Periodic Updates for Constrained OCO with Application to Large-Scale Multi-Antenna Systems

Juncheng Wang, *Student Member, IEEE*, Min Dong, *Senior Member, IEEE*, Ben Liang, *Fellow, IEEE*, and Gary Boudreau, *Senior Member, IEEE*

Abstract—In many dynamic systems, decisions on system operation are updated over time, and the decision maker requires an online learning approach to optimize its strategy in response to the changing environment. When the loss and constraint functions are convex, this belongs to the general family of online convex optimization (OCO). In existing OCO works, the environment is assumed to vary in a time-slotted fashion, while the decisions are updated at each time slot. However, many wireless communication systems permit only periodic decision updates, *i.e.,* each decision is fixed over multiple time slots, while the environment changes between the decision epochs. The standard OCO model is inadequate for these systems. Therefore, in this work, we consider periodic decision updates for OCO. We aim to minimize the accumulation of time-varying convex loss functions, subject to both short-term and long-term constraints. Feedback information about the loss functions within the current update period may be delayed and incomplete. We propose an efficient algorithm, termed Periodic Queueing and Gradient Aggregation (PQGA), which employs novel periodic queues together with possibly multi-step aggregated gradient descent to update the decisions over time. We derive upper bounds on the dynamic regret, static regret, and constraint violation of PQGA. As an example application, we study the performance of PQGA for network virtualization in a large-scale multi-antenna system shared by multiple wireless service providers. Simulation results show that PQGA converges fast and substantially outperforms the current best alternative.

Index Terms—Online convex optimization, long-term constraint, periodic updates, massive MIMO, wireless network virtualization.

1 Introduction

In many signal processing, resource allocation, and machine learning problems, system parameters and loss functions vary over time under dynamic environments. Online learning has emerged as a promising solution to these problems in the presence of uncertainty, where an online decision strategy iteratively adapts to system variations based on historical information [2]. Online convex optimization (OCO) is a subclass of online learning, where the loss and constraint functions are convex with respect to (w.r.t.) the decision [3]. OCO can be seen as a sequential decision-making process between a decision maker and the system. Under the standard OCO setting, at the beginning of each time slot, the decision maker selects a decision from a convex feasible set. Only at the end of each time slot, the system reveals information about the current convex loss function to the decision maker. The goal of the decision maker is to minimize the cumulative loss. Such an OCO framework has many applications, *e.g.*, wireless transmit covariance matrix design [4], dynamic network resource allocation [5], and smart grids with renewable energy supply [6].

In OCO, due to the lack of in-time information about the current convex loss function, the decision maker cannot select an optimal decision at each time slot. Instead, the decision maker aims at minimizing the *regret* [7], *i.e.,* the performance gap between the online decision sequence and some performance benchmark. Most of the early OCO algorithms were evaluated in terms of the *static* regret, which compares the online decision sequence with a static offline benchmark that has apriori information of all the convex loss functions. However, when the environment changes drastically, the static offline benchmark itself may perform poorly. In this case, the static regret may not be a meaningful performance measurement anymore. A more useful *dynamic* regret measures the performance gap between the online decision sequence and a time-varying sequence of per-time-slot optimizers given knowledge of the current convex loss function. The dynamic regret has been recognized as a more attractive but harder-to-track performance measurement for OCO.

In many practical systems, the decision maker often collects the system parameters and makes decisions in a *periodic* manner, *e.g.,* to limit the computation and communication overhead. One application of interest is precoding design in massive multiple-input multiple-output (MIMO) systems, where the precoder is updated based on delayed channel state information (CSI) feedback and is fixed for a period, *i.e.*, one or multiple resource block durations, while the underlying channel may fluctuate fast over time. The resource block duration is fixed in Long-Term Evolution (LTE) and is allowed to change over time for a more flexible network operation in 5G New Radio (NR) [8]. In mobile

- *Juncheng Wang and Ben Liang are with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 1A1, Canada (e-mail: jcwang@ece.utoronto.ca; liang@ece.utoronto.ca).*
- *Min Dong is with the Department of Electrical, Computer and Software Engineering, Ontario Tech University, Oshawa, ON L1G 0C5, Canada (e-mail: min.dong@ontariotechu.ca).*
- *Gary Boudreau is with Ericsson Canada, Ottawa, ON K2K 2V6, Canada (e-mail: gary.boudreau@ericsson.com).*
- *This work was supported in part by Ericsson Canada, in part by the Natural Sciences and Engineering Research Council of Canada, and in part by the Ontario Centre of Innovation.*
- *A preliminary version of this work was presented in IEEE International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), 2020 [1] (DOI: 10.1109/SPAWC48557.2020.9154228).*

edge computing [9], due to the offloading and scheduling latency, the cloud server may periodically collect the offloading tasks from the remote devices and design a fixed computing resource allocation strategy for a certain time period.

However, to the best of our knowledge, all existing works on OCO require both the decision and feedback information are updated at each time slot. Motivated by this discrepancy, in this work, we consider a new constrained OCO problem with *periodic updates*, where the decision maker periodically collects information feedbacks and makes online decisions to minimize the accumulated loss. The duration of update period can be multiple time slots and can vary over time. In the presence of periodic updates, no other known work provides regret bound analysis for OCO.

Furthermore, we consider both short-term and long-term constraints, which are important in many practical optimization problems. For example, in communication systems, the short-term constraint can represent the maximum transmit power, while the long-term power constraint can be seen as a limit on energy usage. An effective constrained OCO algorithm should also bound the *constraint violation*, which is the accumulated violation on the long-term constraints. The need to provide the constraint violation bound further adds to the challenges of regret bound analysis.

The main contributions of this paper are as follows:

- We formulate a new constrained OCO problem with periodic updates. Each update period may last for multiple time slots and may vary over time. At the beginning of each update period, the decision maker selects a decision, fixed for the period, to minimize the accumulated loss subject to both short-term and long-term constraints. The feedback information about the loss functions can be delayed for multiple time slots and partly missing. As explained above, this constrained OCO framework with periodic updates has broad applications in practical communication and computation systems.
- We propose an efficient algorithm, termed Periodic Queueing and Gradient Aggregation (PQGA) for the formulated constrained OCO problem. In PQGA, we propose a novel construction of *periodic queues*, which converts the accumulated constraint violation in an update period into queue dynamics. Furthermore, PQGA collects and aggregates the delayed gradient feedbacks in each update period to minimize the accumulated loss. The periodic queues, together with gradient aggregation, improve the efficacy of periodic decision updates and facilitate the performance bounding of PQGA.
- We analyze the performance of PQGA and study the impact of the periodic queues and gradient aggregation. We prove that PQGA yields $\mathcal{O}(\max\{T^{\frac{1+\nu}{2}}, T^{\delta}\})$ dynamic regret, $\mathcal{O}(\max\{T^{\frac{1}{2}}, T^{\delta}\})$ static regret, and $\mathcal{O}(T^{\frac{1}{2}})$ constraint violation, where T is the time horizon, ν represents the growth rate of the accumulated variation of the per-time-slot optimizer, and δ measures the level of variation of the update period. We further show that, when the number of gradient descent steps within each update period is sufficiently large, PQGA provides improved $\mathcal{O}(\max\{T^{\nu}, T^{\delta}\})$ dynamic regret and $\mathcal{O}(1)$ constraint violation. For the special case of per-time-slot updates, PQGA achieves $\mathcal{O}(T^{\nu})$ dynamic regret and $\mathcal{O}(1)$ constraint violation bound.
- As an application, we use PQGA to solve an online precoding design problem in massive MIMO systems with multiple wireless service providers, where all the antennas and wireless spectrum resources are simultaneously shared by the service providers. In this case, we show that PQGA only involves low-complexity closed-form computation. Simulation results show that PQGA converges fast and substantially outperforms the current best alternative.

Organizations: The rest of this paper is organized as follows. In Section 2, we present the related work. Section 3 describes the mathematical model, problem formulation, and performance measurement for constrained OCO with periodic updates. We present PQGA, derive its performance bounds, and discuss its performance merits in Section 4. The application of PQGA to large-scale multi-antenna systems with multiple wireless service providers is presented in Section 5. Simulation results are provided in Section 6, followed by concluding remarks in Section 7.

Notations: The transpose, Hermitian transpose, complex conjugate, trace, Euclidean norm, Frobenius norm, L_∞ norm, and L_1 norm of a matrix $\mathbf{A}$ are denoted by $\mathbf{A}^T$, $\mathbf{A}^H$, $\mathbf{A}^*$, $\mathrm{tr}\{\mathbf{A}\}$, $\|\mathbf{A}\|$, $\|\mathbf{A}\|_F$, $\|\mathbf{A}\|_\infty$, and $\|\mathbf{A}\|_1$, respectively. The notation $\mathrm{blkdiag}\{\mathbf{A}_1, \ldots, \mathbf{A}_n\}$ denotes a block diagonal matrix with diagonal elements being matrices $\mathbf{A}_1, \ldots, \mathbf{A}_n$, $\mathbb{E}\{\cdot\}$ denotes expectation, $\Re\{\cdot\}$ denotes the real part of the enclosed parameter, $\mathbf{I}$ denotes an identity matrix, and $\mathbf{g} \sim \mathcal{CN}(\mathbf{0}, \sigma^2\mathbf{I})$ means that $\mathbf{g}$ is a circular complex Gaussian random vector with mean $\mathbf{0}$ and variance $\sigma^2\mathbf{I}$.

2 Related Work

In this section, we survey existing works on online learning and online optimization. The differences between the existing literature and our work are summarized in Table 1.

2.1 Online Learning and OCO

Online learning is a method of machine learning, where a learner attempts to tackle some decision-making task by learning from a sequence of data instances. As an important subclass of online learning, OCO has been applied in various areas such as wireless communications [4], cloud networks [5], and smart grids [6]. In the seminal work of OCO [7], a simple projected gradient descent algorithm achieved $\mathcal{O}(T^{\frac{1}{2}})$ static regret [7]. The static regret was further improved to $\mathcal{O}(\log T)$ for strongly convex loss functions [10]. Moreover, [11] and [12] examined the static regret for OCO where information feedbacks of the loss functions are delayed for multiple time slots.

The analysis of static regret was extended to that of the more attractive dynamic regret in [7], [13], [14] for general convex loss functions. Moreover, strong convexity was shown to improve the dynamic regret bound in [15]. By increasing the number of gradient descent steps, the

dynamic regret bound was further improved in [16]. Furthermore, [17] studied the impact of inexact gradient on the dynamic regret bound.

2.2 OCO with Long-Term Constraints

The above OCO works [7], [10]-[17] focused on online problems with short-term constraints represented by a feasible set that must be strictly satisfied. However, long-term constraints arise in many practical applications such as energy control in wireless communications, queueing stability in cloud networks, and power balancing in smart grids. Existing algorithms for OCO with long-term constraints can be categorized into saddle-point-typed algorithms [18]-[21] and virtual-queue-based algorithms [22]-[25].

A saddle-point-typed algorithm was first proposed in [18] and achieved $\mathcal{O}(T^{\frac{1}{2}})$ static regret and $\mathcal{O}(T^{\frac{3}{4}})$ constraint violation. A follow-up work [19] provided $\mathcal{O}(T^{\max\{\mu,1-\mu\}})$ static regret and $\mathcal{O}(T^{1-\frac{\mu}{2}})$ constraint violation, where $\mu \in (0,1)$ is some trade-off parameter. This recovers the performance bounds in [18] as a special case. In the presence of multi-slot delay, [20] achieved $\mathcal{O}(T^{\frac{1}{2}})$ static regret and $\mathcal{O}(T^{\frac{3}{4}})$ constraint violation. The saddle-point-typed algorithm was further modified in [21] with dynamic regret analysis.

As an alternative to saddle-point-typed algorithms, virtual queues can be used to represent the backlog of constraint violation, which facilitates performance bounding through the analysis of a drift-plus-penalty (DPP) like expression. A virtual-queue-based algorithm was first proposed in [22] and established $\mathcal{O}(T^{\frac{1}{2}})$ static regret and $\mathcal{O}(1)$ constraint violation for OCO with fixed long-term constraints. For stochastic constraints that are independent and identically distributed (i.i.d.), another virtual-queue-based algorithm in [23] achieved $\mathcal{O}(T^{\frac{1}{2}})$ static regret and $\mathcal{O}(T^{\frac{1}{2}})$ constraint violation simultaneously. In [24], the virtual-queue-based algorithm was further extended to provide a dynamic regret bound. The impact of multi-slot feedback delay on constrained OCO was considered in [25] with both dynamic and static regret analyses.

However, all of the above works on constrained OCO [18]-[25] are under the standard per-time-slot update setting. No other known work considers periodic updates for OCO. Furthermore, these works only performs single-step gradient descent at each time slot, which does not take full advantage of the potential computational capacity to improve the system performance. In this paper, we propose PQGA, which uses novel periodic queues with possibly multi-step aggregated gradient descent to update the online decision. We believe this is the first of its kind.

A part of this work has appeared as a short paper that focuses only on the application of constrained OCO with period updates to large-scale multi-antenna systems [1]. In the current manuscript, we have substantially extended our prior work, generalizing the PQGA algorithm, enabling multi-step gradient descent, deriving new regret and constraint violation bounds over time-varying update periods, and providing other new derivation, proofs, and simulation results.

2.3 Other Online Approaches

The general Lyapunov optimization technique has been applied to develop online schemes for various applications such as resource allocation [26], mobile computing [27], and smart grid [28]. Note that PQGA is substantially different from the conventional DPP algorithm for Lyapunov optimization [29] in both the decision update and the virtual queue update. Lyapunov optimization makes use of the system state and queueing information to implicitly learn and adapt to system variation with unknown statistics. The standard Lyapunov optimization framework is limited to per-time-slot updates [29]. It has been extended to accommodate variable renewal frames in [30]. However, this approach requires the system states to be i.i.d. or Markovian, while the OCO framework does not have such restriction. Furthermore, [30] assumes the system state to be fixed within each renewal frame, while we allow the loss function to change at each time slot within an update period.

Furthermore, the standard Lyapunov optimization relies on the current accurate system state for decision updates [29]. When the system state feedback is delayed, one can apply Lyapunov optimization by using the historical information to predict the current system state with some error [31]. However, this way of dealing with feedback delay is equivalent to extending Lyapunov optimization to inaccurate system states [32], [33]. In this case, the optimality gap would be $\mathcal{O}(\sigma T)$, where σ is some inaccuracy measure.

The multi-armed bandit (MAB) approach [34] and the partially observed Markov decision process (POMDP) [35] have also been used in various works for online decision making [36]-[38]. Our OCO approach is substantially different from these approaches in a few aspects. First, the OCO approach does not require any assumption on the distribution of the system information. In contrast, the MAB approach generally requires the system distribution to be fixed, and the POMDP approach generally assumes the system states evolve as a Markov process. Also, our OCO approach can deal with both long-term and short-term constraints, while the MAB and POMDP approaches generally cannot handle long-term constraints. Finally, OCO algorithms generally perform gradient descent to minimize the loss function, since the gradient of loss function is often easy to compute. In contrast, the MAB and POMDP approaches minimize the loss function directly, which can be of high computational complexity if the loss function is complicated.

3 Constrained OCO with Periodic Updates

In this section, we first detail the mathematical model of constrained OCO with periodic updates. Then, we present the performance metrics, including the static regret, dynamic regret, and constraint violation, for performance measurement.

