collected in the set $\mathcal{K} = \{1, \dots, K\}$; i.e., we have $\mathcal{I} = \mathcal{J} \bigcup \mathcal{K}$.

Per time t, each mapping node j receives an exogenous data request b_t^j, and forwards the amount x_t^{jk} to each data center k in accordance with bandwidth availability. Each data center k schedules workload y_t^k according to its resource availability. Regarding y_t^k as the weight of a virtual outgoing edge $(k, *)$ from data center k, edge set $\mathcal{E} := \{(j, k), \forall j \in \mathcal{J}, k \in \mathcal{K}\} \bigcup \{(k, *), \forall k \in \mathcal{K}\}$ contains all the links connecting mapping nodes with data centers, and all the "virtual" edges coming out of the data centers. The $I \times E$ node-incidence matrix is formed with the (i, e)-th entry

$$\mathbf{A}_{(i,e)} = \begin{cases} 1, & \text{if link } e \text{ enters node } i \\ -1, & \text{if link } e \text{ leaves node } i \\ 0, & \text{else.} \end{cases} \qquad (35)$$

For compactness, collect the data workloads across edges $e = (i, j) \in \mathcal{E}$ in a resource allocation vector $\mathbf{x}_t := [x_t^{11}, \dots, x_t^{JK}, y_t^1, \dots, y_t^K]^\top \in \mathbb{R}_+^E$, and the exogenous load arrival rates of all nodes in a vector $\mathbf{b}_t := [b_t^1, \dots, b_t^J, 0 \dots, 0]^\top \in \mathbb{R}_+^I$. Then, the aggregate (endogenous plus exogenous) workloads of all nodes are given by $\mathbf{Ax}_t + \mathbf{b}_t$. When the i-th entry of $\mathbf{Ax}_t + \mathbf{b}_t$ is positive, there is service residual at node i; otherwise, node i over-serves the current workload arrival. Assume that each data center and mapping node has a local data queue to buffer unserved workloads [25]. With $\mathbf{q}_t := [q_t^1, \dots, q_t^{J+K}]^\top$ collecting the queue lengths at each mapping node and data center, the queue update is $\mathbf{q}_{t+1} = [\mathbf{q}_t + \mathbf{Ax}_t + \mathbf{b}_t]^+$, where $[\cdot]^+$ ensures that the queue length is
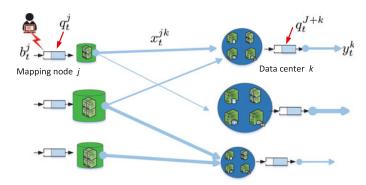
Fig. 1. A diagram of online network resource allocation. Per time t, mapping node j has an exogenous workload b_t^j plus that stored in the queue q_t^j, and schedules workload x_t^{jk} to data center k. Data center k serves an amount of workload y_t^k out of the assigned $\sum_{j=1}^J x_t^{jk}$ as well as that stored in its queue q_t^{J+k}. The thickness of each edge is proportional to its capacity.

always non-negative. The bandwidth limit of link (j, k) is $\bar{x}^{jk}$, and the resource capability of data center k is $\bar{y}^k$, which can be compactly expressed by $\mathbf{x} \in \mathcal{X}$ with $\mathcal{X} := \{0 \leq \mathbf{x} \leq \bar{\mathbf{x}}\}$ and $\bar{\mathbf{x}} := [\bar{x}^{11}, \ldots, \bar{x}^{JK}, \bar{y}^1, \ldots, \bar{y}^K]^\top$. The overall system diagram is depicted in Fig. 1.

For each data center, the power cost $f_t^k(y_t^k) := f^k(y_t^k; \theta_t^k)$ depends on a time-varying parameter θ_t^k, which captures the energy price and the renewable generation at data center k during slot t. The bandwidth cost $f_t^{jk}(x_t^{jk}) := f^{jk}(x_t^{jk}; \theta_t^{jk})$ characterizes the transmission delay and is parameterized by a time-varying scalar θ_t^{jk}. Scalars θ_t^k and θ_t^{jk} can be readily extended to vector forms. To keep the exposition simple, we use scalars to represent time-varying factors at nodes and edges.

Per slot t, the instantaneous cost $f_t(\mathbf{x}_t)$ aggregates the costs of power consumed at all data centers plus the bandwidth costs at all links, namely

$$f_t(\mathbf{x}_t) := \sum_{k \in \mathcal{K}} \underbrace{f_t^k(y_t^k)}_{\text{power cost}} + \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} \underbrace{f_t^{jk}(x_t^{jk})}_{\text{bandwidth cost}} \quad (36)$$

where the objective can be also written as $f_t(\mathbf{x}_t) := f(\mathbf{x}_t; \boldsymbol{\theta}_t)$ with $\boldsymbol{\theta}_t := [\theta_t^1, \ldots, \theta_t^K, \theta_t^{11}, \ldots, \theta_t^{JK}]^\top$ concatenating all time-varying parameters. Aiming to minimize the accumulated cost while serving all workloads, the optimal workload routing and allocation strategy in this cloud network is the solution of the following optimization problem

$$\min_{\{\mathbf{x}_t \in \mathcal{X}, \forall t\}} \sum_{t=1}^T f_t(\mathbf{x}_t) \quad \text{s.t. } \mathbf{q}_{t+1} = [\mathbf{q}_t + \mathbf{A}\mathbf{x}_t + \mathbf{b}_t]^+, \forall t$$

$$\mathbf{q}_1 \geq \mathbf{0}, \ \mathbf{q}_{T+1} = \mathbf{0} \quad (37)$$

where $\mathbf{q}_1$ is the given initial queue length, and $\mathbf{q}_{T+1} = \mathbf{0}$ guarantees that all workloads arrived have been served at the end of the scheduling horizon. Note that (37) is time-coupled, and generally challenging to solve without information of future workload arrivals and time-varying cost functions. Therefore, we reformulate (37) to fit our OCO formulation (1) by relaxing

the queue recursion in (37), namely

$$\mathbf{q}_{T+1} \geq \mathbf{q}_T + \mathbf{A}\mathbf{x}_T + \mathbf{b}_T \geq \mathbf{q}_1 + \sum_{t=1}^T (\mathbf{A}\mathbf{x}_t + \mathbf{b}_t) \quad (38)$$

which readily leads to $\sum_{t=1}^T (\mathbf{A}\mathbf{x}_t + \mathbf{b}_t) \leq \mathbf{q}_{T+1} - \mathbf{q}_1 \leq \mathbf{0}$, since $\mathbf{q}_1 \geq \mathbf{0}$ and $\mathbf{q}_{T+1} = \mathbf{0}$. Therefore, instead of solving (37), we aim to tackle a relaxed problem, given by

$$\min_{\{\mathbf{x}_t \in \mathcal{X}, \forall t\}} \sum_{t=1}^T f_t(\mathbf{x}_t) \quad \text{s.t. } \sum_{t=1}^T (\mathbf{A}\mathbf{x}_t + \mathbf{b}_t) \leq \mathbf{0} \quad (39)$$

where the workload flow conservation constraint $\mathbf{A}\mathbf{x}_t + \mathbf{b}_t \leq \mathbf{0}$ must be satisfied in the long term rather than slot-by-slot. Clearly, (39) is in the form of (1). Therefore, the MOSP algorithm of Section III can be leveraged to solve (39) in an *online* fashion, with provable performance and feasibility guarantees. Specifically, with $\mathbf{g}_t(\mathbf{x}_t) = \mathbf{A}\mathbf{x}_t + \mathbf{b}_t$, the primal update (8) boils down to a simple gradient update $\mathbf{x}_t = \mathcal{P}_{\mathcal{X}} \left(\mathbf{x}_{t-1} - \alpha \nabla f_{t-1}(\mathbf{x}_{t-1}) - \alpha \mathbf{A}^\top \boldsymbol{\lambda}_t \right)$, where $\mathcal{P}_{\mathcal{X}}(\cdot)$ defines projection onto the convex set $\mathcal{X}$. The dual update (9) is $\boldsymbol{\lambda}_{t+1} = \left[\boldsymbol{\lambda}_t + \mu(\mathbf{A}\mathbf{x}_t + \mathbf{b}_t) \right]^+$, which can be nicely regarded as a scaled version of the queue dynamics in (37), with $\mathbf{q}_t = \boldsymbol{\lambda}_t / \mu$.

In addition to simple closed-form updates, MOSP can also afford a fully decentralized implementation by exploiting the problem structure of network resource allocation, where each mapping node or data center decides the amounts on all its *outgoing links*, and only exchanges information with its *one-hop neighbors*. Per time slot t, the primal update at mapping node j includes variables on all its outgoing links, given by

$$x_t^{jk} = \left[x_{t-1}^{jk} - \alpha \nabla f_{t-1}^{jk}(x_{t-1}^{jk}) - \alpha \left(\lambda_t^k - \lambda_t^j \right) \right]_0^{\bar{x}^{jk}}, \ \forall k \in \mathcal{K} \quad (40a)$$

and the dual update reduces to

$$\lambda_{t+1}^j = \left[\lambda_t^j + \mu \left(b_t^j - \sum_{k \in \mathcal{K}} x_t^{jk} \right) \right]^+. \quad (40b)$$

Likewise, for data center k, the primal update becomes

$$y_t^k = \left[y_{t-1}^k - \alpha \left(\nabla f_{t-1}^k(y_{t-1}^k) - \lambda_t^k \right) \right]_0^{\bar{y}^k} \quad (40c)$$

where $[\cdot]_0^{\bar{y}^k} := \min\{\bar{y}^k, \max\{\cdot, 0\}\}$, and the dual recursion is

$$\lambda_{t+1}^k = \left[\lambda_t^k + \mu \left(\sum_{j \in \mathcal{J}} x_t^{jk} - y_t^k \right) \right]^+. \quad (40d)$$

Distributed MOSP for online network resource allocation is summarized in Algorithm 2.

B. Revisiting Stochastic Dual (sub) Gradient

The dynamic network resource allocation problem in Section IV-A has so far been studied in the stochastic setting [29], [35]. Classical approaches include Lyapunov optimization [24], [25] and the stochastic dual (sub)gradient method [26], both of which rely on stochastic approximation (SA) [27]. In the context of stochastic optimization, the time-varying vectors $\{\boldsymbol{\xi}_t\}$ with $\boldsymbol{\xi}_t := [\boldsymbol{\theta}_t^\top, \mathbf{b}_t^\top]^\top$ appearing in the cost and constraint are

---

Algorithm 2: MOSP for Online Resource Allocation.

1: **Initialize:** primal iterate $\mathbf{x}_0$, dual iterate $\boldsymbol{\lambda}_1$, and proper stepsizes α and μ.
2: **for** $t = 1, 2 \ldots$ **do**
3:     Each mapping node j performs (40a) and each data center k runs (40c).
4:     Mapping nodes and data centers observe local costs and workload arrivals.
5:     Each mapping node j performs (40b) and each data center k performs (40d).
6:     Mapping nodes (data centers) send multipliers to all neighboring data centers (mapping nodes).
7: **end for**

---

assumed to be independent realizations of a random variable Ξ.[2] In an SA-based stochastic optimization algorithm, per time t, a policy first observes a realization $\boldsymbol{\xi}_t$ of the random variable Ξ, and then (stochastically) selects an action $\mathbf{x}_t \in \mathcal{X}$. However, in contrast to minimizing the *observed cost* in the OCO setting, the goal of the stochastic resource allocation is usually to minimize the limiting average of the *expected cost* subject to the so-termed stability constraint, namely

$$\min_{\{\mathbf{x}_t \in \mathcal{X}, \mathbf{q}_t, \forall t\}} \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[f_t(\mathbf{x}_t)] \tag{41a}$$

$$\text{s.t.} \quad \mathbf{q}_{t+1} = [\mathbf{q}_t + \mathbf{A}\mathbf{x}_t + \mathbf{b}_t]^+, \forall t \tag{41b}$$

$$\lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[\mathbf{q}_t] \leq \mathbf{0} \tag{41c}$$

where he expectation in (41a) is taken over Ξ and the randomness of $\mathbf{x}_t$ and $\mathbf{q}_t$ induced by all possible sample paths $\{\boldsymbol{\xi}_1, \ldots, \boldsymbol{\xi}_t\}$ via (41b); and the stability constraint (41c) implies a finite bound on the accumulated constraint violation. In contrast to the observed costs in (37), each decision $\mathbf{x}_t$ is evaluated by all possible realizations in Ξ here. However, as $\mathbf{q}_t$ in (41b) couples the optimization variables over an infinite time horizon, (41) is intractable in general.

Prior works [25], [26], [29], [37] have demonstrated that (41) can be tackled via a tractable stationary relaxation, given by

$$\min_{\{\mathbf{x}_t \in \mathcal{X}, \forall t\}} \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[f_t(\mathbf{x}_t)] \tag{42a}$$

$$\text{s.t.} \quad \lim_{T \to \infty} \frac{1}{T} \sum_{t=1}^{T} \mathbb{E}[\mathbf{A}\mathbf{x}_t + \mathbf{b}_t] \leq \mathbf{0} \tag{42b}$$

where the time-coupling constraints (41b) and (41c) are relaxed to the limiting average constraint (42b). Such a relaxation can be verified similar to the queue relaxation in (38); see also [25]. Note that (42) is still challenging since it involves expectations in both costs and constraints, and the distribution of Ξ is usually unknown. Even if the joint probability distribution function

were available, finding the expectations would not scale with the dimensionality of Ξ. A common remedy is to use the stochastic dual gradient (SDG) iteration (a.k.a. Lyapunov optimization) [24], [25], [29]. Specifically, with $\boldsymbol{\lambda} \in \mathbb{R}_+^I$ denoting the multipliers associated with the expectation constraint (42b), the SDG method first observes one realization $\boldsymbol{\xi}_t$ at each slot t, and then performs the dual update as

$$\boldsymbol{\lambda}_{t+1} = [\boldsymbol{\lambda}_t + \mu(\mathbf{A}\mathbf{x}_t + \mathbf{b}_t)]^+, \forall t \tag{43}$$

where $\boldsymbol{\lambda}_t$ is the dual iterate at time t, $\mathbf{A}\mathbf{x}_t + \mathbf{b}_t$ is the stochastic dual gradient, and μ is a positive (and typically constant) stepsize. The actual allocation or the primal variable $\mathbf{x}_t$ appearing in (43) needs be found by solving the following sub-problems, one per slot t

$$\mathbf{x}_t \in \arg\min_{\mathbf{x} \in \mathcal{X}} f_t(\mathbf{x}) + \boldsymbol{\lambda}_t^\top (\mathbf{A}\mathbf{x} + \mathbf{b}_t). \tag{44}$$

For the considered network resource allocation problem, SDG in (43)–(44) entails a well-known cost-delay tradeoff [25]. Specifically, with f^* denoting the optimal objective (42), SDG can achieve an $\mathcal{O}(\mu)$-optimal solution such that $\lim_{T \to \infty}(1/T) \sum_{t=1}^{T} \mathbb{E}[f_t(\mathbf{x}_t)] \leq f^* + \mathcal{O}(\mu)$, and guarantee queue lengths[3] satisfying $\lim_{T \to \infty} (1/T) \sum_{t=1}^{T} \mathbb{E}[\|\mathbf{q}_t\|] = \mathcal{O}(1/\mu)$. Therefore, reducing the optimality gap $\mathcal{O}(\mu)$ will essentially increase the average network delay $\mathcal{O}(1/\mu)$.