3.1 OCO Problem Formulation

We consider a time-slotted system with time indexed by t. Let $f_t(\mathbf{x}) : \mathbb{R}^n \to \mathbb{R}$ be a general loss function at time slot $t \in \mathcal{T} = \{0, \ldots, T-1\}$, where T is the total time slots considered for the problem. The loss function $f_t(\mathbf{x})$

TABLE 1
Summary of Related Works

Reference	Type of benchmark	Long-term constraint	Periodic updates
[7]	Static and dynamic	No	No
[10]-[12]	Static	No	No
[13]-[17]	Dynamic	No	No
[18]-[20], [22], [23]	Static	Yes	No
[21], [24]	Dynamic	Yes	No
[25]	Static and dynamic	Yes	No
PQGA	Static and dynamic	Yes	Yes

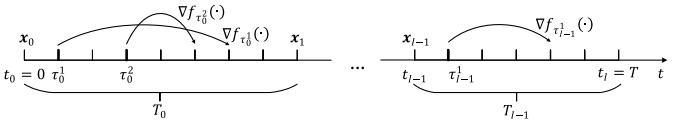


Fig. 1. A timeline illustrating OCO with periodic updates.

is convex and may change arbitrarily over time. The exact expression of the loss function depends on the specific applications under consideration. Let $\mathbf{x}_t \in \mathbb{R}^n$ be the decision vector at time slot t. The goal of the decision maker is to provide online decisions $\mathbf{x}_t$ under the given constraints on $\{\mathbf{x}_t, t \in \mathcal{T}\}$ and available feedback to minimize the accumulated loss $\sum_{t\in\mathcal{T}} f_t(\mathbf{x}_t)$.

In standard OCO, it is assumed that the decision maker can update $\mathbf{x}_t$ at any $t \in \mathcal{T}$, and information feedback is received at each t. However, as explained in Section 1, these two assumptions are often not possible to satisfy in many practical systems. Therefore, in this paper, we consider the more realistic scenario where the decision maker can only update decisions once in several time slots, or in other words, periodically. Furthermore, we assume information feedback may be delayed and incomplete. Following these, we consider periodic decision updates for OCO. Specifically, based on delayed and possibly incomplete information feedback, the decision maker selects a decision in each update period, which remains fixed until the next update, to minimize the accumulated loss while satisfying the constraints.

Suppose the time horizon of T time slots is segmented into I update periods, as shown in Fig. 1. Each update period $i \in \mathcal{I} = \{0, \ldots, I-1\}$ has a duration of $T_i \in \{1, \ldots, T_{\max}\}$ time slots with $T_{\max}$ being the maximum duration of an update period. We have $T = \sum_{i\in\mathcal{I}} T_i$. The update period is known at the decision maker. Let t_i represent the beginning time slot of update period i. The decision vector is updated at the beginning of time slot t_i. For ease of exposition, we slightly abuse the notation and use $\mathbf{x}_i$ to denote this decision vector. It remains unchanged within update period i, *i.e.*, $\mathbf{x}_t = \mathbf{x}_i$ for any $t \in \mathcal{T}_i = \{t_i, t_i+1, \ldots, t_i+T_i-1\}$, $i \in \mathcal{I}$. Under this new per-period update setting, the accumulated loss becomes $\sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}_i} f_t(\mathbf{x}_i)$.

Let $\nabla f_t(\cdot)$, $t \in \mathcal{T}_i$, be the possible gradient information within update period i.[1] We assume that there are $S_i \in \{1, \ldots, T_i\}$ gradient feedbacks received by the decision maker within update period i. Let τ_i^s, $s \in \mathcal{S}_i = \{1, \ldots, S_i\}$, represent the time slot at which the s-th gradient feedback in update period i, denoted by $\nabla f_{\tau_i^s}(\cdot)$, is sent. The decision maker receives $\nabla f_{\tau_i^s}(\cdot)$ after some delay that can last for multiple time slots. Any feedback received after the next decision $\mathbf{x}_{i+1}$ will be dropped. Due to random delays, the gradient feedbacks may be received out of order.

Let $\mathcal{X}_0 \subseteq \mathbb{R}^n$ be a compact convex set that represents the short-term constraints for any $\mathbf{x}_t$, $t \in \mathcal{T}$. Besides $\mathcal{X}_0$, we also consider long-term constraints on $\{\mathbf{x}_t\}$, which arise in many practical applications as discussed in Section 1. Let $\mathbf{g}(\mathbf{x}) = [g^1(\mathbf{x}), \ldots, g^C(\mathbf{x})]^T : \mathbb{R}^n \to \mathbb{R}^C$ be a vector of C constraint functions. Then, the decision sequence is subject to long-term constraints given in a vector form $\sum_{t\in\mathcal{T}} \mathbf{g}(\mathbf{x}_t) \preceq \mathbf{0}$. Under periodic decision updates $\{\mathbf{x}_i\}$, the long-term constraints are equivalent to $\sum_{i\in\mathcal{I}} T_i \mathbf{g}(\mathbf{x}_i) \preceq \mathbf{0}$.

Thus, the goal of constrained OCO with periodic updates is to select a sequence of decisions $\{\mathbf{x}_i\}$, to minimize the accumulated loss functions while meeting both short-term and long-term constraints. This leads to the following optimization problem:

$$\mathbf{P1}: \quad \min_{\{\mathbf{x}_i\}} \quad \sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}_i} f_t(\mathbf{x}_i)$$
$$\text{s.t.} \quad \sum_{i\in\mathcal{I}} T_i \mathbf{g}(\mathbf{x}_i) \preceq \mathbf{0}, \tag{1}$$
$$\mathbf{x}_i \in \mathcal{X}_0, \quad \forall i \in \mathcal{I}. \tag{2}$$

Different from existing works on OCO with only short-term constraints [7], [10]-[17], the additional long-term constraints in (1) of **P1** lead to a more complicated online optimization problem. The periodic decision updates add more complication to the problem as the underlying system varies over time while the online decision is fixed for a period. Note that in the special case when update period $T_i = 1$ for any $i \in \mathcal{I}$, **P1** reduces to the standard constrained OCO problem with per-time-slot updates as in [18], [19], [22].

3.2 Performance Metric

Due to the lack of in-time feedback of the current loss functions under the OCO setting, an optimal solution to **P1** cannot be obtained.[2] We consider the performance measurements typically adopted in the literature for developing the solution for constrained OCO, namely static and dynamic regrets, with a slight modification tailored to periodic updates.

We aim at designing a decision sequence $\{\mathbf{x}_i\}$ over update periods, such that the accumulated loss in the objective of **P1** is competitive with some benchmark under the same

1. Gradient feedbacks are common in the OCO literature, since OCO algorithms are usually based on gradient descent. In practice, gradient feedbacks can be obtained in various methods depending on the application. For our application to massive MIMO systems with multiple service providers, we use the CSI feedbacks to obtain the gradients.

2. In fact, even for the simplest original OCO problem [7] (*i.e.*, under the per-time-slot update setting without long-term constraints (1)), an optimal solution cannot be found [10].

TABLE 2
Summary of Key Notations

Notation	Description
$\mathcal{T}$	Set of time slots
$\mathcal{I}$	Set of update periods
$\mathcal{T}_i$	Set of time slots in update period i
$\mathcal{S}_i$	Set of feedbacks in update period i
T	Total number of time slots
I	Total number of update periods
J	Total number of gradient descent steps
T_i	Duration of update period i
$T_{\max}$	Maximum duration of an update period.
t_i	Beginning time slot of update period i
S_i	Number of feedbacks in update period i
τ_i^s	Time slot of the s-th feedback in update period i
$f_t(\mathbf{x})$	Loss function at time slot t
$\nabla f_{\tau_i^s}(\cdot)$	The s-th gradient feedback in update period
$\mathcal{X}_0$	Set that represents the short-term constraints
$\mathbf{g}(\mathbf{x})$	Vector of C long-term constraint functions
$g^c(\mathbf{x})$	The c-th long-term constraint function
$\mathbf{x}_i$	Decision vector in update period i
$\tilde{\mathbf{x}}_i^j$	Decision after j-step gradient descent in period i
$\mathbf{x}^\star$	Static benchmark
$\{\mathbf{x}_i^\circ\}$	Dynamic benchmark
$\{\mathbf{x}_i^\star\}$	Dynamic benchmark subject to $\mathcal{X}_0$ only
$\mathrm{RE_s}(T)$	Static regret over T time slots
$\mathrm{RE_d}(T)$	Dynamic regret over T time slots
$\mathrm{VO}^c(T)$	Violation of the c-th constraint over T time slots
$\mathbf{Q}_i$	Virtual queue vector in update period i
Q_i^c	The c-th virtual queue in update period i
L_i	Quadratic Lyapunov function in update period i
Δ_i	Lyapunov drift in update period i
α, η, γ	PQGA algorithm parameters
ϱ	Strongly convex constant of $f_t(\mathbf{x})$
L	Smoothness constant of $f_t(\mathbf{x})$
D	Bound on the gradient $\nabla f_t(\mathbf{x})$
β	Lipschitz continuous constant of $\mathbf{g}(\mathbf{x})$
G	Bound on $\mathbf{g}(\mathbf{x})$
ϵ	Bound on the interior point of $\mathbf{g}(\mathbf{x})$
R	Bound on the radius of $\mathcal{X}_0$
ρ	Gradient descent contraction constant
$\Pi_{\mathbf{x}^\circ}$	Accumulated variation of $\{\mathbf{x}_i^\circ\}$
Π_T	Accumulated variation of $\{T_i\}$
Π_∇	Accumulate squared norm of gradient
$\Pi_{\mathbf{x}}$	Accumulated distance between $\{\mathbf{x}_i^\circ\}$ and $\{\mathbf{x}_i^\star\}$
ν	Time variation measure of $\Pi_{\mathbf{x}^\circ}$
δ	Time variation measure of Π_T

set of gradient feedbacks. Thus, for the static regret, we consider the following static offline benchmark, which is generalized from the per-time-slot one used in [18]-[20], [22], [23] to accommodate periodic updates:

$$\mathbf{x}^\star \triangleq \arg\min_{\mathbf{x}\in\mathcal{X}} \sum_{i\in\mathcal{I}} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} f_{\tau_i^s}(\mathbf{x}) \tag{3}$$

where $\mathcal{X} \triangleq \{\mathbf{x} \in \mathcal{X}_0 : \mathbf{g}(\mathbf{x}) \preceq \mathbf{0}\}$. Note that $\mathbf{x}^\star$ is computed offline assuming all the loss functions $f_{\tau_i^s}(\mathbf{x})$, for all $s \in \mathcal{S}_i$ and $i \in \mathcal{I}$, are known in advance. Then, the static regret is the performance gap between $\{\mathbf{x}_i\}$ and $\mathbf{x}^\star$:

$$\mathrm{RE_s}(T) \triangleq \sum_{i\in\mathcal{I}} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} \left(f_{\tau_i^s}(\mathbf{x}_i) - f_{\tau_i^s}(\mathbf{x}^\star)\right). \tag{4}$$

However, the static regret only provides a coarse performance measure when the underlying system is time-varying and may not be an attractive metric to use. A more useful performance benchmark is the dynamic benchmark $\{\mathbf{x}_i^\circ\}$, given by

$$\mathbf{x}_i^\circ \triangleq \arg\min_{\mathbf{x}\in\mathcal{X}} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} f_{\tau_i^s}(\mathbf{x}). \tag{5}$$

For the case of per-time-slot updates, the dynamic benchmark has been originally proposed for OCO with short-term constraints [7] and has been further modified in [21], [24], [25] to incorporate long-term constraints. Here, we generalize it to account for periodic updates. In (5), $\mathbf{x}_i^\circ$ is computed using all the S_i loss functions $f_{\tau_i^s}(\mathbf{x})$ in the current update period i. Then, the dynamic regret is

$$\mathrm{RE_d}(T) \triangleq \sum_{i\in\mathcal{I}} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} \left(f_{\tau_i^s}(\mathbf{x}_i) - f_{\tau_i^s}(\mathbf{x}_i^\circ)\right). \tag{6}$$

The gap between the static and dynamic regrets can be as large as $\mathcal{O}(T)$ [39]. In this paper, for comprehensive performance analysis, we provide upper bounds for both $\mathrm{RE_s}(T)$ and $\mathrm{RE_d}(T)$.

Note that with incomplete gradient feedbacks, *i.e.*, $\sum_{i\in\mathcal{I}} S_i = S < T$, our regret definitions in (4) and (6) fully utilize the feedback information. Our accumulated loss in each period i is the average loss over those time slots when the gradient feedbacks are provided, *i.e.*, $\frac{1}{S_i}\sum_{s\in\mathcal{S}_i} f_{\tau_i^s}(\mathbf{x}_i)$, multiplied by the duration T_i of the update period i, for $i \in \mathcal{I}$. If the environment is mean stationary, *i.e.*, $\mathbb{E}\{f_t(\mathbf{x})\} = \mathbb{E}\{f_{t'}(\mathbf{x})\}$ for any $t, t' \in \mathcal{T}$, then in the expectation sense, the accumulated loss in our regret definitions is the same as that in the objective of **P1**, *i.e.*,

$$\mathbb{E}\left\{\sum_{i\in\mathcal{I}} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} \left(f_{\tau_i^s}(\mathbf{x}_i)\right)\right\} = \mathbb{E}\left\{\sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}_i} \left(f_t(\mathbf{x}_i)\right)\right\}.$$

In general, suppose there exists a constant $d > 0$ such that $|f_t(\mathbf{x}) - f_{t'}(\mathbf{x})| \le d$, for any $\mathbf{x} \in \mathcal{X}_0$ and $t, t' \in \mathcal{T}_i$, $i \in \mathcal{I}$. Then, the gap between the accumulated loss in the regret definition and that in the objective is $\sum_{i\in\mathcal{I}} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} (f_{\tau_i^s}(\mathbf{x}_i)) - \sum_{i\in\mathcal{I}} \sum_{t\in\mathcal{T}_i} (f_t(\mathbf{x}_i)) = \mathcal{O}(d(T - S))$. This measurement gap can be small if the system does not fluctuate too much over time. Furthermore, the gap approaches zero as the amount of feedback increases to become complete, *i.e.*, $S \to T$.

Besides the accumulated loss, we also need to measure the accumulated violation of each long-term constraint $c \in \mathcal{C} = \{1, \ldots, C\}$. Define the constraint violation as

$$\mathrm{VO}^c(T) \triangleq \sum_{i\in\mathcal{I}} T_i g^c(\mathbf{x}_i), \quad \forall c \in \mathcal{C}. \tag{7}$$

We point out that the constraint violation defined in [18]-[25] is under the standard per-time-slot update setting. In contrast, the constraint violation defined in (7) is in a more general form that can accommodate updating periods with varying durations.

4 The Periodic Queueing and Gradient Aggregation (PQGA) Algorithm

We now present an efficient algorithm, PQGA, to solve the constrained OCO problem **P1**. The algorithm uses a

periodic virtual queue for the online decisions to satisfy the long-term constraints. It updates the online decision in each update period by solving a per-period optimization problem that is convex and hence practically solvable. We then show that, despite being a simple algorithm, PQGA provides provable performance guarantees in terms of dynamic regret, static regret, and constraint violation bounds.

4.1 PQGA Algorithm Description

In PQGA, we introduce a *periodic* virtual queue vector $\mathbf{Q}_i = [Q_i^1, \ldots, Q_i^C]^T$ in each update period $i \in \mathcal{I}$, with the following updating rule:

$$Q_{i+1}^c = \max\{-\gamma T_i g^c(\mathbf{x}_i), Q_i^c + \gamma T_i g^c(\mathbf{x}_i)\}, \quad \forall c \in \mathcal{C} \quad (8)$$

where $\gamma > 0$ is an algorithm parameter. The role of $\mathbf{Q}_i$ is similar to the Lagrange multiplier vector for the long-term constraints in (1), and the value of $\mathbf{Q}_i$ reflects the accumulated violation of the long-term constraints. In (8), $T_i g^c(\mathbf{x}_i)$ is the accumulated constraint violation in update period i, which is then scaled by an appropriate factor γ. This way of forming a virtual queue is unique to our proposed approach. We point out here that Q_i^c in (8) is different from the virtual queues used in the standard Lyapunov optimization [29], as the maximization in (8) is taken over the negative constraint violation $-\gamma T_i g^c(\mathbf{x}_i)$ in stead of 0 in the standard form. The periodic virtual queue is also different from existing works on constrained OCO [22]-[25] that update the virtual queue at each slot.