Remark 5: The optimality of SDG is established relative to the offline optimal solution of (42), which can be thought as the time-average *optimality gap* in (25) under the OCO setting. Interestingly though, the optimality gap under the stochastic setting is equivalent to the (expected) dynamic regret (4), since their (expected) difference $V(\{\mathbb{E}[\mathcal{D}_t]\}_{t=1}^T)$ in (31) reduces to zero. To see this, note that $\mathbb{E}[f_t(\mathbf{x})]$ and $\mathbb{E}[\mathbf{A}\mathbf{x} + \mathbf{b}_t]$ are time-invariant, hence the dual problem of each per-slot subproblem in (42) is time-invariant. This reduction means that the SDG solver of the dynamic problem in (41) leverages its inherent stationarity (through the stationary dual problem), in contrast to the non-stationary nature of the OCO framework.

Remark 6: Below we highlight several differences of the novel MOSP in Algorithm 2 with the SDG recursion in (43)–(44) for the dynamic network resource allocation task.

(D1) From an operational perspective, SDG observes the current state $\boldsymbol{\xi}_t$ first, and then performs the resource allocation decision $\mathbf{x}_t$ accordingly. Therefore, at the beginning of slot t, SDG needs to precisely know the non-causal information $\boldsymbol{\xi}_t$. Inheriting the merits of OCO, on the other hand, MOSP operates in a fully *predictive* mode, which decides $\mathbf{x}_t$ without knowing the cost $f_t(\mathbf{x})$ and the constraint $\mathbf{g}_t(\mathbf{x})$ (or $\boldsymbol{\xi}_t$) at time t. This feature of MOSP is of major practical importance when costs and availability of resources are not available at the point of making decisions; e.g., online demand response in smart grids [32], [39] and resource allocation in wireless networking [40].

(D2) From a computational point of view, MOSP reduces to a simple saddle-point recursion with primal (projected) gradient descent and dual gradient ascent for the network resource allocation problem, both of which incur affordable complexity.

---

[2]Extension is also available when $\{\boldsymbol{\xi}_t\}$ constitute a sample path from an ergodic stochastic process $\{\Xi_t\}$, which converges to a stationary distribution; see e.g., [28], [36].

[3]According to Little's law [38], the time-average delay is proportional to the time-average queue length given the arrival rate.

However, the primal update of SDG in (44) generally requires solving a convex program per time slot t, which leads to much higher computational complexity in general.

(D3) With regards to the theoretical claims, the time-varying vector $\boldsymbol{\xi}_t$ in SDG typically requires a rather restrictive probabilistic assumption, to establish SDG optimality in either the ensemble average [25] or the limiting ergodic average sense [36]. In contrast, leveraging the OCO framework, MOSP admits finite-sample performance analysis with non-stochastic observed costs and constraints, which can even be adversarial.

C. Numerical Experiments

In this section, we provide numerical tests to demonstrate the merits of the proposed MOSP algorithm in the application of dynamic network resource allocation. Consider the geographical workload routing and allocation task in (39) with $J = 10$ mapping nodes and $K = 10$ data centers. The instantaneous network cost in (36) is

$$f_t(\mathbf{x}_t) := \sum_{k \in \mathcal{K}} p_t^k (y_t^k)^2 + \sum_{j \in \mathcal{J}} \sum_{k \in \mathcal{K}} c^{jk} (x_t^{jk})^2 \qquad (45)$$

where p_t^k is the energy price at data center k at time t, and c^{jk} is the per-unit bandwidth cost for transmitting from mapping node j to data center k. With the bandwidth limit $\bar{x}^{jk}$ uniformly randomly generated within $[10, 100]$, we set the bandwidth cost of each link (j, k) as $c^{jk} = 40/\bar{x}^{jk}, \forall j, k$. The resource capacities $\{\bar{y}^k, \forall k\}$ at all data centers are uniformly randomly generated from $[100, 200]$. We consider the following two cases for the time-varying parameters $\{p_t^k, \forall t, k\}$ and $\{b_t^j, \forall t, j\}$:

Case 1: Parameters $\{p_t^k, \forall t, k\}$ and $\{b_t^j, \forall t, j\}$ are independently drawn from time-invariant distributions. Specifically, p_t^k is uniformly distributed over $[1, 3]$, and the delay-tolerant workload b_t^j arrives at each mapping node j according to a uniform distribution over $[50, 150]$.

Case 2: Parameters $\{p_t^k, \forall t, k\}$ and $\{b_t^j, \forall t, j\}$ are generated according to non-stationary stochastic processes. Specifically, $p_t^k = \sin(\pi t/12) + n_t^k$ with i.i.d. noise n_t^k uniformly distributed over $[1, 3]$, while $b_t^j = 50\sin(\pi t/12) + v_t^j$ with i.i.d. noise v_t^j uniformly distributed over $[99, 101]$. One can verify that Assumption 5 is satisfied in this case, as the constraints vary slowly. Intuitively, it means that the network capacity margin is large relative to the temporal variation of arrival rates here.

Finally, with time horizon $T = 500$, the stepsize in (40a) and (40c) is set to $\alpha = 0.05/T^{1/3}$, and for (40b) and (40d) to $\mu = 50/T^{1/3}$. MOSP is benchmarked by three strategies: SDG in Section IV-B, the sequence of per-slot best minimizers in (5), and the offline optimal solution that solves (1) at once with all future costs and constraints available. Note that at the beginning of each slot t, the exact prices $\{p_t^k, \forall k\}$ and demands $\{b_t^j, \forall j\}$ for the coming slot are generally not available in practice [39]–[42]. Since the original SDG updates (43) and (44) require non-causal knowledge of $\{p_t^k, \forall k\}$ and $\{b_t^j, \forall j\}$ to decide $\mathbf{x}_t$, we modify them for fairness in this online setting by using the prices and demands at slot $t-1$ to obtain $\mathbf{x}_t$, which we term online dual gradient (ODG). As shown next, different constant stepsizes for ODG's dual update in (43) lead to quite different performance
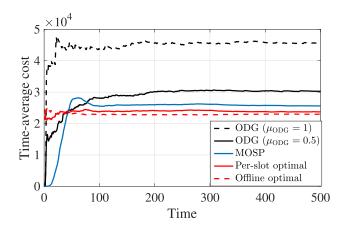

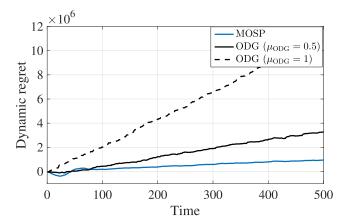Fig. 2. Time-average cost for Case 1.
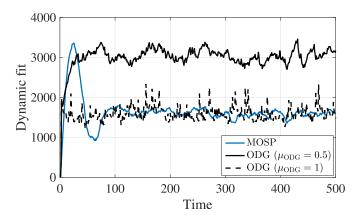

Fig. 3. Dynamic regret for Case 1.

Fig. 4. Dynamic fit for Case 1.

and feasibility behaviors; i.e., a larger stepsize results in higher regret but smaller fit, and vice versa. For this reason, ODG is studied under two different stepsizes: $\mu_{\text{ODG}} = 0.5$ balancing the regret and fit of ODG, and $\mu_{\text{ODG}} = 1$ allowing ODG to have similar fit with MOSP.

Figs. 2–4 show the test results for Case 1 under i.i.d. costs and constraints. Clearly, MOSP in Fig. 2 converges to a smaller time-average cost than ODG with the two stepsizes. The time-
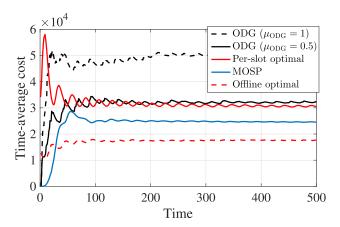
Fig. 5.　Time-average cost for Case 2.
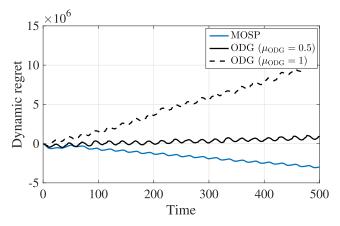

Fig. 7.　Dynamic fit for Case 2.