In the basic form of PQGA, instead of solving **P1** directly, we solve a *per-period* problem at the beginning of each update period $i+1$ for $\mathbf{x}_{i+1}$ with the short-term constraints only. It is given by

$$\mathbf{P2}: \min_{\mathbf{x}\in\mathcal{X}_0} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} [\nabla f_{\tau_i^s}(\mathbf{x}_i)]^T(\mathbf{x}-\mathbf{x}_i) + \alpha\|\mathbf{x}-\mathbf{x}_i\|^2 + [\mathbf{Q}_{i+1} + \gamma T_i \mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x})]$$

where $\alpha, \gamma > 0$ are algorithm parameters. In the first term of the above objective function, the gradient direction is aggregated based on all the gradient feedbacks $\{\nabla f_{\tau_i^s}(\cdot), s \in \mathcal{S}_i\}$ collected in the previous update period i. The second term $\alpha\|\mathbf{x}-\mathbf{x}_i\|^2$ is a regularization term, which controls how much the new decision $\mathbf{x}_{i+1}$ can change from the previous decision $\mathbf{x}_i$. The last term is an inner-product between the predicted queue length in the next update period based on $\mathbf{x}_i$ and the weighted accumulated constraint violation in update period $i+1$. It represents the penalty of constraint violation in $\mathbf{g}(\mathbf{x})$. As such, we convert the long-term constraints in (1) into a penalty term on $\mathbf{g}(\mathbf{x})$ as one part of the objective function in **P2**.

The basic form of PQGA uses a single step of gradient descent as shown in **P2**. In addition to this basic form, we can further modify PQGA to enable multi-step gradient descent. Multi-step gradient descent has previously been shown to reduce the growth rate of the dynamic regret for OCO with short-term constraints [16]. In this work, we will verify that it also improves the performance of PQGA under both short-term and long-term constraints. Specifically, at the beginning of each update period $i+1$, after updating the periodic virtual queue in (8), we initialize an intermediate decision vector $\tilde{\mathbf{x}}_i^0 = \mathbf{x}_i$. We then perform J-step aggregated gradient descent to generate $\tilde{\mathbf{x}}_i^J$, for $J \geq 0$. If $J = 0$, we readily have $\tilde{\mathbf{x}}_i^J = \mathbf{x}_i$ from initialization. Otherwise, for each gradient descent step $j \in \mathcal{J} = \{1, \ldots, J\}$, we solve the following optimization problem for $\tilde{\mathbf{x}}_i^j$:

$$\min_{\mathbf{x}\in\mathcal{X}_0} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} [\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^{j-1})]^T(\mathbf{x}-\tilde{\mathbf{x}}_i^{j-1}) + \alpha\|\mathbf{x}-\tilde{\mathbf{x}}_i^{j-1}\|^2.$$

The above problem is similar to the standard projected gradient descent problem. Therefore, its solution is readily available:

$$\tilde{\mathbf{x}}_i^j = \mathcal{P}_{\mathcal{X}_0}\left\{\tilde{\mathbf{x}}_i^{j-1} - \frac{1}{2\alpha}\left(\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^{j-1})\right)\right\} \quad (9)$$

where $\mathcal{P}_{\mathcal{X}_0}\{\mathbf{x}\} \triangleq \arg\min_{\mathbf{y}\in\mathcal{X}_0}\{\|\mathbf{y}-\mathbf{x}\|^2\}$ is the projection operator onto the convex feasible set $\mathcal{X}_0$, and α can be viewed as the step-size parameter.

With both $\mathbf{x}_i$ and $\tilde{\mathbf{x}}_i^J$, we modify **P2** and have the following per-period optimization problem for $\mathbf{x}_{i+1}$:

$$\mathbf{P2}': \min_{\mathbf{x}\in\mathcal{X}_0} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} [\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)]^T(\mathbf{x}-\tilde{\mathbf{x}}_i^J) + \alpha\|\mathbf{x}-\tilde{\mathbf{x}}_i^J\|^2 + \eta\|\mathbf{x}-\mathbf{x}_i\|^2 + [\mathbf{Q}_{i+1} + \gamma T_i \mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x})]$$

where $\alpha, \eta, \gamma > 0$. Note that there are two regularization terms in **P2′** for $\mathbf{x}_i$ and $\tilde{\mathbf{x}}_i^J$, respectively. The intuition behind the double regularization is that both $\mathbf{x}_i$ and $\tilde{\mathbf{x}}_i^J$ provide useful information in minimizing the accumulate loss and constraint violation. Therefore, it is beneficial for the new decision $\mathbf{x}_{i+1}$ to be not too far away from either of $\mathbf{x}_i$ or $\tilde{\mathbf{x}}_i^J$. In Section 4.2, this double regularization will be analytically shown to provide PQGA a substantial performance advantage over the existing methods (or algorithms) in terms of performance bounds.

The PQGA algorithm is summarized in Algorithm 1. During each update period $i \in \mathcal{I}$, the decision maker collects the delayed and possibly incomplete gradient information $\nabla f_{\tau_i^s}(\cdot), s \in \mathcal{S}_i$. At the beginning of the next update period $i+1$, it first computes the accumulated constraint violation caused by its previous decision $\mathbf{x}_i$ and updates the periodic virtual queue $\mathbf{Q}_{i+1}$ in (8). It then learns the gradient descent direction from the collected gradient feedbacks over period i and performs J-step aggregated gradient descent to generate $\tilde{\mathbf{x}}^J$. Finally, based on gradient descent direction, updated virtual queue, and both $\tilde{\mathbf{x}}_i^J$ and $\mathbf{x}_i$, the per-period optimization problem $\mathbf{P2}'$ is solved to compute the decision $\mathbf{x}_{i+1}$.[3] Note that PQGA has four algorithm parameters α, η, γ, and J. We leave the discussion on the choice of their values to Section 4.3, where after deriving the performance bounds, we explain the impact of these parameters on these bounds.

Algorithm Complexity Discussion

The computational complexity of the PQGA algorithm mainly lies in the computational complexity in solving **P2′** (or **P2**). It depends on the specific form of loss function $f_t(\mathbf{x})$, constraint functions $\mathbf{g}(\mathbf{x})$, and feasible set $\mathcal{X}_0$. Note

3. When $J = 0$, the double regularization in **P2′** on $\tilde{\mathbf{x}}^J$ and $\mathbf{x}_i$ can be combined into a single regularization on $\mathbf{x}_i$, and **P2′** is reduced to **P2**.

Algorithm 1 The PQGA Algorithm

1: **Initialization**: $\alpha, \eta, \gamma > 0$, $J \geq 0$, $\mathbf{x}_0 \in \mathcal{X}_0$, and $\mathbf{Q}_0 = \mathbf{0}$.
2: At the beginning of each update period $i+1$, do:
3: Update the periodic virtual queue $\mathbf{Q}_{i+1}$ via (8).
4: Initialize the intermediate decision vector $\tilde{\mathbf{x}}_i^0 = \mathbf{x}_i$.
5: **for** $j = 1$ **to** J
6: Update $\tilde{\mathbf{x}}^j$ via (9).
7: **end for**
8: Update the periodic decision $\mathbf{x}_{i+1}$ by solving **P2'** using $\mathbf{Q}_{i+1}$, $\mathbf{x}_i$, and $\tilde{\mathbf{x}}^J$.

that the objective functions in **P2'** and **P2** are strongly convex. Thus, in general, they can be solved efficiently using existing convex optimization solvers. In Section 5, we consider the application of PQGA to a massive MIMO system visualization problem with multiple service providers, where we will provide detailed computational complexity analysis of PQGA in this application. In particular, we show that in this case, **P2'** has a closed-form solution for the decision at each period update, and thus PQGA has negligible computational complexity.

We also note that, if the constraint function $\mathbf{g}(\mathbf{x})$ is separable w.r.t. $\mathbf{x}$, **P2** and **P2'** can be decomposed into independent subproblems of smaller size to solve. In this case, PQGA can be implemented in a distributive fashion with potentially significantly lower computational complexity.

4.2 Regret and Constraint Violation Bounds for PQGA

Existing analysis techniques for the standard per-time-slot OCO setting with single-step gradient descent [18]-[25] are inadequate for studying the performance of PQGA. In this section, we present new techniques to derive the regret and constraint violation bounds for PQGA. They are developed to account for the periodic virtual queues and multi-step aggregated gradient descent in the analysis. Although a small part of our derivations uses techniques from Lyapunov drift analysis, PQGA is structurally different from Lyapunov optimization as explained in Section 2.

We first make the following assumptions that are common in the literature of OCO.

Assumption 1. For any t, the loss function $f_t(\mathbf{x})$ satisfies the following:

1.1) $f_t(\mathbf{x})$ is 2ϱ-strongly convex over $\mathcal{X}_0$: $\exists\ \varrho > 0$, s.t., for any $\mathbf{x}, \mathbf{y} \in \mathcal{X}_0$ and t

$$f_t(\mathbf{y}) \geq f_t(\mathbf{x}) + [\nabla f_t(\mathbf{x})]^T(\mathbf{y}-\mathbf{x}) + \varrho\|\mathbf{y}-\mathbf{x}\|^2. \quad (10)$$

1.2) $f_t(\mathbf{x})$ is $2L$-smooth over $\mathcal{X}_0$: $\exists\ L > 0$, s.t., for any $\mathbf{x}, \mathbf{y} \in \mathcal{X}_0$ and t

$$f_t(\mathbf{y}) \leq f_t(\mathbf{x}) + [\nabla f_t(\mathbf{x})]^T(\mathbf{y}-\mathbf{x}) + L\|\mathbf{y}-\mathbf{x}\|^2. \quad (11)$$

Assumption 2. The gradient $\nabla f_t(\mathbf{x})$ is bounded: $\exists\ D > 0$, s.t.,

$$\|\nabla f_t(\mathbf{x})\| \leq D, \quad \forall \mathbf{x} \in \mathcal{X}_0, \quad \forall t \in \mathcal{T}. \quad (12)$$

Assumption 3. The long-term constraint functions satisfy the following:

3.1) $\mathbf{g}(\mathbf{x})$ is Lipschitz continuous on $\mathcal{X}_0$: $\exists\ \beta > 0$, s.t.,

$$\|\mathbf{g}(\mathbf{x}) - \mathbf{g}(\mathbf{y})\| \leq \beta\|\mathbf{x}-\mathbf{y}\|, \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}_0. \quad (13)$$

3.2) $\mathbf{g}(\mathbf{x})$ is bounded: $\exists\ G > 0$, s.t.,

$$\|\mathbf{g}(\mathbf{x})\| \leq G, \quad \forall \mathbf{x} \in \mathcal{X}_0. \quad (14)$$

3.3) Existence of an interior point: $\exists\ \epsilon > 0$ and $\mathbf{x}' \in \mathcal{X}_0$, s.t.,

$$\mathbf{g}(\mathbf{x}') \preceq -\epsilon\mathbf{1}. \quad (15)$$

Assumption 4. The radius of $\mathcal{X}_0$ is bounded: $\exists\ R > 0$, s.t.,

$$\|\mathbf{x}-\mathbf{y}\| \leq R, \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}_0. \quad (16)$$

Note that the assumption of the loss function $f_t(\mathbf{x})$ being strongly convex in Assumption 1 holds in many practical problems. Strongly convex loss functions arise in many machine learning and signal processing applications, such as Lasso regression, support vector machine, softmax classifier, and robust subspace tracking. Furthermore, for applications with general convex loss functions, it is common to add a simple regularization term such as $\mu\|\mathbf{x}\|^2$, so that the overall optimization objective becomes strongly convex [17].

4.2.1 Bounding the Dynamic Regret

A main goal of this paper is to analyze the impact of variable update periods and multi-step aggregated gradient descent on the dynamic regret for constrained OCO, which has not been studied in the existing literature. To this end, we define the accumulated variation of the dynamic benchmark $\{\mathbf{x}_i^\circ\}$ (referred to as the path length in [7]) as

$$\Pi_{\mathbf{x}^\circ} \triangleq \sum_{i\in\mathcal{I}} \|\mathbf{x}_i^\circ - \mathbf{x}_{i+1}^\circ\|. \quad (17)$$

Also, we define the accumulated variation of the variable update periods $\{T_i\}$ as

$$\Pi_T \triangleq \sum_{i\in\mathcal{I}} (T_i - T_{i+1})^2. \quad (18)$$

We first provide bounds on the periodic virtual queues $\{\mathbf{Q}_i\}$ produced by PQGA in the following lemma.

Lemma 1. The following statements hold for any $i \in \mathcal{I}$:

$$\mathbf{Q}_i \succeq \mathbf{0}, \quad (19)$$
$$\mathbf{Q}_{i+1} + \gamma T_i \mathbf{g}(\mathbf{x}_i) \succeq \mathbf{0}, \quad (20)$$
$$\|\mathbf{Q}_{i+1}\| \geq \|\gamma T_i \mathbf{g}(\mathbf{x}_i)\|, \quad (21)$$
$$\|\mathbf{Q}_{i+1}\| \leq \|\mathbf{Q}_i\| + \|\gamma T_i \mathbf{g}(\mathbf{x}_i)\|. \quad (22)$$

Proof: The periodic virtual queue vector is initialized as $\mathbf{Q}_0 = \mathbf{0}$. For any $c \in \mathcal{C}$ and $i \in \mathcal{I}$, by induction, we first assume $Q_i^c \geq 0$. From the periodic virtual queue dynamics in (8), if $\gamma T_i g^c(\mathbf{x}_i) \geq 0$, then $Q_{i+1}^c \geq Q_i^c + \gamma T_i g^c(\mathbf{x}_i) \geq 0$; otherwise, we have $Q_{i+1}^c \geq -\gamma T_i g^c(\mathbf{x}_i) \geq 0$. Combining the above two cases, we have (19).

From (8), for any $c \in \mathcal{C}$ and $i \in \mathcal{I}$, we have $Q_{t+1}^c \geq -\gamma T_i g^c(\mathbf{x}_i)$, which yields (20).

For any $c \in \mathcal{C}$ and $i \in \mathcal{I}$, from (8) and $Q_i^c \geq 0$ in (19), if $\gamma T_i g^c(\mathbf{x}_i) \geq 0$, then $Q_{i+1}^c \geq Q_i^c + \gamma T_i g^c(\mathbf{x}_i) \geq \gamma T_i g^c(\mathbf{x}_i)$; otherwise, we have $Q_{i+1}^c \geq -\gamma T_i g^c(\mathbf{x}_i)$. Therefore, we have $Q_{i+1}^c \geq |\gamma T_i g^c(\mathbf{x}_i)|$. Squaring both sides and summing over $c \in \mathcal{C}$ yields (21).

From (8), for any $c \in \mathcal{C}$ and $i \in \mathcal{I}$, we have $Q_{i+1}^c \leq Q_i^c + |\gamma T_i g^c(\mathbf{x}_i)|$. Since $Q_i^c \geq 0$ in (19), by the triangle inequality, we have $\|\mathbf{Q}_{i+1}\| \leq \sqrt{\sum_{c\in\mathcal{C}}(Q_i^c + |\gamma T_i g^c(\mathbf{x}_i)|)^2} \leq \|\mathbf{Q}_i\| + \|\gamma T_i \mathbf{g}(\mathbf{x}_i)\|$, which yields (22). ■

Define $L_i \triangleq \frac{1}{2}\|\mathbf{Q}_i\|^2$ as the quadratic Lyapunov function and $\Delta_i \triangleq L_{i+1} - L_i$ as the Lyapunov drift for each update period $i \in \mathcal{I}$. Based on Lemma 1, we provide an upper bound on the Lyapunov drift Δ_i in the following lemma.

Lemma 2. The Lyapunov drift is upper bounded for any $i \in \mathcal{I}$ as follows:

$$\Delta_i \leq \mathbf{Q}_i^T[\gamma T_i \mathbf{g}(\mathbf{x}_i)] + \|\gamma T_i \mathbf{g}(\mathbf{x}_i)\|^2. \tag{23}$$

Proof: For any $c \in \mathcal{C}$ and $i \in \mathcal{I}$, we first prove

$$\frac{1}{2}(Q_{i+1}^c)^2 - \frac{1}{2}(Q_i^c)^2 \leq Q_i^c[\gamma T_i g^c(\mathbf{x}_i)] + [\gamma T_i g^c(\mathbf{x}_i)]^2 \tag{24}$$

by considering the following two cases.

1) $Q_i^c + \gamma T_i g^c(\mathbf{x}_i) \geq -\gamma T_i g^c(\mathbf{x}_i)$*:* From the virtual queue dynamics in (8), we have $Q_{i+1}^c = Q_i^c + \gamma T_i g^c(\mathbf{x}_i)$. It then follows that

$$\begin{aligned}\frac{1}{2}(Q_{i+1}^c)^2 &= \frac{1}{2}[Q_i^c + \gamma T_i g^c(\mathbf{x}_i)]^2\\ &\leq \frac{1}{2}(Q_i^c)^2 + Q_i^c[\gamma T_i g^c(\mathbf{x}_i)] + [\gamma T_i g^c(\mathbf{x}_i)]^2.\end{aligned}$$

2) $Q_i^c + \gamma T_i g^c(\mathbf{x}_i) < -\gamma T_i g^c(\mathbf{x}_i)$*:* We have $Q_{i+1}^c = -\gamma T_i g^c(\mathbf{x}_i)$ from (8). It then follows that

$$\begin{aligned}\frac{1}{2}(Q_{i+1}^c)^2 &\leq \frac{1}{2}[\gamma T_i g^c(\mathbf{x}_i)]^2 + \frac{1}{2}[Q_i^c + \gamma T_i g^c(\mathbf{x}_i)]^2\\ &= \frac{1}{2}(Q_i^c)^2 + Q_i^c[\gamma T_i g^c(\mathbf{x}_i)] + [\gamma T_i g^c(\mathbf{x}_i)]^2.\end{aligned}$$

Combining the above two cases, we have (24). Summing both sides of (24) over $c \in \mathcal{C}$, we have (23). ■

Based on Lemmas 1-2, we provide a dynamic regret bound for PQGA in the following theorem.

Theorem 1. For $J = 0$, if $\alpha > \beta^2\gamma^2 T_{\max}^2$ and $\gamma > 0$, the dynamic regret of PQGA is upper bounded by

$$\begin{aligned}\mathrm{RE_d}(T) \leq \frac{D^2 T_{\max}}{4(\alpha - \beta^2\gamma^2 T_{\max}^2)} T + \alpha(R^2 + 2R\Pi_{\mathbf{x}^\circ})\\ + \gamma^2 G^2(T_{\max}^2 + \Pi_T).\end{aligned} \tag{25}$$

Proof: We first state the following property of a 2ϱ-strongly convex function, which is shown in Lemma 2.8 in [3]:

Lemma 3. ([3, Lemma 2.8]) Let $\mathcal{Z} \subseteq \mathbb{R}^n$ be a nonempty convex set. Let $h(\mathbf{z}) : \mathbb{R}^n \to \mathbb{R}$ be a 2ϱ-strongly-convex function over $\mathcal{Z}$ w.r.t. $\|\cdot\|$. Let $\mathbf{w} = \arg\min_{\mathbf{z}\in\mathcal{Z}}\{h(\mathbf{z})\}$. Then, for any $\mathbf{u} \in \mathcal{Z}$, we have $h(\mathbf{w}) \leq h(\mathbf{u}) - \varrho\|\mathbf{u} - \mathbf{w}\|^2$.

For $J = 0$, PQGA solves **P2** for $\mathbf{x}_{i+1}$. The objective function of **P2** is 2α-strongly convex over $\mathcal{X}_0$ w.r.t. $\|\cdot\|$ due to the regularization term $\alpha\|\mathbf{x} - \mathbf{x}_i\|^2$. Recall that $\mathbf{x}_i^\circ$ is the dynamic benchmark. Since $\mathbf{x}_{i+1}$ minimizes **P2** over $\mathcal{X}_0$ for any $i \in \mathcal{I}$, we have

$$\begin{aligned}&\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\mathbf{x}_i)]^T(\mathbf{x}_{i+1} - \mathbf{x}_i) + \alpha\|\mathbf{x}_{i+1} - \mathbf{x}_i\|^2\\ &\quad + [\mathbf{Q}_{i+1} + \gamma T_i \mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})]\\ &\overset{(a)}{\leq} \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\mathbf{x}_i)]^T(\mathbf{x}_i^\circ - \mathbf{x}_i) + \alpha\|\mathbf{x}_i^\circ - \mathbf{x}_i\|^2\\ &\quad + [\mathbf{Q}_{i+1} + \gamma T_i \mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_i^\circ)] - \alpha\|\mathbf{x}_i^\circ - \mathbf{x}_{i+1}\|^2\\ &\overset{(b)}{\leq} \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\mathbf{x}_i)]^T(\mathbf{x}_i^\circ - \mathbf{x}_i)\\ &\quad + \alpha(\|\mathbf{x}_i^\circ - \mathbf{x}_i\|^2 - \|\mathbf{x}_i^\circ - \mathbf{x}_{i+1}\|^2)\end{aligned} \tag{26}$$

$$\overset{(c)}{\leq} \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\mathbf{x}_i)]^T(\mathbf{x}_i^\circ - \mathbf{x}_i) + \alpha(\Psi_i + 2R\psi_i) \tag{27}$$

where (a) follows from Lemma 3, (b) is because of $\mathbf{Q}_{i+1} + \gamma T_i \mathbf{g}(\mathbf{x}_i) \succeq \mathbf{0}$ in (20) and $\mathbf{g}(\mathbf{x}_i^\circ) \preceq \mathbf{0}$ for $\mathbf{x}_i^\circ \in \mathcal{X}$ defined in (5), which yields $[\mathbf{Q}_{i+1} + \gamma T_i \mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_i^\circ)] \preceq \mathbf{0}$ for any $i \in \mathcal{I}$, and (c) follows from $\|\mathbf{a}+\mathbf{b}\|^2 \geq \|\mathbf{a}\|^2 + \|\mathbf{b}\|^2 - 2\|\mathbf{a}\|\|\mathbf{b}\|$ and (16) in Assumption 4, which lead to

$$\begin{aligned}&\|\mathbf{x}_i^\circ - \mathbf{x}_i\|^2 - \|\mathbf{x}_i^\circ - \mathbf{x}_{i+1}\|^2\\ &\leq \|\mathbf{x}_i^\circ - \mathbf{x}_i\|^2 - \|\mathbf{x}_{i+1}^\circ - \mathbf{x}_{i+1}\|^2 - \|\mathbf{x}_i^\circ - \mathbf{x}_{i+1}^\circ\|^2\\ &\quad + 2\|\mathbf{x}_{i+1}^\circ - \mathbf{x}_{i+1}\|\|\mathbf{x}_i^\circ - \mathbf{x}_{i+1}^\circ\|\\ &\leq \Psi_i - \|\mathbf{x}_i^\circ - \mathbf{x}_{i+1}^\circ\|^2 + 2R\psi_i\end{aligned} \tag{28}$$

where $\Psi_i \triangleq \|\mathbf{x}_i^\circ - \mathbf{x}_i\|^2 - \|\mathbf{x}_{i+1}^\circ - \mathbf{x}_{i+1}\|^2$ and $\psi_i \triangleq \|\mathbf{x}_i^\circ - \mathbf{x}_{i+1}^\circ\|$.

Add $\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i} f_{\tau_i^s}(\mathbf{x}_i)$ to both sides of (27). Then, from the convexity of $f_{\tau_i^s}(\mathbf{x})$ for any $s \in \mathcal{S}_i$, we note that $f_{\tau_i^s}(\mathbf{x}_i) + [\nabla f_{\tau_i^s}(\mathbf{x}_i)]^T(\mathbf{x}_i^\circ - \mathbf{x}_i) \leq f_{\tau_i^s}(\mathbf{x}_i^\circ)$. Following this and rearranging terms, we have

$$\begin{aligned}&\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\left(f_{\tau_i^s}(\mathbf{x}_i) - f_{\tau_i^s}(\mathbf{x}_i^\circ)\right)\\ &\leq -\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\mathbf{x}_i)]^T(\mathbf{x}_{i+1} - \mathbf{x}_i) - \alpha\|\mathbf{x}_{i+1} - \mathbf{x}_i\|^2\\ &\quad -[\mathbf{Q}_{i+1} + \gamma T_i \mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})] + \alpha(\Psi_i + 2R\psi_i).\end{aligned} \tag{29}$$

We now bound the right-hand side (RHS) of (29). Note that

$$\begin{aligned}&-[\mathbf{Q}_{i+1} + \gamma T_i \mathbf{g}(\mathbf{x}_i)]^T[\gamma \mathbf{g} T_{i+1}(\mathbf{x}_{i+1})]\\ &\overset{(a)}{\leq} -\Delta_{i+1} + \|\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})\|^2 - [\gamma T_i \mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})]\\ &\overset{(b)}{=} -\Delta_{i+1} + \frac{\gamma^2}{2}\underbrace{(\|T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})\|^2 - \|T_i\mathbf{g}(\mathbf{x}_i)\|^2)}_{\triangleq\Phi_i}\\ &\quad + \frac{\gamma^2}{2}\|T_i\mathbf{g}(\mathbf{x}_i) - T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})\|^2\\ &\overset{(c)}{\leq} -\Delta_{i+1} + \frac{\gamma^2}{2}\Phi_i + \beta^2\gamma^2 T_i^2\|\mathbf{x}_{i+1} - \mathbf{x}_i\|^2\\ &\quad + \gamma^2 G^2(T_i - T_{i+1})^2\end{aligned} \tag{30}$$

where (a) is because of (23) in Lemma 2, which leads to $-\mathbf{Q}_{i+1}^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})] \leq -\Delta_{i+1} + \|\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})\|^2$, (b) is due to $\mathbf{a}^T\mathbf{b} = \frac{1}{2}(\|\mathbf{a}\|^2 + \|\mathbf{b}\|^2 - \|\mathbf{a}-\mathbf{b}\|^2)$, and (c) follows from $\frac{1}{2}\|\mathbf{a}+\mathbf{b}\|^2 \leq \|\mathbf{a}\|^2 + \|\mathbf{b}\|^2$ and Assumption 3 that $\mathbf{g}(\mathbf{x})$ is Lipschitz continuous in (13) and bounded in (14) such that

$$\begin{aligned}&\frac{1}{2}\|T_i\mathbf{g}(\mathbf{x}_i) - T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})\|^2\\ &\leq \|T_i\mathbf{g}(\mathbf{x}_i) - T_i\mathbf{g}(\mathbf{x}_{i+1})\|^2 + \|T_i\mathbf{g}(\mathbf{x}_{i+1}) - T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})\|^2\\ &\leq \beta^2 T_i^2\|\mathbf{x}_i - \mathbf{x}_{i+1}\|^2 + G^2(T_i - T_{i+1})^2.\end{aligned}$$

Applying (30) to the third term at the RHS of (29) and rearranging the terms on both sides of (29), we have

$$\begin{aligned}&\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\left(f_{\tau_i^s}(\mathbf{x}_i)-f_{\tau_i^s}(\mathbf{x}_i^\circ)\right)\\&\overset{(a)}{\leq}\frac{D^2T_{\max}}{4(\alpha-\beta^2\gamma^2T_{\max}^2)}T_i+\alpha(\Psi_i+2R\psi_i)-\Delta_{i+1}+\frac{\gamma^2}{2}\Phi_i\\&\quad+\gamma^2G^2(T_i-T_{i+1})^2\end{aligned}\tag{31}$$

where (a) follows from Assumption 2 that $\nabla f_t(\mathbf{x})$ is bounded in (12) and therefore

$$\begin{aligned}&-\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\mathbf{x}_i)]^T(\mathbf{x}_{i+1}-\mathbf{x}_i)-(\alpha-\beta^2\gamma^2T_i^2)\|\mathbf{x}_{i+1}-\mathbf{x}_i\|^2\\&=-\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\left(\left\|\frac{\nabla f_{\tau_i^s}(\mathbf{x}_i)}{2\sqrt{\frac{\alpha-\beta^2\gamma^2T_i^2}{T_i}}}+\sqrt{\frac{\alpha-\beta^2\gamma^2T_i^2}{T_i}}(\mathbf{x}_{i+1}-\mathbf{x}_i)\right\|^2\right.\\&\quad\left.-\frac{T_i\left\|\nabla f_{\tau_i^s}(\mathbf{x}_i)\right\|^2}{4(\alpha-\beta^2\gamma^2T_i)}\right)\leq\frac{D^2T_{\max}}{4(\alpha-\beta^2\gamma^2T_{\max}^2)}T_i.\end{aligned}$$

Summing both sides of (31) over $i\in\mathcal{I}$, we have

$$\begin{aligned}&\mathrm{RE_d}(T)=\sum_{i\in\mathcal{I}}\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\left(f_{\tau_i^s}(\mathbf{x}_i)-f_{\tau_i^s}(\mathbf{x}_i^\circ)\right)\\&\overset{(a)}{\leq}\frac{D^2T_{\max}}{4(\alpha-\beta^2\gamma^2T_{\max}^2)}T+\alpha\left(\|\mathbf{x}_0^\circ-\mathbf{x}_0\|^2+2R\sum_{i\in\mathcal{I}}\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}^\circ\|\right)\\&\quad+L_1+\frac{\gamma^2}{2}\|T_I\mathbf{g}(\mathbf{x}_I)\|^2+\gamma^2G^2\sum_{i\in\mathcal{I}}(T_i-T_{i+1})^2\\&\overset{(b)}{\leq}\frac{D^2T_{\max}}{4(\alpha-\beta^2\gamma^2T_{\max}^2)}T+\alpha\left(R^2+2R\Pi_{\mathbf{x}^\circ}\right)\\&\quad+\gamma^2G^2(T_{\max}^2+\Pi_T)\end{aligned}\tag{32}$$

where (a) follows by noting that Ψ_i, Δ_{i+1}, and Φ_i are telescoping terms, and their respective summations over $i\in\mathcal{I}$ are upper bounded by $\|\mathbf{x}_0^\circ-\mathbf{x}_0\|^2$, L_1, and $\|T_I\mathbf{g}(\mathbf{x}_I)\|^2$; (b) follows from (16) in Assumption 4, $L_1=\frac{1}{2}\|\mathbf{Q}_1\|^2=\frac{1}{2}\|\gamma\mathbf{g}(\mathbf{x}_0)T_0\|^2\leq\frac{1}{2}\gamma^2G^2T_{\max}^2$, and $\|T_I\mathbf{g}(\mathbf{x}_I)\|^2\leq G^2T_{\max}^2$.■

The dynamic regret bound (25) in Theorem 1 is for PQGA under single-step gradient descent. Next, we provide another dynamic regret bound for PQGA with multi-step gradient descent for J being sufficiently large.