Fig. 6.　Dynamic regret for Case 2.

average cost of MOSP is slightly higher than the per-slot optimal solution, as well as the offline optimal solution with all information of the costs and constraints available over horizon T. Fig. 3 confirms the conclusion made from Fig. 2, where the dynamic regret (cf. (4)) of MOSP grows much slower than that of ODG. Regarding the dynamic fit (cf. (6)), Fig. 4 demonstrates that ODG with $\mu_{\mathrm{ODG}} = 1$ has a smaller fit than that of $\mu_{\mathrm{ODG}} = 0.5$, and similar to the dynamic fit of MOSP. According to the well-known trade-off between cost (optimality) and delay (constraint violations) in [25], increasing μ_{ODG} will improve the dynamic fit of ODG but degrade its dynamic regret. Therefore, MOSP is favorable in Case 1 since it has much smaller regret when its dynamic fit is similar to that of ODG with $\mu_{\mathrm{ODG}} = 1$. It is worth mentioning that theoretically speaking, the dynamic regret of MOSP may not be sub-linear in this i.i.d. case, since the accumulated cost and constraint variation is not necessarily small enough (cf. Theorem 2). However, MOSP is robust in this aspect at least for the numerical tests we carried.

Simulation tests using non-stationary costs and constraints are shown in Figs. 5–7. Different from Case 1, the time-average cost of MOSP is not only smaller than ODG, but also smaller than the per-slot optimum obtained via (3); see Fig. 5. A similar conclusion can be also drawn through the growths of dynamic regret in Fig. 6. From a high level, this is because the difference
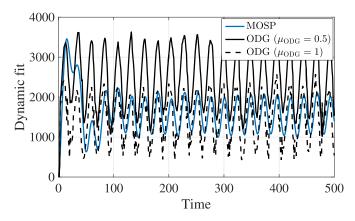
between the cost of the per-slot minimizers and that of the offline solutions is no longer small in the non-stationary case. Regarding Fig. 7, both ODG and MOSP have finite dynamic fits in the sense that the accumulated constraint violations do not increase with time. The dynamic fit of MOSP is much smaller than that of ODG with $\mu_{\mathrm{ODG}} = 0.5$, and comparable to that of ODG with $\mu_{\mathrm{ODG}} = 1$. Therefore, in this non-stationary case, MOSP also markedly outperforms ODG in both regret and fit.

V. CONCLUDING REMARKS

OCO with both adversarial costs and constraints has been studied in this paper. Different from existing works, the focus is on a setting where some of the constraints are revealed after taking actions, they are tolerable to instantaneous violations, but must be satisfied on average. Performance of the novel OCO algorithm is measured by: i) the difference of its objective relative to the best dynamic solution with one-slot-ahead information of the cost and the constraint (dynamic regret); and, ii) its accumulated amount of constraint violations (dynamic fit). It has been shown that the proposed MOSP algorithm adapts to the considered OCO setting with adversarial constraints. Under standard assumptions, MOSP simultaneously yields sub-linear dynamic regret and fit, if the accumulated variations of the per-slot minimizers and adversarial constraints are sub-linearly increasing with time. Algorithm design and performance analysis in this novel OCO setting, under adversarial constraints and with a dynamic benchmark, broaden the applicability of OCO to a wider application regime, which includes dynamic network resource allocation and online demand response in smart grids. Numerical tests demonstrated that the proposed algorithm outperforms state-of-the-art alternatives under different scenarios.

The novel OCO setting with long-term time-varying constraints opens up several interesting research directions. For instance, when nonlinear constraints are present, how to reduce the computational complexity of MOSP needs further investigation. Developing an adaptive version of MOSP with parameter-free implementation is also of particular interest.

APPENDIX

Before proving Theorems 1 and 2, we first bound the variation of the dual variable for the MOSP recursion (8)–(9). With the

dual drift defined as $\Delta(\boldsymbol{\lambda}_t) := \left(\|\boldsymbol{\lambda}_{t+1}\|^2 - \|\boldsymbol{\lambda}_t\|^2\right)/2$, we have the following lemma.

Lemma 1: Per slot t, the dual drift of the MOSP recursion (8)–(9) is upper-bounded as

$$\Delta(\boldsymbol{\lambda}_t) \le \mu \boldsymbol{\lambda}_t^\top \mathbf{g}_t(\mathbf{x}_t) + \frac{\mu^2}{2}\|\mathbf{g}_t(\mathbf{x}_t)\|^2. \tag{46}$$

Proof: Squaring the dual variable update (9), we have

$$\|\boldsymbol{\lambda}_{t+1}\|^2 = \left\|\left[\boldsymbol{\lambda}_t + \mu\mathbf{g}_t(\mathbf{x}_t)\right]^+\right\|^2 \le \|\boldsymbol{\lambda}_t + \mu\mathbf{g}_t(\mathbf{x}_t)\|^2$$

$$= \|\boldsymbol{\lambda}_t\|^2 + 2\mu\boldsymbol{\lambda}_t^\top \mathbf{g}_t(\mathbf{x}_t) + \mu^2\|\mathbf{g}_t(\mathbf{x}_t)\|^2. \tag{47}$$

The proof is complete after dividing both sides by 2. ∎

A. Proof of Theorem 1

The proof follows the steps in [13, Theorem 7], but generalizes the result from static regret with time-invariant constraints to dynamic regret with time-varying and long-term constraints. Recall that the primal iterate $\mathbf{x}_{t+1}$ is the optimal solution to the following optimization problem (cf. (8))

$$\min_{\mathbf{x}\in\mathcal{X}} h_t(\mathbf{x}) := \nabla f_t(\mathbf{x}_t)^\top(\mathbf{x} - \mathbf{x}_t) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x})$$

$$+ \frac{1}{2\alpha}\|\mathbf{x} - \mathbf{x}_t\|^2. \tag{48}$$

Then for any interior point $\tilde{\mathbf{x}}_t \in \mathcal{X}$ in Assumption 4, it follows that

$$\nabla f_t(\mathbf{x}_t)^\top(\mathbf{x}_{t+1} - \mathbf{x}_t) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_{t+1}) + \frac{1}{2\alpha}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2$$

$$\le \nabla f_t(\mathbf{x}_t)^\top(\tilde{\mathbf{x}}_t - \mathbf{x}_t) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\tilde{\mathbf{x}}_t) + \frac{1}{2\alpha}\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\|^2$$

$$\overset{(a)}{\le} \nabla f_t(\mathbf{x}_t)^\top(\tilde{\mathbf{x}}_t - \mathbf{x}_t) - \epsilon\boldsymbol{\lambda}_{t+1}^\top \mathbf{1} + \frac{1}{2\alpha}\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\|^2$$

$$\overset{(b)}{\le} \nabla f_t(\mathbf{x}_t)^\top(\tilde{\mathbf{x}}_t - \mathbf{x}_t) - \epsilon\|\boldsymbol{\lambda}_{t+1}\| + \frac{1}{2\alpha}\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\|^2 \tag{49}$$

where (a) follows by choosing $\tilde{\mathbf{x}}_t$ such that $\mathbf{g}_t(\tilde{\mathbf{x}}_t) \le -\epsilon\mathbf{1}$ and recalling the non-negativity of $\boldsymbol{\lambda}_{t+1}$; inequality (b) is because $\|\boldsymbol{\lambda}_{t+1}\| \le \boldsymbol{\lambda}_{t+1}^\top \mathbf{1}$ holds for any non-negative vector $\boldsymbol{\lambda}_{t+1}$.