Theorem 2. For $J>\log_\rho^{\frac{1}{8}}$ with $\rho=\frac{\alpha-\varrho}{\alpha+\varrho}<1$, if $\alpha\geq T_{\max}L$, $\eta\geq\max\{\alpha,\beta^2\gamma^2T_{\max}^2\}$, and $\gamma>0$, then for any $\xi>0$, the dynamic regret of PQGA is upper bounded by

$$\begin{aligned}\mathrm{RE_d}(T)\leq\frac{1}{4\xi}\Pi_\nabla+\frac{L+\xi}{1-8\rho^J}\Bigg[&(5+2T_{\max})\Delta_{\mathbf{x}}+R^2+\frac{D^2}{2\alpha^2}T\\&+\frac{2\eta}{\alpha}\left(R^2+2R\Pi_{\mathbf{x}^\circ}\right)+\frac{2\gamma^2}{\alpha}G^2(T_{\max}^2+\Pi_T)\Bigg]\end{aligned}\tag{33}$$

where $\Pi_\nabla\triangleq\sum_{i\in\mathcal{I}}\|\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\nabla f_{\tau_i^s}(\mathbf{x}_i^\circ)\|^2$ is the accumulated squared norm of gradient and $\Delta_{\mathbf{x}}\triangleq\sum_{i\in\mathcal{I}}\|\mathbf{x}_i^\circ-\mathbf{x}_i^\star\|^2$ is the accumulated squared distance between two dynamic benchmarks $\{\mathbf{x}_i^\circ\}$ and $\{\mathbf{x}_i^\star\}$, in which $\mathbf{x}_i^\star\triangleq\arg\min_{\mathbf{x}\in\mathcal{X}_0}\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}f_{\tau_i^s}(\mathbf{x})$ is the dynamic benchmark under the short-term constraints.

Proof: From the property of smooth functions, we have

$$\begin{aligned}&\mathrm{RE_d}(T)=\sum_{i\in\mathcal{I}}\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\left(f_{\tau_i^s}(\mathbf{x}_i)-f_{\tau_i^s}(\mathbf{x}_i^\circ)\right)\\&\overset{(a)}{\leq}\sum_{i\in\mathcal{I}}\left(\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\mathbf{x}_i^\circ)]^T(\mathbf{x}_i-\mathbf{x}_i^\circ)+L\|\mathbf{x}_i-\mathbf{x}_i^\circ\|^2\right)\\&\overset{(b)}{\leq}\frac{1}{4\xi}\Pi_\nabla+(L+\xi)\sum_{i\in\mathcal{I}}\|\mathbf{x}_i-\mathbf{x}_i^\circ\|^2\end{aligned}\tag{34}$$

where (a) follows from (11) in Assumption 1, and (b) is because of $\mathbf{a}^T\mathbf{b}\leq\frac{1}{4\xi}\|\mathbf{a}\|^2+\xi\|\mathbf{b}\|^2$ for any $\xi>0$.

We now bound the term $\sum_{i\in\mathcal{I}}\|\mathbf{x}_i-\mathbf{x}_i^\circ\|^2$ at the RHS of (34). By the inequality $\|\mathbf{a}+\mathbf{b}\|^2\leq2(\|\mathbf{a}\|^2+\|\mathbf{b}\|^2)$ and (16) in Assumption 4, we have

$$\begin{aligned}\sum_{i\in\mathcal{I}}\|\mathbf{x}_i-\mathbf{x}_i^\circ\|^2&=\|\mathbf{x}_0-\mathbf{x}_0^\circ\|^2+\sum_{i\in\mathcal{I}}\|\mathbf{x}_{i+1}-\mathbf{x}_{i+1}^\circ\|^2\\&\leq R^2+2\sum_{i\in\mathcal{I}}\left(\|\mathbf{x}_{i+1}-\mathbf{x}_i^\circ\|^2+\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}^\circ\|^2\right).\end{aligned}\tag{35}$$

To bound $\sum_{i\in\mathcal{I}}\|\mathbf{x}_{i+1}-\mathbf{x}_i^\circ\|^2$ at the RHS of (35), recall that PQGA solves **P2′** for $\mathbf{x}_{i+1}$ for $J>0$. The objective function of **P2′** is $2(\alpha+\eta)$-strongly convex over $\mathcal{X}_0$ w.r.t. $\|\cdot\|$ due to the double regularization terms. Since $\mathbf{x}_{i+1}$ minimizes **P2′** over $\mathcal{X}_0$ for any $i\in\mathcal{I}$, similar to the derivations in (26), we can show that

$$\begin{aligned}&\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)]^T(\mathbf{x}_{i+1}-\tilde{\mathbf{x}}_i^J)+\alpha\|\mathbf{x}_{i+1}-\tilde{\mathbf{x}}_i^J\|^2\\&\quad+[\mathbf{Q}_{i+1}+\gamma T_i\mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})]+\eta\|\mathbf{x}_{t+1}-\mathbf{x}_i\|^2\\&\leq\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)]^T(\mathbf{x}_i^\circ-\tilde{\mathbf{x}}_i^J)\\&\quad+\alpha(\|\mathbf{x}_i^\circ-\tilde{\mathbf{x}}_i^J\|^2-\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}\|^2)\\&\quad+\eta(\|\mathbf{x}_i^\circ-\mathbf{x}_i\|^2-\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}\|^2).\end{aligned}\tag{36}$$

From (11), since $f_{\tau_i^s}(\mathbf{x})$ is $2L$-smooth over $\mathcal{X}_0$, we have

$$\begin{aligned}f_{\tau_i^s}(\mathbf{x}_{i+1})\leq f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)+[\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)]^T(\mathbf{x}_{i+1}-\tilde{\mathbf{x}}_i^J)\\+L\|\mathbf{x}_{i+1}-\tilde{\mathbf{x}}_i^J\|^2,\quad\forall s\in\mathcal{S}_i.\end{aligned}\tag{37}$$

Since $f_{\tau_i^s}(\mathbf{x})$ is convex over $\mathcal{X}_0$, we have

$$f_{\tau_i^s}(\mathbf{x}_i^\circ)\geq f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)+[\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)]^T(\mathbf{x}_i^\circ-\tilde{\mathbf{x}}_i^J),\ \forall s\in\mathcal{S}_i.\tag{38}$$

Applying (37) and (38) to the left-hand side (LHS) and RHS of (36), respectively, and rearranging terms on both sides, we have

$$\begin{aligned}&\alpha\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}\|^2\\&\leq\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\left(f_{\tau_i^s}(\mathbf{x}_i^\circ)-f_{\tau_i^s}(\mathbf{x}_{i+1})\right)-(\alpha-T_iL)\|\mathbf{x}_{i+1}-\tilde{\mathbf{x}}_i^J\|^2\\&\quad-[\mathbf{Q}_{i+1}+\gamma T_i\mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})]-\eta\|\mathbf{x}_{i+1}-\mathbf{x}_i\|^2\\&\quad+\eta(\|\mathbf{x}_i^\circ-\mathbf{x}_i\|^2-\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}\|^2)+\alpha\|\mathbf{x}_i^\circ-\tilde{\mathbf{x}}_i^J\|^2.\end{aligned}\tag{39}$$

We now bound the RHS of (39). For the first term on the RHS of (39), we have

$$\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\left(f_{\tau_i^s}(\mathbf{x}_i^\circ)-f_{\tau_i^s}(\mathbf{x}_{i+1})\right)\overset{(a)}{\leq}\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\left(f_{\tau_i^s}(\mathbf{x}_i^\circ)-f_{\tau_i^s}(\mathbf{x}_i^\star)\right)$$

$$\overset{(b)}{\leq} \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\mathbf{x}_i^\circ)]^T(\mathbf{x}_i^\circ-\mathbf{x}_i^\star) \overset{(c)}{\leq} \frac{D^2}{4\alpha}T_i+\alpha T_i\|\mathbf{x}_i^\circ-\mathbf{x}_i^\star\|^2 \quad (40)$$

where (a) follows from the definition of $\mathbf{x}_i^\star$ below (33) that $\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i} f_{\tau_i^s}(\mathbf{x}_i^\star) \leq \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i} f_{\tau_i^s}(\mathbf{x}_{i+1})$, (b) is because of the convexity of $f_{\tau_i^s}(\mathbf{x})$, and (c) follows from $\mathbf{a}^T\mathbf{b} \leq \frac{1}{4\alpha}\|\mathbf{a}\|^2+\alpha\|\mathbf{b}\|^2$ for any $\alpha>0$ and $\nabla f_{\tau_i^s}(\mathbf{x})$ being bounded in (12) in Assumption 2. The bounds for the third and fifth terms at the RHS of (39) are given in (30) and (28), in the proof of Theorem 1, respectively.

To bound the last term at the RHS of (39), we first note that the aggregated loss function $\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i} f_{\tau_i^s}(\mathbf{x})$ is $T_i\varrho$-strongly convex and T_iL-smooth. We provide the following property of a 2ϱ-strongly convex and $2L$-smooth function, which is shown in Lemma 1 in [16]:

Lemma 4. ([16, Lemma 1]) Let $\mathcal{Z}\subseteq\mathbb{R}^n$ be a nonempty convex set. Let $h(\mathbf{z}):\mathbb{R}^n\to\mathbb{R}$ be a 2ϱ-strongly-convex and $2L$-smooth function over $\mathcal{Z}$ w.r.t. $\|\cdot\|$. Let $\mathbf{v}=\arg\min_{\mathbf{z}\in\mathcal{Z}}\{[\nabla h(\mathbf{u})]^T(\mathbf{z}-\mathbf{u})+\upsilon\|\mathbf{z}-\mathbf{u}\|^2\}$ and $\mathbf{w}=\arg\min_{\mathbf{z}\in\mathcal{Z}}\{h(\mathbf{z})\}$. Then, for any $\upsilon\geq L$, we have $\|\mathbf{w}-\mathbf{v}\|^2\leq\frac{\upsilon-\varrho}{\upsilon+\varrho}\|\mathbf{w}-\mathbf{u}\|^2$.

Applying Lemma 4 to the update of $\tilde{\mathbf{x}}_i^j$ in (9), for any $\alpha\geq T_iL$, we have

$$\|\mathbf{x}_i^\star-\tilde{\mathbf{x}}_i^j\|^2\leq\frac{\alpha-T_i\varrho}{\alpha+T_i\varrho}\|\mathbf{x}_i^\star-\tilde{\mathbf{x}}_i^{j-1}\|^2,\quad\forall j\in\mathcal{J}.$$

Note that the constant for strong convexity ϱ is smaller than the constant for gradient Lipschitz continuity, *i.e.*, $\varrho\leq L$ [15]. Therefore, we have $\frac{\alpha-T_i\varrho}{\alpha+T_i\varrho}\geq 0$ in the above J inequalities. Combining the above J inequalities and noting that $\tilde{\mathbf{x}}_i^0=\mathbf{x}_i$, and $1\leq T_i\leq T_{\max}$. Combining the above J inequalities and choosing $\alpha\geq T_{\max}L$ such that $\frac{\alpha-T_i\varrho}{\alpha+T_i\varrho}\leq\rho$, we have

$$\|\mathbf{x}_i^\star-\tilde{\mathbf{x}}_i^j\|^2\leq\rho^J\|\mathbf{x}_i^\star-\mathbf{x}_i\|^2 \quad (41)$$

From (41) and $\|\mathbf{a}+\mathbf{b}\|^2\leq 2(\|\mathbf{a}\|^2+\|\mathbf{b}\|^2)$, we have

$$\begin{aligned}\|\mathbf{x}_i^\circ-\tilde{\mathbf{x}}_i^J\|^2&\leq 2\|\mathbf{x}_i^\star-\tilde{\mathbf{x}}_i^J\|^2+2\|\mathbf{x}_i^\circ-\mathbf{x}_i^\star\|^2\\&\leq 2\rho^J\|\mathbf{x}_i^\star-\mathbf{x}_i\|^2+2\|\mathbf{x}_i^\circ-\mathbf{x}_i^\star\|^2\\&\leq 4\rho^J\|\mathbf{x}_i^\circ-\mathbf{x}_i\|^2+(4\rho^J+2)\|\mathbf{x}_i^\circ-\mathbf{x}_i^\star\|. \quad (42)\end{aligned}$$

Applying (28), (30), (40), (42) to the respective terms at the RHS of (39), we have

$$\begin{aligned}&\alpha\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}\|^2\\&\leq-(\alpha-T_iL)\|\mathbf{x}_{i+1}-\tilde{\mathbf{x}}_i^J\|^2-(\eta-\beta^2\gamma^2T_i^2)\|\mathbf{x}_{i+1}-\mathbf{x}_i\|^2\\&\quad+4\alpha\rho^J\|\mathbf{x}_i^\circ-\mathbf{x}_i\|^2+\alpha(4\rho^J+2+T_i)\|\mathbf{x}_i^\circ-\mathbf{x}_i^\star\|^2\\&\quad+\frac{D^2}{4\alpha}T_i-\Delta_{i+1}+\frac{\gamma^2}{2}\Phi_i+\gamma^2G^2(T_i-T_{i+1})^2\\&\quad+\eta(\Psi_i-\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}^\circ\|^2+2R\psi_i). \quad (43)\end{aligned}$$

Note that we set the step-size parameters as $\alpha\geq T_{\max}L$ and $\eta\geq\beta^2\gamma^2T_{\max}^2$. Thus, the first two terms at the RHS of (43) are non-positive. Dividing both sides of (43) by α and summing it over $i\in\mathcal{I}$, we have

$$\begin{aligned}&\sum_{i\in\mathcal{I}}\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}\|^2\\&\overset{(a)}{\leq}4\rho^J\sum_{i\in\mathcal{I}}\|\mathbf{x}_i^\circ-\mathbf{x}_i\|^2+(4\rho^J+2+T_{\max})\sum_{i\in\mathcal{I}}\|\mathbf{x}_i^\circ-\mathbf{x}_i^\star\|^2\\&\quad+\frac{D^2}{4\alpha^2}T-\frac{\eta}{\alpha}\sum_{i\in\mathcal{I}}\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}^\circ\|^2+\frac{\eta}{\alpha}\left(R^2+2R\Pi_{\mathbf{x}^\circ}\right)\\&\quad+\frac{\gamma^2}{\alpha}G^2(T_{\max}^2+\Pi_T) \quad (44)\end{aligned}$$

where (a) follows from steps (a) and (b) of the derivations in (32).

Applying (44) to the third term at the RHS of (35) and rearranging terms, we have

$$\begin{aligned}&(1-8\rho^J)\sum_{i\in\mathcal{I}}\|\mathbf{x}_i-\mathbf{x}_i^\circ\|^2\\&\leq 2(4\rho^J+2+T_{\max})\sum_{i\in\mathcal{I}}\|\mathbf{x}_i^\circ-\mathbf{x}_i^\star\|^2+R^2+\frac{D^2}{2\alpha^2}T\\&\quad-2\left(\frac{\eta}{\alpha}-1\right)\sum_{i\in\mathcal{I}}\|\mathbf{x}_i^\circ-\mathbf{x}_{i+1}^\circ\|^2+\frac{2\eta}{\alpha}\left(R^2+2R\Pi_{\mathbf{x}^\circ}\right)\\&\quad+\frac{2\gamma^2}{\alpha}G^2(T_{\max}^2+\Pi_T). \quad (45)\end{aligned}$$

For $8\rho^J<1$ and $\eta\geq\alpha$, we divide both sides of (45) by $1-8\rho^J$ and apply it to the second term at the RHS of (34). Then, we have (33). ■

4.2.2 Bounding the Static Regret

Using the techniques in the proof for the dynamic regret $\mathrm{RE_d}(T)$ in Theorem 1, we provide an upper bound on the static regret $\mathrm{RE_s}(T)$ generated by PQGA in the following theorem.