Rearranging terms in (49), it follows that

$$\boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_{t+1}) \le \nabla f_t(\mathbf{x}_t)^\top(\tilde{\mathbf{x}}_t - \mathbf{x}_t) - \nabla f_t(\mathbf{x}_t)^\top(\mathbf{x}_{t+1} - \mathbf{x}_t)$$

$$- \epsilon\|\boldsymbol{\lambda}_{t+1}\| + \frac{1}{2\alpha}\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\|^2 - \frac{1}{2\alpha}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2$$

$$\overset{(c)}{\le} \nabla f_t(\mathbf{x}_t)^\top(\tilde{\mathbf{x}}_t - \mathbf{x}_t) - \nabla f_t(\mathbf{x}_t)^\top(\mathbf{x}_{t+1} - \mathbf{x}_t)$$

$$- \epsilon\|\boldsymbol{\lambda}_{t+1}\| + \frac{R^2}{2\alpha}$$

$$\overset{(d)}{\le} \|\nabla f_t(\mathbf{x}_t)\|\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\| + \|\nabla f_t(\mathbf{x}_t)\|\|\mathbf{x}_{t+1} - \mathbf{x}_t\|$$

$$- \epsilon\|\boldsymbol{\lambda}_{t+1}\| + \frac{R^2}{2\alpha} \overset{(e)}{\le} 2GR - \epsilon\|\boldsymbol{\lambda}_{t+1}\| + \frac{R^2}{2\alpha} \tag{50}$$

where (c) holds since $\mathcal{X}$ confines $\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\|^2 \le R^2$ and $\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \ge 0$; (d) uses the Cauchy-Schwartz inequality

twice; (e) leverages the bounds in Assumption 3, namely, $\|\nabla f_t(\mathbf{x}_t)\| \le G$, $\|\tilde{\mathbf{x}}_t - \mathbf{x}_t\| \le R$, and $\|\mathbf{x}_{t+1} - \mathbf{x}_t\| \le R$.

Plugging (50) into (46) in Lemma 1, we have

$$\Delta(\boldsymbol{\lambda}_{t+1}) \le \mu\boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_{t+1}(\mathbf{x}_{t+1}) + \frac{\mu^2}{2}\|\mathbf{g}_{t+1}(\mathbf{x}_{t+1})\|^2$$

$$\overset{(f)}{\le} \mu\boldsymbol{\lambda}_{t+1}^\top\left(\mathbf{g}_{t+1}(\mathbf{x}_{t+1}) - \mathbf{g}_t(\mathbf{x}_{t+1})\right) - \epsilon\mu\|\boldsymbol{\lambda}_{t+1}\|$$

$$+ 2\mu GR + \frac{\mu R^2}{2\alpha} + \frac{\mu^2 M^2}{2}$$

$$\overset{(g)}{\le} \mu\boldsymbol{\lambda}_{t+1}^\top\left[\mathbf{g}_{t+1}(\mathbf{x}_{t+1}) - \mathbf{g}_t(\mathbf{x}_{t+1})\right]^+ - \epsilon\mu\|\boldsymbol{\lambda}_{t+1}\|$$

$$+ 2\mu GR + \frac{\mu R^2}{2\alpha} + \frac{\mu^2 M^2}{2}$$

$$\overset{(h)}{\le} \mu\bar{V}(\mathbf{g})\|\boldsymbol{\lambda}_{t+1}\| - \epsilon\mu\|\boldsymbol{\lambda}_{t+1}\| + 2\mu GR + \frac{\mu R^2}{2\alpha} + \frac{\mu^2 M^2}{2} \tag{51}$$

where (f) uses the upper bound in Assumption 2 such that $\|\mathbf{g}_{t+1}(\mathbf{x}_{t+1})\| \le M$, (g) holds since $\boldsymbol{\lambda}_{t+1} \ge \mathbf{0}$, and (h) follows from the Cauchy-Schwartz inequality and the definition of the maximum variation $\bar{V}(\mathbf{g})$ in Assumption 5.

We prove the dual upper bound (12) by contradiction. Without loss of generality, suppose that $t+2$ is the first time that (12) does not hold. Therefore, we have

$$\|\boldsymbol{\lambda}_{t+1}\| \le \|\bar{\boldsymbol{\lambda}}\| = \mu M + \frac{2GR + R^2/(2\alpha) + (\mu M^2)/2}{\epsilon - \bar{V}(\mathbf{g})} \tag{52a}$$

and correspondingly

$$\|\boldsymbol{\lambda}_{t+2}\| > \|\bar{\boldsymbol{\lambda}}\| = \mu M + \frac{2GR + R^2/(2\alpha) + (\mu M^2)/2}{\epsilon - \bar{V}(\mathbf{g})}. \tag{52b}$$

In this case, it follows that

$$\|\boldsymbol{\lambda}_{t+1}\| \ge \|\boldsymbol{\lambda}_{t+2}\| - \|\boldsymbol{\lambda}_{t+2} - \boldsymbol{\lambda}_{t+1}\|$$

$$= \|\boldsymbol{\lambda}_{t+2}\| - \|[\boldsymbol{\lambda}_{t+1} + \mu\mathbf{g}_{t+1}(\mathbf{x}_{t+1})]^+ - \boldsymbol{\lambda}_{t+1}\|$$

$$\overset{(i)}{\ge} \|\boldsymbol{\lambda}_{t+2}\| - \|\mu\mathbf{g}_{t+1}(\mathbf{x}_{t+1})\|$$

$$\overset{(j)}{>} \frac{2GR + R^2/(2\alpha) + (\mu M^2)/2}{\epsilon - \bar{V}(\mathbf{g})} \tag{53}$$

where (i) is due to the non-expansive property of the projection operator, and inequality (j) uses (52b) and $\|\mathbf{g}_{t+1}(\mathbf{x}_{t+1})\| \le M$ in Assumption 2. However, since $\epsilon > \bar{V}(\mathbf{g})$, (51) implies that we have $\Delta(\boldsymbol{\lambda}_{t+1}) < 0$ if (53) holds. By definition of the dual drift, $\Delta(\boldsymbol{\lambda}_{t+1}) < 0$ implies that $\|\boldsymbol{\lambda}_{t+2}\| < \|\boldsymbol{\lambda}_{t+1}\|$, which contradicts (52a) and (52b). In addition, observe that the dual variable is initialized by $\boldsymbol{\lambda}_1 = \mathbf{0}$, and consequently $\|\boldsymbol{\lambda}_2\| \le \mu M$. Therefore, for every t, we have that $\|\boldsymbol{\lambda}_t\| \le \|\bar{\boldsymbol{\lambda}}\|$ holds.

Using the dual recursion in (9), it follows that $\boldsymbol{\lambda}_{T+1} \ge \boldsymbol{\lambda}_T + \mu\mathbf{g}_T(\mathbf{x}_T) \ge \boldsymbol{\lambda}_1 + \sum_{t=1}^T \mu\mathbf{g}_t(\mathbf{x}_t)$. Rearranging terms, we have

$$\sum_{t=1}^T \mathbf{g}_t(\mathbf{x}_t) \le \frac{\boldsymbol{\lambda}_{T+1}}{\mu} - \frac{\boldsymbol{\lambda}_1}{\mu} \le \frac{\boldsymbol{\lambda}_{T+1}}{\mu}. \tag{54}$$

With $\boldsymbol{\lambda}_{T+1} \geq \mathbf{0}$, (54) implies that $[\sum_{t=1}^{T} \mathbf{g}_t(\mathbf{x}_t)]^+ \leq \boldsymbol{\lambda}_{T+1}/\mu$, which completes the proof by taking norms on both sides and using the dual upper bound (12).