Theorem 3. For $J=0$, if $\alpha>\beta^2\gamma^2T_{\max}^2$ and $\gamma>0$, the static regret generated by PQGA is upper bounded by

$$\mathrm{RE_s}(T)\leq\frac{D^2T_{\max}}{4(\alpha-\beta^2\gamma^2T_{\max}^2)}T+\alpha R^2+\gamma^2G^2(T_{\max}^2+\Pi_T). \quad (46)$$

Proof: The proof is similar to that for Theorem 1. Here, we only provide an outline. Replacing all the per-period optimizers $\{\mathbf{x}_i^\circ\}$ with the static offline benchmark $\mathbf{x}^\star$ in the proof of Theorem 1, we can show that for any $\alpha>\beta^2\gamma^2T_{\max}^2$ and $\gamma>0$, the bound in (31) still holds by redefining $\Psi_i\triangleq\|\mathbf{x}^\star-\mathbf{x}_i\|^2-\|\mathbf{x}^\star-\mathbf{x}_{i+1}\|^2$ and $\psi_i=0$. Summing both sides of (31) over $i\in\mathcal{I}$, and noting that Ψ_i is a telescoping term, we have (46). ■

4.2.3 Bounding the Constraint Violation

We now proceed to provide an upper bound on the constraint violation $\mathrm{VO}^c(T)$ for PQGA. We first relate the virtual queue vector $\mathbf{Q}_I$ to $\mathrm{VO}^c(T)$ in the following lemma.

Lemma 5. The periodic virtual queue vector yielded by PQGA satisfies the following inequality:

$$\mathrm{VO}^c(T)\leq\frac{1}{\gamma}\|\mathbf{Q}_I\|,\quad\forall c\in\mathcal{C}. \quad (47)$$

Proof: From the periodic virtual queue dynamics in (8), for any $c\in\mathcal{C}$ and $i\in\mathcal{I}$, we have

$$\gamma T_ig^c(\mathbf{x}_i)\leq Q_{i+1}^c-Q_i^c. \quad (48)$$

Summing (48) over $i\in\mathcal{I}$, we have

$$\mathrm{VO}^c(T)=\sum_{i\in\mathcal{I}}T_ig^c(\mathbf{x}_i)\leq\frac{1}{\gamma}\sum_{i\in\mathcal{I}}(Q_{i+1}^c-Q_i^c)$$

$$= \frac{1}{\gamma}(Q_I^c - Q_0^c) \overset{(a)}{=} \frac{1}{\gamma}Q_I^c \overset{(b)}{\leq} \frac{1}{\gamma}\|\mathbf{Q}_I\| \quad (49)$$

where (a) follows from $Q_0^c = 0$ by initialization, and (b) is because $\|\mathbf{a}\|_\infty \leq \|\mathbf{a}\|$. ■

Using Lemma 5, we can bound the constraint violation $\text{VO}^c(T)$ through an upper bound on the virtual queue vector $\mathbf{Q}_I$. The result is stated in the following theorem.

Theorem 4. For any $J \geq 0$ and $\alpha, \eta, \gamma > 0$, the constraint violation produced by PQGA for any $c \in \mathcal{C}$ is upper bounded by

$$\text{VO}^c(T) \leq 2GT_{\max} + \frac{(\alpha+\eta)R^2 + DRT_{\max}^2 + 2\gamma^2G^2T_{\max}}{\epsilon\gamma^2}. \quad (50)$$

Proof: For any $i \in \mathcal{I}$, since $\mathbf{x}_{i+1}$ is the solution for $\mathbf{P2}'$, we have

$$\begin{aligned}
&\frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)]^T(\mathbf{x}_{i+1} - \tilde{\mathbf{x}}_i^J) + \alpha\|\mathbf{x}_{i+1} - \tilde{\mathbf{x}}_i^J\|^2 \\
&\quad + [\mathbf{Q}_{i+1} + \gamma T_i\mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})] + \eta\|\mathbf{x}_{i+1} - \mathbf{x}_i\|^2 \\
&\leq \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)]^T(\mathbf{x}' - \tilde{\mathbf{x}}_i^J) + \alpha\|\mathbf{x}' - \tilde{\mathbf{x}}_i^J\|^2 \\
&\quad + [\mathbf{Q}_{i+1} + \gamma T_i\mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}')] + \eta\|\mathbf{x}' - \mathbf{x}_i\|^2 \quad (51)
\end{aligned}$$

where $\mathbf{x}'$ is an interior point of $\mathcal{X}$ defined below (3) that satisfies $\mathbf{g}(\mathbf{x}') \preceq \epsilon\mathbf{1}$ from Assumption 3.3). Also, note that

$$\begin{aligned}
&[\mathbf{Q}_{i+1} + \gamma T_i\mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}')] \\
&\overset{(a)}{\leq} -\epsilon\gamma T_{i+1}[\mathbf{Q}_{i+1} + \gamma T_i\mathbf{g}(\mathbf{x}_i)]^T\mathbf{1} \\
&\overset{(b)}{\leq} -\epsilon\gamma T_{i+1}\|\mathbf{Q}_{i+1} + \gamma T_i\mathbf{g}(\mathbf{x}_i)\| \\
&\overset{(c)}{\leq} -\epsilon\gamma T_{i+1}(\|\mathbf{Q}_{i+1}\| - \|\gamma T_i\mathbf{g}(\mathbf{x}_i)\|) \quad (52)
\end{aligned}$$

where (a) follows from (15) and (20), (b) is because of $\|\mathbf{a}\| \leq \|\mathbf{a}\|_1$, and (c) follows from $|(\|\mathbf{a}\| - \|\mathbf{b}\|)| \leq \|\mathbf{a}-\mathbf{b}\|$ and (21).

Applying (52) to the third term at the RHS of (51) and rearranging the terms on both sides of (51), we have

$$\begin{aligned}
&\mathbf{Q}_{i+1}^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})] \\
&\leq -\epsilon\gamma T_{i+1}(\|\mathbf{Q}_{i+1}\| - \|\gamma T_i\mathbf{g}(\mathbf{x}_i)\|) + \alpha\|\mathbf{x}' - \tilde{\mathbf{x}}_i^J\|^2 \\
&\quad + \eta\|\mathbf{x}' - \mathbf{x}_i\|^2 + \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}[\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)]^T(\mathbf{x}' - \mathbf{x}_{i+1}) \\
&\quad - [\gamma T_i\mathbf{g}(\mathbf{x}_i)]^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})] \\
&\overset{(a)}{\leq} -\epsilon\gamma T_{i+1}\|\mathbf{Q}_{i+1}\| + \epsilon\gamma^2 T_{i+1}\|T_i\mathbf{g}(\mathbf{x}_i)\| + \alpha\|\mathbf{x}' - \tilde{\mathbf{x}}_i^J\|^2 \\
&\quad + \eta\|\mathbf{x}' - \mathbf{x}_i\|^2 + \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\|\nabla f_{\tau_i^s}(\tilde{\mathbf{x}}_i^J)\|\|\mathbf{x}' - \mathbf{x}_{i+1}\| \\
&\quad + \gamma^2\|T_i\mathbf{g}(\mathbf{x}_i)\|\|T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})\| \\
&\overset{(b)}{\leq} -\epsilon\gamma T_{i+1}\|\mathbf{Q}_{i+1}\| + \epsilon\gamma^2 GT_{i+1}T_i + (\alpha+\eta)R^2 \\
&\quad + DRT_i + \gamma^2G^2T_{i+1}T_i \quad (53)
\end{aligned}$$

where (a) is because of $|\mathbf{a}^T\mathbf{b}| \leq \|\mathbf{a}\|\|\mathbf{b}\|$, and (b) follows from the bound on $\mathbf{g}(\mathbf{x})$ in (14), the bound on $\mathcal{X}_0$ in (16), and the bound on $\nabla f_t(\mathbf{x})$ in (12). From (23) in Lemma 2, we have

$$\Delta_{i+1} \leq \mathbf{Q}_{i+1}^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})] + \|\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})\|^2$$
$$\leq \mathbf{Q}_{i+1}^T[\gamma T_{i+1}\mathbf{g}(\mathbf{x}_{i+1})] + \gamma^2G^2T_{i+1}^2. \quad (54)$$

Applying (53) to the first term at the RHS of (54), we have

$$\begin{aligned}
\Delta_{i+1} &\leq -\epsilon\gamma T_{i+1}\|\mathbf{Q}_{i+1}\| + \epsilon\gamma^2GT_{i+1}T_i + (\alpha+\eta)R^2 \\
&\quad + DRT_i + \gamma^2G^2T_{i+1}T_i + \gamma^2G^2T_{i+1}^2. \quad (55)
\end{aligned}$$

Since $1 \leq T_i \leq T_{\max}$ for any $i \in \mathcal{I}$, from (55), the sufficient condition for $\Delta_{i+1} < 0$ is

$$\|\mathbf{Q}_{i+1}\| > \gamma GT_{\max} + \frac{(\alpha+\eta)R^2 + DRT_{\max}^2 + 2\gamma^2G^2T_{\max}}{\epsilon\gamma}.$$

If the above inequality holds, we have $\|\mathbf{Q}_{i+2}\| \leq \|\mathbf{Q}_{i+1}\|$, *i.e.*, the virtual queue length decreases; otherwise, by (22), the increment from $\|\mathbf{Q}_{i+1}\|$ to $\|\mathbf{Q}_{i+2}\|$ is upper bounded, since $\|\mathbf{Q}_{i+2}\| \leq \|\mathbf{Q}_{i+1}\| + \|\gamma\mathbf{g}(\mathbf{x}_{i+1})T_{i+1}\| \leq \|\mathbf{Q}_{i+1}\| + \gamma GT_{\max}$. It follows that, the virtual queue vector $\mathbf{Q}_I$ for the last updating period $I-1$ is upper bounded by

$$\|\mathbf{Q}_I\| \leq 2\gamma GT_{\max} + \frac{(\alpha+\eta)R^2 + DRT_{\max}^2 + 2\gamma^2G^2T_{\max}}{\epsilon\gamma}.$$

Applying the above inequality into (47) in Lemma 5, we have (50). ■

4.3 Discussion on the Performance Bounds

We now provide some further discussions on the regret and constraint violation bounds of PQGA obtained in Section 4.2 and the choice of algorithm parameters. To describe the level of time variation in the dynamic benchmark and update periods, we define parameters $\nu \geq 0$ and $\delta \geq 0$ such that the time variations of the system in (17) and (18) can be respectively expressed as

$$\Pi_{\mathbf{x}^\circ} = \mathcal{O}(T^\nu), \quad \Pi_T = \mathcal{O}(T^\delta). \quad (56)$$

We show below that suitable values of parameters α, η, and γ for PQGA depend on whether ν is known. Furthermore, the growth behavior of the regret and constraint violation over time also depends on the number of aggregated gradient descent steps J. We summarize the growth behavior of the regret and constraint violation yielded by PQGA under different values of J in Tables 3 and 4.

4.3.1 Regret and Constraint Violation Bounds for $J = 0$

From Theorems 1, 3, and 4, we obtain the following two corollaries regarding the regret and constraint violation bounds for $J = 0$. The results can be easily derived by substituting the chosen parameters α and γ into the general performance bounds in (25), (46), and (50), and thus we omit the derivation details to avoid repetition.

Corollary 1. (Algorithm parameters with knowledge of ν) Let $\gamma = 1$ in PQGA. Then, for $J = 0$, $\text{RE}_\text{d}(T) = \mathcal{O}(\max\{T^{\frac{1+\nu}{2}}, T^\delta\})$ if $\alpha = T^{\frac{1-\nu}{2}} + \beta^2\gamma^2T_{\max}^2$, and $\text{RE}_\text{s}(T) = \mathcal{O}(\max\{T^{\frac{1}{2}}, T^\delta\})$ if $\alpha = T^{\frac{1}{2}} + \beta^2\gamma^2T_{\max}^2$. In both cases, $\text{VO}^c(T) = \mathcal{O}(T^{\frac{1}{2}})$. Therefore, for any $0 \leq \nu < 1$ and $0 \leq \delta < 1$, the dynamic and static regrets are sublinear, and the constraint violation are sublinear.

Corollary 2. (Algorithm parameters without knowledge of ν) Let $\alpha = T^{\frac{1}{2}} + \beta^2\gamma^2T_{\max}^2$ and $\gamma = 1$ in PQGA. Then, for $J = 0$, $\text{RE}_\text{d}(T) = \mathcal{O}(\max\{T^{\frac{1}{2}+\nu}, T^\delta\})$, $\text{RE}_\text{s}(T) = \mathcal{O}(\max\{T^{\frac{1}{2}}, T^\delta\})$, and $\text{VO}^c(T) = \mathcal{O}(T^{\frac{1}{2}})$.

TABLE 3
Dynamic Regret, Static Regret, and Constraint Violation Bounds of PQGA for $J = 0$ ($\Pi_{\mathbf{x}^\circ} = \mathcal{O}(T^\nu)$ and $\Pi_T = \mathcal{O}(T^\delta)$)

$\Pi_T = \mathcal{O}(1)$	Require ν	$\mathrm{RE_d}(T)$	$\mathrm{RE_s}(T)$	$\mathrm{VO}^c(T)$
No	Yes	$\mathcal{O}(\max\{T^{\frac{1+\nu}{2}}, T^\delta\})$	$\mathcal{O}(\max\{T^{\frac{1}{2}}, T^\delta\})$	$\mathcal{O}(T^{\frac{1}{2}})$
No	No	$\mathcal{O}(\max\{T^{\frac{1}{2}+\nu}, T^\delta\})$	$\mathcal{O}(\max\{T^{\frac{1}{2}}, T^\delta\})$	$\mathcal{O}(T^{\frac{1}{2}})$
Yes	Yes	$\mathcal{O}(T^{\frac{1+\nu}{2}})$	$\mathcal{O}(T^{\frac{1}{2}})$	$\mathcal{O}(1)$
Yes	No	$\mathcal{O}(T^{\frac{1}{2}+\nu})$	$\mathcal{O}(T^{\frac{1}{2}})$	$\mathcal{O}(1)$

TABLE 4
Improved Dynamic Regret and Constraint Violation Bounds of PQGA for $J \gg 0$ ($\Pi_{\mathbf{x}^\circ} = \mathcal{O}(T^\nu)$ and $\Pi_T = \mathcal{O}(T^\delta)$)

$\Pi_T = \mathcal{O}(1)$	Require ν	$\mathrm{RE_d}(T)$	$\mathrm{VO}^c(T)$
No	No	$\mathcal{O}(\max\{T^\nu, T^\delta\})$	$\mathcal{O}(1)$
Yes	No	$\mathcal{O}(T^\nu)$	$\mathcal{O}(1)$

Corollaries 1 and 2 indicate that a sufficient condition for PQGA to yield sublinear regrets under periodic updates is that the time variation measures $\Pi_{\mathbf{x}^\circ}$ and Π_T of the system grow sublinearly over time. Note that the sublinearity of the system variation measures is necessary to have sublinear dynamic regret for OCO [10]. Otherwise, if the system varies too fast over time, no online algorithm can track it due to the lack of in-time information. This can be seen from the dynamic regret bounds derived in [7], [13]-[17], [21], [24], [25] even under the standard per-time-slot update setting. In practice, for many online applications, the system tends to stabilize over time, resulting in sublinear time variation and thus sublinear regrets under our proposed algorithm.

4.3.2 Improved Dynamic Regret Bound for $J \gg 0$

Using Theorems 2 and 4, we obtain the following corollaries regarding the dynamic regret and constraint violation bounds for PQGA, when the number of aggregated gradient descent steps J is sufficiently large.

Corollary 3. Suppose $\Pi_\nabla = \mathcal{O}(T^\nu)$ and $\Delta_{\mathbf{x}} = \mathcal{O}(T^\nu)$. Let $\alpha = T^{\frac{1}{2}} + T_{\max}L$, $\eta = \max\{\alpha, \beta^2\gamma^2 T_{\max}^2\}$, and $\gamma^2 = T^{\frac{1}{2}}$. Then, for $J > \log_\rho^{\frac{1}{8}}$, $\mathrm{RE_d}(T) = \mathcal{O}(\max\{T^\nu, T^\delta\})$ and $\mathrm{VO}^c(T) = \mathcal{O}(1)$. Therefore, for any $0 \le \nu < 1$ and $0 \le \delta < 1$, both the dynamic regret and the constraint violation are sublinear.

Compared with the growth rate of the dynamic regret $\mathcal{O}(\max\{T^{\frac{1+\nu}{2}}, T^\delta\})$ for $J = 0$ in Corollary 1 (or $\mathcal{O}(\max\{T^{\frac{1}{2}+\nu}, T^\delta\})$ in Corollary 2), Corollary 3 shows that when J is sufficiently large, the growth rate of the dynamic regret reduces to $\mathcal{O}(\max\{T^\nu, T^\delta\})$, and the growth rate of the constraint violation is reduced from $\mathcal{O}(T^{\frac{1}{2}})$ to $\mathcal{O}(1)$. Note that the choice of algorithm parameters in Corollary 1 requires the knowledge of the time variation measure of the system ν. In contrast, setting the algorithm parameters in Corollary 3 does not require such knowledge of the system variation. We further note that the accumulated squared norm of gradient Π_∇ in Corollary 3 can be very small [16]. In particular, we have $\Pi_\nabla = 0$ if $\mathbf{x}_i^\circ$ is an interior point of $\mathcal{X}_0$ (or there is no short-term constraint), *i.e.*, $\sum_{s\in\mathcal{S}_i} \nabla f_{\tau_i^s}(\mathbf{x}_i^\circ) = \mathbf{0}$ for any $i \in \mathcal{I}$. In addition, the accumulated squared distance $\Delta_{\mathbf{x}}$ between the two dynamic benchmarks $\{\mathbf{x}_i^\circ\}$ and $\{\mathbf{x}_i^\star\}$ can also be small. Specifically, if the distance between $\mathbf{x}_i^\circ$ and $\mathbf{x}_i^\star$ satisfies $\|\mathbf{x}_i^\circ - \mathbf{x}_i^\star\| \propto T^{\frac{\nu-1}{2}}, \forall i \in \mathcal{I}$, then we have $\Delta_{\mathbf{x}} = \mathcal{O}(T^\nu)$.

4.3.3 A Special Case of Bounded Π_T

We also have the following results on the regret and constraint violation bounds when the accumulated variation of update periods is upper bounded by a constant, *i.e.*, $\Pi_T = \mathcal{O}(1)$. In particular, the bounded Π_T includes the case when the update periods are fixed over time. These results are obtained by setting $\delta = 0$ in Corollaries 1-3, respectively, and we omit the proofs for brevity.

Corollary 4. (Algorithm parameters with knowledge of ν) Let $\gamma^2 = T^{\frac{1}{2}}$ in PQGA. Then, for $J = 0$, $\mathrm{RE_d}(T) = \mathcal{O}(T^{\frac{1+\nu}{2}})$ if $\alpha = T^{\frac{1-\nu}{2}} + \beta^2\gamma^2 T_{\max}^2$, and $\mathrm{RE_s}(T) = \mathcal{O}(T^{\frac{1}{2}})$ if $\alpha = T^{\frac{1}{2}} + \beta^2\gamma^2 T_{\max}^2$. In both cases, $\mathrm{VO}^c(T) = \mathcal{O}(1)$.

Corollary 5. (Algorithm parameters without knowledge of ν) Let $\gamma^2 = T^{\frac{1}{2}}$ and $\alpha = T^{\frac{1}{2}} + \beta^2\gamma^2 T_{\max}^2$ in PQGA. Then, for $J = 0$, $\mathrm{RE_d}(T) = \mathcal{O}(T^{\frac{1}{2}+\nu})$, $\mathrm{RE_s}(T) = \mathcal{O}(T^{\frac{1}{2}})$, and $\mathrm{VO}^c(T) = \mathcal{O}(1)$.

Corollary 6. Suppose $\Pi_\nabla = \mathcal{O}(T^\nu)$ and $\Delta_{\mathbf{x}} = \mathcal{O}(T^\nu)$. Let $\alpha = T^{\frac{1}{2}} + T_{\max}L$, $\eta = \max\{\alpha, \beta^2\gamma^2 T_{\max}^2\}$, and $\gamma = T^{\frac{1}{2}}$. Then, for $J > \log_\rho^{\frac{1}{8}}$, $\mathrm{RE_d}(T) = \mathcal{O}(T^\nu)$ and $\mathrm{VO}^c(T) = \mathcal{O}(1)$.

From the above results, we see that by increasing J, the growth rate of the dynamic regret yielded by PQGA is reduced from $\mathcal{O}(T^{\frac{1+\nu}{2}})$ in Corollary 4 to $\mathcal{O}(T^\nu)$ in Corollary 6, while $\mathcal{O}(1)$ growth rate of the constraint violation is maintained. To the best of our knowledge, even under the standard per-time-slot update setting, no existing algorithms for constrained OCO have simultaneously achieved $\mathcal{O}(T^\nu)$ dynamic regret and $\mathcal{O}(1)$ constraint violation.

Note that PQGA can be applied to the special case of per-time-slot updates [18]-[25], where $T_i = T_{\max} = 1$ for all i's. In this case, from Corollaries 4 and 5, we see that PQGA achieves $\mathcal{O}(T^{\frac{1}{2}})$ growth rate of the static regret and $\mathcal{O}(1)$ growth rate of the constraint violation, which are the current best results provided in [22]. We point out that [22] does not provide any dynamic regret bound. In contrast, we show that PQGA can achieve $\mathcal{O}(T^{\frac{1+\nu}{2}})$ growth rate of the

dynamic regret.[4]

5 Application to Network Virtualization in Massive MIMO Systems

In many wireless systems, CSI is only available after a sequence of channel estimation, quantization, and feedback processes. The resulting feedback delay on the CSI is especially severe for massive MIMO systems, where the channel state space is large and the channel state may fluctuate fast over time. For such massive MIMO systems, PQGA can be applied to solve a variety of problems in the presence of time-varying system states and delayed information feedback.

As an example to study the algorithm performance in practical systems, we apply PQGA to online network virtualization in massive MIMO systems, where multiple service providers (SPs) simultaneously share all the antennas and spectrum resource provided by an infrastructure provider (InP). Most of the existing works on MIMO virtualization have focused on static optimization problems [40]-[45]. Furthermore, these works have adopted strict physical isolation among the SPs. Such physical isolation approach does not take full advantage of spatial spectrum sharing enabled by MIMO precoding. In contrast, in [33] and [46], spatial isolation approach via MIMO precoding has been adopted to achieve virtualization, where the SPs share all antenna and spectrum resource simultaneously. The virtualization solutions in [33] and [46] are online strategies. However, they are based on Lyapunov optimization and require the current CSI. Furthermore, neither of them considers periodic precoder updates, which are essential to practical wireless networks, such as LTE and 5G NR.

5.1 Online Precoding-Based Massive MIMO Network Virtualization

We consider an InP performing network virtualization in a massive MIMO cellular network. In each cell, the InP owns a base station (BS) equipped with N antennas, serving M SPs. Let $\mathcal{M} = \{1, \ldots, M\}$. Each SP m has K_m users. The total number of users in the cell is $K = \sum_{m\in\mathcal{M}} K_m$. The system is time slotted with time indexed by t.

5.1.1 Precoding-Based Network Virtualization

Let $\mathbf{H}_t^m \in \mathbb{C}^{K_m \times N}$ be the local CSI between the BS and the K_m users of SP m at time t. For ease of exposition, we first consider an idealized massive MIMO virtualization framework, where CSI feedback per time slot experiences no delay, as shown in Fig. 2. At each time slot t, the InP shares the corresponding local CSI $\mathbf{H}_t^m$ with SP m and allocates transmit power P_m to the SP. The power allocation is limited by the total transmit power budget $P_{\max}$, *i.e.*, $\sum_{m\in\mathcal{M}} P_m \le P_{\max}$. Using $\mathbf{H}_t^m$, each SP m designs its own precoding matrix $\mathbf{W}_t^m \in \mathbb{C}^{N\times K_m}$ based on the service needs

4. The performance analysis in [22] is for convex loss functions, while our focus is on the strongly convex loss function case. Nonetheless, one can easily verify that the proofs of Theorems 1, 3, and 4 also hold for the convex loss function case. Therefore, PQGA still achieves $\mathcal{O}(T^{\frac{1+\nu}{2}})$ dynamic regret, $\mathcal{O}(T^{\frac{1}{2}})$ static regret, and $\mathcal{O}(1)$ constraint violation for convex loss functions.

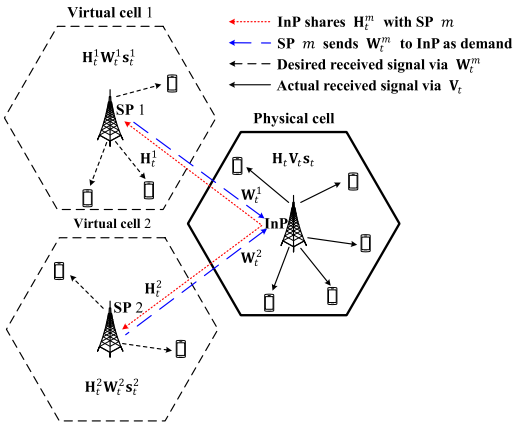


Fig. 2. An illustration of idealized massive MIMO virtualization in a cell with one InP and two SPs serving users in their respective virtual cell.

of its users, while ensuring $\|\mathbf{W}_t^m\|_F^2 \le P_m$. The SP then sends $\mathbf{W}_t^m$ to the InP as its service demand. Note that each SP m designs $\mathbf{W}_t^m$ based only on its local CSI without the knowledge of users of the other SPs. For SP m, its *desired* received signal vector $\tilde{\mathbf{y}}_t^m$ at its K_m users is given by

$$\tilde{\mathbf{y}}_t^m = \mathbf{H}_t^m \mathbf{W}_t^m \mathbf{s}_t^m, \quad \forall m \in \mathcal{M}$$

where $\mathbf{s}_t^m$ is the transmitted signal vector from SP m to its K_m users. Let $\tilde{\mathbf{y}}_t \triangleq [\tilde{\mathbf{y}}_t^{1\,H}, \ldots, \tilde{\mathbf{y}}_t^{M\,H}]^H$ be the desired received signal vector at all K users, $\mathbf{D}_t \triangleq \text{blkdiag}\{\mathbf{H}_t^1\mathbf{W}_t^1, \ldots, \mathbf{H}_t^M\mathbf{W}_t^M\}$ be the virtualization demand from all the SPs, and $\mathbf{s}_t \triangleq [\mathbf{s}_t^{1\,H}, \ldots, \mathbf{s}_t^{M\,H}]^H$. Then we have $\tilde{\mathbf{y}}_t = \mathbf{D}_t\mathbf{s}_t$. We assume that the transmitted signals to all users are independent to each other, with $\mathbb{E}\{\mathbf{s}_t\mathbf{s}_t^H\} = \mathbf{I}, \forall t$.

At each time slot t, the InP has the global CSI $\mathbf{H}_t = [\mathbf{H}_t^{1\,H}, \ldots, \mathbf{H}_t^{M\,H}]^H \in \mathbb{C}^{K\times N}$ and designs the *actual* global downlink precoding matrix $\mathbf{V}_t \triangleq [\mathbf{V}_t^1, \ldots, \mathbf{V}_t^M] \in \mathbb{C}^{N\times K}$ to serve all K users, where $\mathbf{V}_t^m \in \mathbb{C}^{N\times K_m}$ is the actual downlink precoding matrix for SP m. Then, the actual received signal vector $\mathbf{y}_t^m$ at the users of SP m is given by

$$\mathbf{y}_t^m = \mathbf{H}_t^m \mathbf{V}_t^m \mathbf{s}_t^m + \sum_{l\neq m, l\in\mathcal{M}} \mathbf{H}_t^m \mathbf{V}_t^l \mathbf{s}_t^l, \quad \forall m \in \mathcal{M}$$

where the second term is the inter-SP interference from the other SPs to the users of SP m. The actual received signal vector $\mathbf{y}_t \triangleq [\mathbf{y}_t^{1\,H}, \ldots, \mathbf{y}_t^{M\,H}]^H$ at all K users is given by $\mathbf{y}_t = \mathbf{H}_t\mathbf{V}_t\mathbf{s}_t$.

For downlink massive MIMO network virtualization, the InP designs the precoding matrix $\mathbf{V}_t$ to mitigate the inter-SP interference in order to meet the virtualization demand $\mathbf{D}_t$ received from the SPs. The expected deviation of the actual received signals from that of the SPs' virtualization demand is given by $\mathbb{E}\{\|\mathbf{y}_t - \tilde{\mathbf{y}}_t\|^2\} = \|\mathbf{H}_t\mathbf{V}_t - \mathbf{D}_t\|_F^2$. As such, we define the *precoding deviation* between any precoding matrix $\mathbf{V}$ and the virtualization demand $\mathbf{D}_t$ as:

$$f_t(\mathbf{V}) \triangleq \|\mathbf{H}_t\mathbf{V} - \mathbf{D}_t\|_F^2, \quad \forall t \in \mathcal{T}, \tag{57}$$

which we use as the design metric for massive MIMO network virtualization. Note that $f_t(\mathbf{V})$ measures the difference between the actual global precoder executed at the BS and the virtual local precoders demanded by the SPs. Furthermore, it is strongly convex in $\mathbf{V}$.

5.1.2 Online Precoding Optimization with Periodic Updates

Under a typical cellular network architecture, such as LTE and 5G NR, we consider an online periodic virtualization demand-response mechanism. An update period may correspond to the duration of one or multiple resource blocks and may vary over time. Within each update period $i \in \mathcal{I}$, the InP, which is the decision maker as defined in Section 3, receives multiple delayed CSI $\mathbf{H}_t$ and virtualization demand $\mathbf{D}_t$ feedbacks for $t \in \mathcal{T}_i$. At the beginning of each update period i, the InP determines $\mathbf{V}_i$ in the compact convex set

$$\mathcal{V}_0 \triangleq \{\mathbf{V} : \|\mathbf{V}\|_F^2 \leq P_{\max}\} \tag{58}$$

to meet the short-term transmit power constraint. We also consider a long-term transmit power constraint as in (1), where

$$g(\mathbf{V}) \triangleq \|\mathbf{V}\|_F^2 - \bar{P} \tag{59}$$

is the transmit power constraint function, and $\bar{P} \leq P_{\max}$ is the average transmit power budget. As a result, our online optimization problem for massive MIMO network virtualization has the same form as **P1**, with the loss function, short-term constraint, and long-term constraint function given in (57), (58), and (59), respectively.

5.2 Online Precoding Solution

Using the proposed PQGA algorithm in Algorithm 1, at the beginning of each update period $i+1$, we first initialize an intermediate precoder $\tilde{\mathbf{V}}_i^0 = \mathbf{V}_i$. If $J > 0$, for each $j \in \mathcal{J}$, we solve the following precoder optimization problem for $\tilde{\mathbf{V}}_i^j$:

$$\min_{\mathbf{V}\in\mathcal{V}_0} \frac{T_i}{S_i} \sum_{s\in\mathcal{S}_i} 2\Re\{\mathrm{tr}\{[\nabla_{\tilde{\mathbf{V}}_i^{j-1*}} f_{\tau_i^s}(\tilde{\mathbf{V}}_i^{j-1})]^H(\mathbf{V}-\tilde{\mathbf{V}}_i^{j-1})\}\} + \alpha\|\mathbf{V}-\tilde{\mathbf{V}}_i^{j-1}\|_F^2$$

where $\nabla_{\tilde{\mathbf{V}}_i^{j-1*}} f_{\tau_i^s}(\tilde{\mathbf{V}}_i^{j-1}) = \mathbf{H}_{\tau_i^s}^H(\mathbf{H}_{\tau_i^s}\tilde{\mathbf{V}}_i^{j-1} - \mathbf{D}_{\tau_i^s})$. The optimal solution to the above projected gradient descent problem can be directly obtained from the closed-form expression provided in (9), and is given by

$$\tilde{\mathbf{V}}_i^j = \begin{cases} \tilde{\mathbf{X}}_i^j, & \text{if } \|\tilde{\mathbf{X}}_i^j\|_F^2 \leq P_{\max} \\ \sqrt{P_{\max}}\frac{\tilde{\mathbf{X}}_i^j}{\|\tilde{\mathbf{X}}_i^j\|_F}, & \text{o.w.} \end{cases} \tag{60}$$

where $\tilde{\mathbf{X}}_i^j = \tilde{\mathbf{V}}_i^{j-1} - \frac{T_i}{\alpha S_i}\sum_{s\in\mathcal{S}_i}\mathbf{H}_{\tau_i^s}^H(\mathbf{H}_{\tau_i^s}\tilde{\mathbf{V}}_i^{j-1} - \mathbf{D}_{\tau_i^s})$.