B. Proof of Theorem 2

With $h_t(\mathbf{x})$ defining the objective in (48), it can be shown that $h_t(\mathbf{x})$ is $1/\alpha$-strongly convex, which implies that for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^I$, we have [43, Theorem 2.1.8]

$$h_t(\mathbf{y}) \geq h_t(\mathbf{x}) + \nabla h_t(\mathbf{x})^\top (\mathbf{y} - \mathbf{x}) + \frac{1}{2\alpha}\|\mathbf{y} - \mathbf{x}\|^2. \quad (55)$$

Since $\mathbf{x}_{t+1}$ is the minimizer of the problem $\min_{\mathbf{x} \in \mathcal{X}} h_t(\mathbf{x})$, the optimality condition [33] implies that

$$\nabla h_t(\mathbf{x}_{t+1})^\top (\mathbf{y} - \mathbf{x}_{t+1}) \geq 0, \; \forall \mathbf{y} \in \mathcal{X}. \quad (56)$$

Setting $\mathbf{y} = \mathbf{x}_t^*$ and $\mathbf{x} = \mathbf{x}_{t+1}$ in (55), we have that (cf. (56))

$$h_t(\mathbf{x}_t^*) \geq h_t(\mathbf{x}_{t+1}) + \frac{1}{2\alpha}\|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2. \quad (57)$$

Hence, replacing $h_t(\mathbf{x})$ with the objective in (48) leads to

$$\nabla f_t(\mathbf{x}_t)^\top (\mathbf{x}_{t+1} - \mathbf{x}_t) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_{t+1}) + \frac{\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2}{2\alpha}$$
$$\stackrel{(a)}{\leq} \nabla f_t(\mathbf{x}_t)^\top (\mathbf{x}_t^* - \mathbf{x}_t) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_t^*) + \frac{\|\mathbf{x}_t^* - \mathbf{x}_t\|^2}{2\alpha}$$
$$- \frac{\|\mathbf{x}_{t+1} - \mathbf{x}_t^*\|^2}{2\alpha} \quad (58)$$

where (a) uses the strong convexity of the objective in (8); see also [13, Corollary 1]. Adding $f_t(\mathbf{x}_t)$ in (58) yields

$$f_t(\mathbf{x}_t) + \nabla f_t(\mathbf{x}_t)^\top (\mathbf{x}_{t+1} - \mathbf{x}_t) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_{t+1})$$
$$+ \frac{\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2}{2\alpha}$$
$$\leq f_t(\mathbf{x}_t) + \nabla f_t(\mathbf{x}_t)^\top (\mathbf{x}_t^* - \mathbf{x}_t) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_t^*)$$
$$+ \frac{\|\mathbf{x}_t^* - \mathbf{x}_t\|^2}{2\alpha} - \frac{\|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2}{2\alpha}$$
$$\stackrel{(b)}{\leq} f_t(\mathbf{x}_t^*) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_t^*) + \frac{\|\mathbf{x}_t^* - \mathbf{x}_t\|^2}{2\alpha} - \frac{\|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2}{2\alpha}$$
$$\stackrel{(c)}{\leq} f_t(\mathbf{x}_t^*) + \frac{\|\mathbf{x}_t^* - \mathbf{x}_t\|^2}{2\alpha} - \frac{\|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2}{2\alpha} \quad (59)$$

where (b) is due to the convexity of $f_t(\mathbf{x})$, and (c) comes from the fact that $\boldsymbol{\lambda}_{t+1} \geq \mathbf{0}$ and the per-slot optimal solution $\mathbf{x}_t^*$ is feasible (i.e., $\mathbf{g}_t(\mathbf{x}_t^*) \leq \mathbf{0}$) such that $\boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_t^*) \leq 0$.

Next, we bound the term $\nabla f_t(\mathbf{x}_t)^\top (\mathbf{x}_{t+1} - \mathbf{x}_t)$ by

$$-\nabla f_t(\mathbf{x}_t)^\top (\mathbf{x}_{t+1} - \mathbf{x}_t) \leq \|\nabla f_t(\mathbf{x}_t)\|\|\mathbf{x}_{t+1} - \mathbf{x}_t\| \quad (60)$$
$$\leq \frac{\|\nabla f_t(\mathbf{x}_t)\|^2}{2\eta} + \frac{\eta}{2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2 \stackrel{(d)}{\leq} \frac{G^2}{2\eta} + \frac{\eta}{2}\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2$$

where η is an arbitrary positive constant, and (d) is from the bound of gradients in Assumption 2. Plugging (60) into (59),

we have

$$f_t(\mathbf{x}_t) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_{t+1}) \leq f_t(\mathbf{x}_t^*) + \left(\frac{\eta}{2} - \frac{1}{2\alpha}\right)\|\mathbf{x}_{t+1} - \mathbf{x}_t\|^2$$
$$+ \frac{1}{2\alpha}\left(\|\mathbf{x}_t^* - \mathbf{x}_t\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2\right) + \frac{G^2}{2\eta}$$
$$\stackrel{(e)}{=} f_t(\mathbf{x}_t^*) + \frac{1}{2\alpha}\left(\|\mathbf{x}_t^* - \mathbf{x}_t\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2\right) + \frac{\alpha G^2}{2}$$
$$(61)$$

where (e) follows by choosing $\eta = 1/\alpha$ so that $\eta/2 - 1/(2\alpha) = 0$.

Using the dual drift bound (46) in Lemma 1 again, we have

$$\Delta(\boldsymbol{\lambda}_{t+1})/\mu + f_t(\mathbf{x}_t) \leq f_t(\mathbf{x}_t) + \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_{t+1})$$
$$+ \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_{t+1}(\mathbf{x}_{t+1}) - \boldsymbol{\lambda}_{t+1}^\top \mathbf{g}_t(\mathbf{x}_{t+1}) + \frac{\mu}{2}\|\mathbf{g}_{t+1}(\mathbf{x}_{t+1})\|^2$$
$$\stackrel{(f)}{\leq} f_t(\mathbf{x}_t^*) + \frac{1}{2\alpha}\left(\|\mathbf{x}_t^* - \mathbf{x}_t\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2\right)$$
$$+ \boldsymbol{\lambda}_{t+1}^\top (\mathbf{g}_{t+1}(\mathbf{x}_{t+1}) - \mathbf{g}_t(\mathbf{x}_{t+1})) + \frac{\mu\|\mathbf{g}_{t+1}(\mathbf{x}_{t+1})\|^2}{2} + \frac{\alpha G^2}{2}$$
$$\stackrel{(g)}{\leq} f_t(\mathbf{x}_t^*) + \frac{1}{2\alpha}\left(\|\mathbf{x}_t^* - \mathbf{x}_t\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2\right)$$
$$+ \boldsymbol{\lambda}_{t+1}^\top [\mathbf{g}_{t+1}(\mathbf{x}_{t+1}) - \mathbf{g}_t(\mathbf{x}_{t+1})]^+ + \frac{\mu M^2}{2} + \frac{\alpha G^2}{2}$$
$$\stackrel{(h)}{\leq} f_t(\mathbf{x}_t^*) + \frac{1}{2\alpha}\left(\|\mathbf{x}_t^* - \mathbf{x}_t\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2\right)$$
$$+ \|\boldsymbol{\lambda}_{t+1}\|V(\mathbf{g}_t) + \frac{\mu M^2}{2} + \frac{\alpha G^2}{2} \quad (62)$$

where (f) follows from (61); (g) uses non-negativity of $\boldsymbol{\lambda}_{t+1}$ and the gradient upper bound $\|\mathbf{g}_{t+1}(\mathbf{x})\| \leq M, \forall \mathbf{x} \in \mathcal{X}$; and (h) follows from the Cauchy-Schwartz inequality and the definition of the constraint variation $V(\mathbf{g}_t)$ in (16).

By interpolating intermediate terms in $\|\mathbf{x}_t^* - \mathbf{x}_t\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2$, we have that

$$\|\mathbf{x}_t^* - \mathbf{x}_t\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2$$
$$= \|\mathbf{x}_t^* - \mathbf{x}_t\|^2 - \|\mathbf{x}_t - \mathbf{x}_{t-1}^*\|^2 + \|\mathbf{x}_t - \mathbf{x}_{t-1}^*\|^2$$
$$- \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2$$
$$= \|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\|\|\mathbf{x}_t^* - 2\mathbf{x}_t + \mathbf{x}_{t-1}^*\| + \|\mathbf{x}_t - \mathbf{x}_{t-1}^*\|^2$$
$$- \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2$$
$$\stackrel{(i)}{\leq} 2R\|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\| + \|\mathbf{x}_t - \mathbf{x}_{t-1}^*\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2 \quad (63)$$

where (i) follows from the radius of $\mathcal{X}$ in Assumption 3 such that $\|\mathbf{x}_t^* - 2\mathbf{x}_t + \mathbf{x}_{t-1}^*\| \leq \|\mathbf{x}_t^* - \mathbf{x}_t\| + \|\mathbf{x}_t - \mathbf{x}_{t-1}^*\| \leq 2R$. Plugging (63) into (62), it readily leads to

$$\Delta(\boldsymbol{\lambda}_{t+1})/\mu + f_t(\mathbf{x}_t) \leq f_t(\mathbf{x}_t^*) + \|\boldsymbol{\lambda}_{t+1}\|V(\mathbf{g}_t) + \frac{\mu M^2}{2} + \frac{\alpha G^2}{2}$$
$$+ \frac{1}{2\alpha}\left(2R\|\mathbf{x}_t^* - \mathbf{x}_{t-1}^*\| + \|\mathbf{x}_t - \mathbf{x}_{t-1}^*\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2\right).$$
$$(64)$$

Summing up (64) over $t = 1, 2, \ldots, T$, we find

$$\sum_{t=1}^{T} \Delta(\boldsymbol{\lambda}_{t+1})/\mu + \sum_{t=1}^{T} f_t(\mathbf{x}_t)$$

$$\leq \sum_{t=1}^{T} f_t(\mathbf{x}_t^*) + \frac{1}{2\alpha} \sum_{t=1}^{T} \left(\|\mathbf{x}_t - \mathbf{x}_{t-1}^*\|^2 - \|\mathbf{x}_t^* - \mathbf{x}_{t+1}\|^2 \right)$$

$$+ \frac{RV(\{\mathbf{x}_t^*\}_{t=1}^T)}{\alpha} + \sum_{t=1}^{T} \|\boldsymbol{\lambda}_{t+1}\| V(\mathbf{g}_t) + \frac{\mu M^2 T}{2} + \frac{\alpha G^2 T}{2}$$

$$\overset{(j)}{\leq} \sum_{t=1}^{T} f_t(\mathbf{x}_t^*) + \frac{1}{2\alpha} \left(\|\mathbf{x}_1 - \mathbf{x}_0^*\|^2 - \|\mathbf{x}_T^* - \mathbf{x}_{T+1}\|^2 \right)$$

$$+ \frac{RV(\{\mathbf{x}_t^*\}_{t=1}^T)}{\alpha} + \|\bar{\boldsymbol{\lambda}}\| \sum_{t=1}^{T} V(\mathbf{g}_t) + \frac{\mu M^2 T}{2} + \frac{\alpha G^2 T}{2}$$

$$\overset{(k)}{\leq} \sum_{t=1}^{T} f_t(\mathbf{x}_t^*) + \frac{1}{2\alpha} \left(\|\mathbf{x}_1 - \mathbf{x}_0^*\|^2 \right) + \frac{RV(\{\mathbf{x}_t^*\}_{t=1}^T)}{\alpha}$$

$$+ \|\bar{\boldsymbol{\lambda}}\| V(\{\mathbf{g}_t\}_{t=1}^T) + \frac{\mu M^2 T}{2} + \frac{\alpha G^2 T}{2} \qquad (65)$$

where (j) uses the upper bound of $\|\boldsymbol{\lambda}_t\|$ in (12) that we define as $\|\bar{\boldsymbol{\lambda}}\|$, and (k) follows from the definition of accumulated variations $V(\{\mathbf{g}_t\}_{t=1}^T)$ in (16). The definition of dynamic regret in (4) finally implies that

$$\text{Reg}_T^d \leq \frac{RV(\{\mathbf{x}_t^*\}_{t=1}^T)}{\alpha} + \frac{\|\mathbf{x}_1 - \mathbf{x}_0^*\|^2}{2\alpha} + \|\bar{\boldsymbol{\lambda}}\| V(\{\mathbf{g}_t\}_{t=1}^T)$$

$$+ \frac{\mu M^2 T}{2} + \frac{\alpha G^2 T}{2} - \sum_{t=1}^{T} \frac{\Delta(\boldsymbol{\lambda}_{t+1})}{\mu}$$

$$= \frac{RV(\{\mathbf{x}_t^*\}_{t=1}^T)}{\alpha} + \frac{\|\mathbf{x}_1 - \mathbf{x}_0^*\|^2}{2\alpha} + \|\bar{\boldsymbol{\lambda}}\| V(\{\mathbf{g}_t\}_{t=1}^T)$$

$$+ \frac{\mu M^2 T}{2} + \frac{\alpha G^2 T}{2} - \frac{\|\boldsymbol{\lambda}_{T+2}\|^2}{2\mu} + \frac{\|\boldsymbol{\lambda}_2\|^2}{2\mu}$$

$$\overset{(l)}{\leq} \frac{RV(\{\mathbf{x}_t^*\}_{t=1}^T)}{\alpha} + \frac{R^2}{2\alpha} + \|\bar{\boldsymbol{\lambda}}\| V(\{\mathbf{g}_t\}_{t=1}^T)$$

$$+ \frac{\mu M^2 T}{2} + \frac{\alpha G^2 T}{2} + \frac{\mu M^2}{2} \qquad (66)$$

where (l) follows since: i) $\|\mathbf{x}_1 - \mathbf{x}_0^*\| \leq R$ due to the compactness of $\mathcal{X}$; ii) $\|\boldsymbol{\lambda}_{T+2}\|^2 \geq 0$; and, iii) $\|\boldsymbol{\lambda}_2\|^2 \leq \mu^2 M^2$ if $\boldsymbol{\lambda}_1 = \mathbf{0}$. This completes the proof.

REFERENCES

[1] T. Chen, Q. Ling, and G. B. Giannakis, "Online convex optimization for dynamic network resource allocation," in *Proc. Eur. Signal Process. Conf.*, Kos Island, Greece, Aug./Sep., 2017, pp. 146–150.

[2] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. Int. Conf. Mach. Learn.*, Washington D.C., Aug. 2003.

[3] 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.

[4] S. Shalev-Shwartz, "Online learning and online convex optimization," *Found. Trends Mach. Learn.*, vol. 4, no. 2, pp. 107–194, 2011.

[5] E. Hazan, "Introduction to online convex optimization," *Found. Trends Mach. Learn.*, vol. 2, nos. 3/4, pp. 157–325, 2016.

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

[7] 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.

[8] A. Jadbabaie, A. Rakhlin, S. Shahrampour, and K. Sridharan, "Online optimization: Competing with dynamic comparators," in *Proc. Int. Conf. Artif. Intell. Statist.*, San Diego, CA, USA, May 2015, pp. 398–406.

[9] A. Mokhtari, S. Shahrampour, A. Jadbabaie, and A. Ribeiro, "Online optimization in dynamic environments: Improved regret rates for strongly convex problems," in *Proc. IEEE Conf. Decision Control*, Las Vegas, NV, USA, Dec. 2016, pp. 7195–7201.

[10] N. Chen, J. Comden, Z. Liu, A. Gandhi, and A. Wierman, "Using predictions in online optimization: Looking forward with an eye on the past," in *Proc. ACM SIGMETRICS Int. Conf. Meas. Model. Comput. Sci.*, Antibes Juan-les-Pins, France, Jun. 2016, pp. 193–206.

[11] L. L. Andrew *et al.*, "A tale of two metrics: Simultaneous bounds on competitiveness and regret," in *Proc. Annu. Conf. Learn. Theory*, Princeton, NJ, USA, Jun. 2013, pp. 329–330.

[12] 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.

[13] H. Yu and M. J. Neely, "A low complexity algorithm with $\mathcal{O}(\sqrt{T})$ regret and constraint violations for online convex optimization with long term constraints," arXiv: 1604.02218, Apr. 2016.

[14] S. Paternain and A. Ribeiro, "Online learning of feasible strategies in unknown environments," *IEEE Trans. Automat. Control*, vol. 62, no. 6, pp. 2807–2822, Jun. 2017.

[15] A. Koppel, F. Y. Jakubiec, and A. Ribeiro, "A saddle point algorithm for networked online convex optimization," *IEEE Trans. Signal Process.*, vol. 63, no. 19, pp. 5149–5164, Oct. 2015.

[16] H. Wang and A. Banerjee, "Online alternating direction method," in *Proc. Int. Conf. Mach. Learn.*, Edinburgh, U.K., Jun. 2012, pp. 1119–1126.

[17] S. Shahrampour and A. Jadbabaie, "Distributed online optimization in dynamic environments using mirror descent," *IEEE Trans. Automat. Control*, doi: 10.1109/TAC.2017.2743462.

[18] S. H. Low and D. E. Lapsley, "Optimization flow control-I: Basic algorithm and convergence," *IEEE/ACM Trans. Netw.*, vol. 7, no. 6, pp. 861–874, Dec. 1999.

[19] L. Xiao, M. Johansson, and S. P. Boyd, "Simultaneous routing and resource allocation via dual decomposition," *IEEE Trans. Commun.*, vol. 52, no. 7, pp. 1136–1144, Jul. 2004.

[20] E. Ghadimi, I. Shames, and M. Johansson, "Multi-step gradient methods for networked optimization," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5417–5429, Nov. 2013.

[21] A. Beck, A. Nedic, A. Ozdaglar, and M. Teboulle, "An $\mathcal{O}(1/k)$ gradient method for network resource allocation problems," *IEEE Trans. Control Netw. Syst.*, vol. 1, no. 1, pp. 64–73, Mar. 2014.

[22] E. Wei, A. Ozdaglar, and A. Jadbabaie, "A distributed Newton method for network utility maximization-I: Algorithm," *IEEE Trans. Automat. Control*, vol. 58, no. 9, pp. 2162–2175, Sep. 2013.

[23] T. Chen, Y. Zhang, X. Wang, and G. B. Giannakis, "Robust workload and energy management for sustainable data centers," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 3, pp. 651–664, Mar. 2016.

[24] L. Tassiulas and A. Ephremides, "Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks," *IEEE Trans. Automat. Control*, vol. 37, no. 12, pp. 1936–1948, Dec. 1992.

[25] M. J. Neely, "Stochastic network optimization with application to communication and queueing systems," *Synthesis Lectures Commun. Netw.*, vol. 3, no. 1, pp. 1–211, 2010.

[26] A. G. Marques, L. M. Lopez-Ramos, G. B. Giannakis, J. Ramos, and A. J. Caamaño, "Optimal cross-layer resource allocation in cellular networks using channel-and queue-state information," *IEEE Trans. Veh. Technol.*, vol. 61, no. 6, pp. 2789–2807, Jul. 2012.

[27] A. Nemirovski, A. Juditsky, G. Lan, and A. Shapiro, "Robust stochastic approximation approach to stochastic programming," *SIAM J. Optim.*, vol. 19, no. 4, pp. 1574–1609, 2009.

[28] J. C. Duchi, A. Agarwal, M. Johansson, and M. I. Jordan, "Ergodic mirror descent," *SIAM J. Optim.*, vol. 22, no. 4, pp. 1549–1578, 2012.

[29] T. Chen, A. Mokhtari, X. Wang, A. Ribeiro, and G. B. Giannakis, "Stochastic averaging for constrained optimization with application to online resource allocation," *IEEE Trans. Signal Process.*, vol. 65, no. 12, pp. 3078–3093, Jun. 2017.

[30] M. Svorenova *et al.*, "Resource-performance trade-off analysis for mobile robots," arXiv:1609.04888, Sep. 2016.

[31] E. Dall'Anese and A. Simonetto, "Optimal power flow pursuit," *IEEE Trans. Smart Grid*, to appear, doi: 10.1109/TSG.2016.2571982

[32] W.-J. Ma, J. Wang, V. Gupta, and C. Chen, "Distributed energy management for networked microgrids using online alternating direction method of multipliers with regret," *IEEE Trans. Smart Grid*, to appear, doi: 10.1109/TSG.2016.2569604

[33] D. P. Bertsekas, *Nonlinear Programming*. Belmont, MA, USA: Athena Scientific, 1999.

[34] K.-S. Jun, F. Orabona, R. Willett, and S. Wright, "Improved strongly adaptive online learning using coin betting," in *Proc. Int. Conf. Artif. Intell. Statist.*, Fort Lauderdale, FL, USA, Apr. 2017, pp. 943–951.

[35] T. Chen, X. Wang, and G. B. Giannakis, "Cooling-aware energy and workload management in data centers via stochastic optimization," *IEEE J. Sel. Topics Signal Process.*, vol. 10, no. 2, pp. 402–415, Mar. 2016.

[36] A. Ribeiro, "Ergodic stochastic optimization algorithms for wireless communication and networking," *IEEE Trans. Signal Process.*, vol. 58, no. 12, pp. 6369–6386, Dec. 2010.

[37] L. Georgiadis, M. Neely, and L. Tassiulas, "Resource allocation and cross-layer control in wireless networks," *Found. Trends Netw.*, vol. 1, pp. 1–144, 2006.

[38] J. D. Little, "A proof for the queuing formula: L = λw," *Oper. Res.*, vol. 9, no. 3, pp. 383–387, Jun. 1961.

[39] S. J. Kim and G. Giannakis, "An online convex optimization approach to real-time energy pricing for demand response," *IEEE Trans. Smart Grid*, to appear, doi: 10.1109/TSG.2016.2539948.

[40] J. Tadrous, A. Eryilmaz, and H. El Gamal, "Proactive resource allocation: Harnessing the diversity and multicast gains," *IEEE Trans. Inf. Theory*, vol. 59, no. 8, pp. 4833–4854, Aug. 2013.

[41] "Midcontinent independent system operator (MISO) locational marginal price," Nov. 2016. [Online]. Available: http://www.misoenergy.org/MarketsOperations/Prices

[42] L. Huang, S. Zhang, M. Chen, and X. Liu, "When backpressure meets predictive scheduling," *IEEE/ACM Trans. Netw.*, vol. 24, no. 4, pp. 2237–2250, Aug. 2016.

[43] Y. Nesterov, *Introductory Lectures on Convex Optimization: A Basic Course*, vol. 87. Berlin, Germany: Springer, 2013.

Tianyi Chen (S'14) received the B.Eng. degree (highest Hons.) in communication science and engineering from Fudan University, Shanghai, China, and the M.Sc. degree in electrical and computer engineering (ECE) from the University of Minnesota (UMN), Minneapolis, MN, USA, in 2014 and 2016, respectively. He is currently working toward the Ph.D. degree at UMN. His research interests include online learning, online convex optimization, and stochastic network optimization with applications to smart grids, sustainable cloud networks, and Internet-of-Things. He received the National Scholarship from China in 2013, UMN ECE Department Fellowship in 2014, and the UMN Doctoral Dissertation Fellowship in 2017.

Qing Ling (SM'15) received the B.E. degree in automation and the 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 Postdoctoral Research Fellow in the Department of Electrical and Computer Engineering, Michigan Technological University, Houghton, MI, USA, from 2006 to 2009, and an Associate Professor in the Department of Automation, University of Science and Technology of China from 2009 to 2017. He is currently a Professor in the School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, China. His current research interests include decentralized network optimization and its applications. He is an Associate Editor of IEEE SIGNAL PROCESSING LETTERS.

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

From 1982 to 1986, he was with the University of Southern California. From 1987 to 1998, he was with the University of Virginia. Since 1999, he has been a Professor with the University of Minnesota, Minneapolis, MN, USA, where he holds an Endowed Chair in Wireless Telecommunications, the University of Minnesota McKnight Presidential Chair in ECE, and serves as the Director of the Digital Technology Center. His general interests include the areas of communications, networking, and statistical signal processing—subjects on which he has published more than 400 journal papers, 700 conference papers, 25 book chapters, two edited books, and two research monographs (h-index 127). His current research interests include learning from Big Data, wireless cognitive radios, and network science with applications to social, brain, and power networks with renewables. He is the coinventor of 30 patents issued. He received eight best paper awards from the IEEE Signal Processing (SP) and Communications Societies, including the G. Marconi Prize Paper Award in Wireless Communications. He also received the Technical Achievement Awards from the SP Society in 2000, from 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 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.