After performing J-step aggregated gradient descent, we obtain both $\tilde{\mathbf{V}}^J$ and $\mathbf{V}_i$. Then, we solve the following precoder optimization problem for the precoding matrix $\mathbf{V}_{i+1}$ at the InP:

$$\mathbf{P3}': \min_{\mathbf{V}\in\mathcal{V}_0} \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i} 2\Re\{\mathrm{tr}\{[\nabla_{\tilde{\mathbf{V}}_i^{J*}} f_{\tau_i^s}(\tilde{\mathbf{V}}_i^J)]^H(\mathbf{V}-\tilde{\mathbf{V}}_i^J)\}\}$$
$$+\alpha\|\mathbf{V}-\tilde{\mathbf{V}}_i^J\|_F^2 + \eta\|\mathbf{V}-\mathbf{V}_i\|_F^2 + [Q_{i+1}+\gamma T_i g(\mathbf{V}_i)][\gamma T_{i+1} g(\mathbf{V})]$$

where Q_{i+1} is a periodic virtual queue with updating rule given in (8). Since **P3′** is a convex optimization problem, we can solve it via Karush-Kuhn-Tucker (KKT) conditions [47]. The Lagrangian for **P3′** is

$$L(\mathbf{V},\lambda) = \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i} 2\Re\{\mathrm{tr}\{[\nabla_{\tilde{\mathbf{V}}_i^{J*}} f_{\tau_i^s}(\tilde{\mathbf{V}}_i^J)]^H(\mathbf{V}-\tilde{\mathbf{V}}_i^J)\}\}$$
$$+\alpha\|\mathbf{V}-\tilde{\mathbf{V}}_i^J\|_F^2 + [Q_{i+1}+\gamma T_i g(\mathbf{V}_i)][\gamma T_{i+1} g(\mathbf{V})]$$
$$+\eta\|\mathbf{V}-\mathbf{V}_i\|_F^2 + \lambda(\|\mathbf{V}\|_F^2 - P_{\max})$$

where λ is the Lagrange multiplier associated with the short-term transmit power constraint in (58). The KKT conditions for $(\mathbf{V}^\star, \lambda^\star)$ are given by $\|\mathbf{V}^\star\|_F^2 - P_{\max} \leq 0$, $\lambda^\star \geq 0$, $\lambda^\star(\|\mathbf{V}^\star\|_F^2 - P_{\max}) = 0$, and

$$\mathbf{V}^\star = \frac{\alpha\tilde{\mathbf{V}}_i^J + \eta\mathbf{V}_i - \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\mathbf{H}_{\tau_i^s}^H(\mathbf{H}_{\tau_i^s}\tilde{\mathbf{V}}_i^J - \mathbf{D}_{\tau_i^s})}{\alpha+\eta+[Q_{i+1}+\gamma T_i g(\mathbf{V}_i)]\gamma T_{i+1}+\lambda^\star}, \tag{61}$$

where (61) is obtained by setting $\nabla_{\mathbf{V}^*} L(\mathbf{V},\lambda) = \mathbf{0}$. From these KKT conditions and by noting that $\lambda^\star$ serves as a power regularization factor for $\mathbf{V}^\star$ in (61), we have a closed-form solution for $\mathbf{V}_{i+1}$, given by

$$\mathbf{V}_{i+1} = \begin{cases} \mathbf{X}_i, & \text{if } \|\mathbf{X}_i\|_F^2 \leq P_{\max} \\ \sqrt{P_{\max}}\frac{\mathbf{X}_i}{\|\mathbf{X}_i\|_F}, & \text{o.w.} \end{cases} \tag{62}$$

where $\mathbf{X}_i = \frac{\alpha\tilde{\mathbf{V}}_i^J + \eta\mathbf{V}_i - \frac{T_i}{S_i}\sum_{s\in\mathcal{S}_i}\mathbf{H}_{\tau_i^s}^H(\mathbf{H}_{\tau_i^s}\tilde{\mathbf{V}}_i^J - \mathbf{D}_{\tau_i^s})}{\alpha+\eta+[Q_{i+1}+\gamma T_i g(\mathbf{V}_i)]\gamma T_{i+1}}$.

Algorithm Complexity Analysis

For implementing PQGA in this massive MIMO virtualization problem, the online precoding solutions are obtained in closed-form as in (60) and (62). The computational complexity is mainly from matrix multiplication, which is in the order of $\mathcal{O}(NK^2)$. Note that it is similar to the complexity of the zero forcing (ZF) precoding scheme commonly employed for multi-antenna transmission in practical systems. Since only closed-form computation is involved, the overall computational complexity in PQGA in Algorithm 1 is very low for the InP.

We also note that additional short-term per-antenna transmit power constraints can be incorporated in the convex set $\mathcal{V}_0$. In this case, both precoder optimization problems can be equivalently decomposed into N subproblems, each with a closed-form solution similar to (60) and (62).

5.3 Performance Bounds

We assume that the channel gain is bounded by a constant $B > 0$ at any time t, given by

$$\|\mathbf{H}_t\|_F \leq B, \quad \forall t \in \mathcal{T}. \tag{63}$$

In the following lemma, we show that our online massive MIMO network virtualization problem satisfies Assumptions 1-4 for OCO in Section 4.2. The proof directly follows from the bounded channel gain in (63) and the short-term transmit power limits $P_{\max}$ and P_m on $\mathbf{V}_t$ and $\mathbf{W}_t^m$, and thus is omitted for brevity.

Lemma 6. Assume the bounded channel gain in (63). Then, Assumptions 1-4 hold with the corresponding constants given by $\varrho = 2$, $L = B^2$, $D = 4B^2\sqrt{P_{\max}}$, $\beta = 2\sqrt{P_{\max}}$, $G = \sqrt{\max\{\bar{P}^2, (P_{\max}-\bar{P})^2\}}$, $\epsilon = \bar{P}$, and $R = 2\sqrt{P_{\max}}$.

Following the results in Theorems 1-4, the performance bounds yielded by $\{\mathbf{V}_i\}$ are given by (25), (33), (46), and (50), with the corresponding values of $\varrho, L, D, \beta, G, \epsilon, R$ given in Lemma 6.

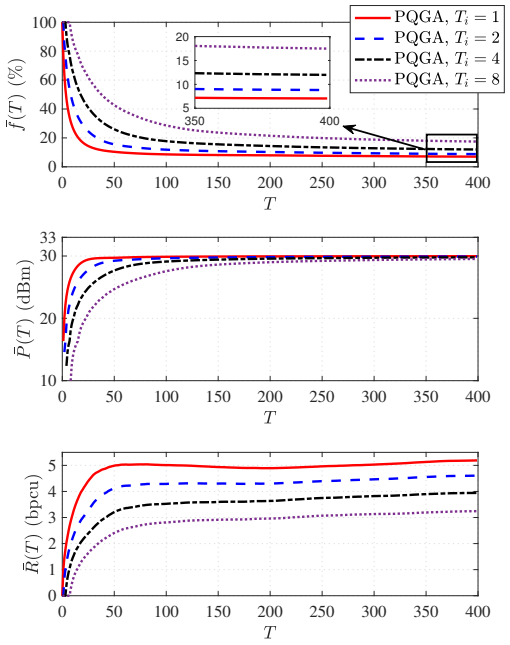


Fig. 3. $\bar{f}(T)$, $\bar{P}(T)$, and $\bar{R}(T)$ vs. T with different T_i values.

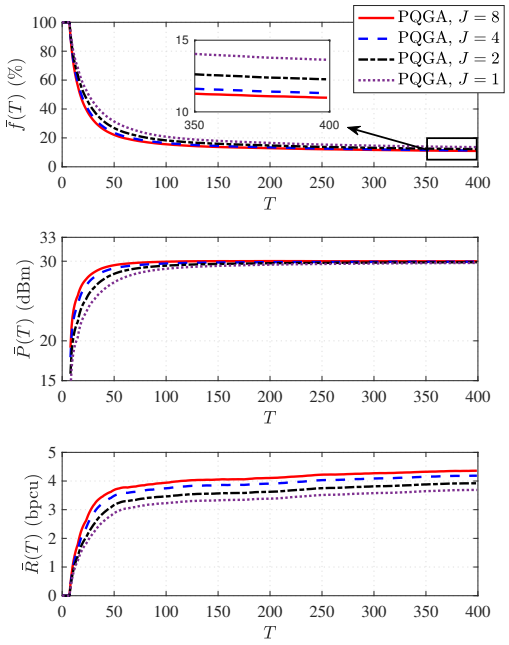


Fig. 4. $\bar{f}$(T), $\bar{P}(T)$, and $\bar{R}(T)$ vs. T with different J values.

6 Simulation Results

In this section, we study the performance of PQGA applied to online precoding-based massive MIMO network virtualization under typical urban micro-cell LTE network settings.

6.1 Simulation Setup

We consider an urban hexagon micro-cell of radius 500 m. An InP owns the BS, equipped with $N = 32$ antennas by default. The InP performs network virtualization and serves $M = 4$ SPs. We focus on the radio channel over one subcarrier with bandwidth $B_W = 15$ kHz. Over this channel, each SP $m \in \mathcal{M}$ serves $K_m = 2$ users, who are uniformly distributed in the cell, with a total of $K = 8$ users in the cell. As the default system parameters, we set the maximum transmit power limit $P_{\max} = 33$ dBm, the time-averaged transmit power limit $\bar{P} = 30$ dBm, noise power spectral density $N_0 = -174$ dBm/Hz, and noise figure $N_F = 10$ dB.

We model the fading channel as a first-order Gaussian-Markov process $\mathbf{h}_{t+1}^k = \alpha_{\mathbf{h}} \mathbf{h}_t^k + \mathbf{z}_t^k, \forall k \in \mathcal{K} = \{1, \ldots, K\}$, where $\mathbf{h}_t^k \sim \mathcal{CN}(\mathbf{0}, \beta_k \mathbf{I})$, with $\beta_k[\text{dB}] = -31.54 - 33 \log_{10}(d_k) - \psi_k$ capturing the path-loss and shadowing, with d_k being the distance from the BS to user k, and $\psi_k \sim \mathcal{CN}(0, \sigma_\phi^2)$ modeling the shadowing with $\sigma_\phi = 8$ dB; also, $\alpha_{\mathbf{h}} \in [0, 1]$ is the channel correlation coefficient, and $\mathbf{z}_t^k \sim \mathcal{CN}(\mathbf{0}, (1 - \alpha_{\mathbf{h}}^2)\beta_k \mathbf{I})$ is the innovation sequence independent of $\mathbf{h}_t^k$. We set $\alpha_{\mathbf{h}} = 0.997$ by default, which under the standard LTE transmission structure, corresponding to the pedestrian speed 1 m/s [48]. [5] We set the time slot duration $\Delta t = \frac{1}{B_W} = 66.6\ \mu s$, such that an update period of 8 time slots is similar to one resource block time duration in LTE. We set the total time horizon $T = 400$. We simulate the proposed PQGA algorithm using MATLAB on a MacBook Pro laptop, Apple M1 Pro CPU, with 16 GB memory.

5. We emphasize here that the Gauss-Markov channel model is used for illustration only. PQGA can be applied to any arbitrary wireless environment, and the InP does not need to know the channel statistics.

We assume that each SP $m \in \mathcal{M}$ uses ZF precoding scheme $\varpi_t^m \mathbf{H}_t^{mH} (\mathbf{H}_t^m \mathbf{H}_t^{mH})^{-1}$ to design its virtual precoding matrix $\mathbf{W}_t^m$, where ϖ_t^m is a power normalizing factor such that $\|\mathbf{W}_t^m\|_F^2 = P_m = \frac{P_{\max}}{M}$. For the performance evaluation, we define the time-averaged precoding deviation normalized against the virtualization demand as $\bar{f}(T) \triangleq \frac{1}{T} \sum_{i \in \mathcal{I}} \sum_{t \in \mathcal{T}_i} \frac{f_t(\mathbf{V}_i)}{\|\mathbf{D}_t\|_F^2}$, the time-averaged transmit power as $\bar{P}(T) \triangleq \frac{1}{T} \sum_{i \in \mathcal{I}} T_i \|\mathbf{V}_i\|_F^2$, and the time-averaged per-user rate as $\bar{R}(T) \triangleq \frac{1}{TK} \sum_{i \in \mathcal{I}} \sum_{t \in \mathcal{T}_i} \sum_{k \in \mathcal{K}} \log_2 \left(1 + \text{SINR}_t^{ik}\right)$, where $\text{SINR}_t^{ik} = \frac{|\mathbf{h}_t^{kT} \mathbf{v}_i^k|^2}{\sum_{j \neq k, j \in \mathcal{K}} |\mathbf{h}_t^{kT} \mathbf{v}_i^j|^2 + \sigma_n^2}$, with $\mathbf{h}_t^k$ and $\mathbf{v}_i^k$ being the channel vector at time slot t and precoding vector in the i-th update period for user k, respectively, and $\sigma_n^2 = N_0 B_W + N_F$ being the noise power.

6.2 Impact of Update Periods

We first fix the update periods $\{T_i\}$ over time, and consider only one CSI feedback is received at the beginning of each update period i. Fig. 3 shows $\bar{f}(T)$, $\bar{P}(T)$, and $\bar{R}(T)$ versus T for different values of the update period T_i. We observe that PQGA converges fast, usually within 50-150 time slots for T_i ranging from 1 to 8. As expected, the convergence rate becomes slower as T_i increases. Also, as T_i increases, the steady-state value of the time-averaged precoding deviation $\bar{f}(T)$ yielded by PQGA increases from around 7% to 17%, and the steady-state value of the time-averaged per-user rate $\bar{R}(T)$ decreases from around 5 bpcu to 3 bpcu. This demonstrates how the system performance is affected by the channel variation over time, as the precoder updates become less frequent. In all values of T_i, the time-averaged transmit power $\bar{P}(T)$ quickly converges to the average transmit power limit $\bar{P}$.

6.3 Impact of Number of Aggregated Gradient Descent Steps

For fixed update period $T_i = 8$ and one CSI feedback, Fig. 4 shows $\bar{f}(T)$, $\bar{P}(T)$ and $\bar{R}(T)$ versus T for different

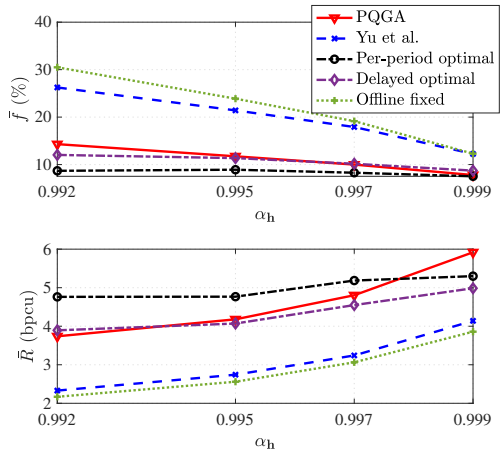


Fig. 5. Performance comparison on $\bar{f}$ and $\bar{R}$ vs. $\alpha_{\mathbf{h}}$.

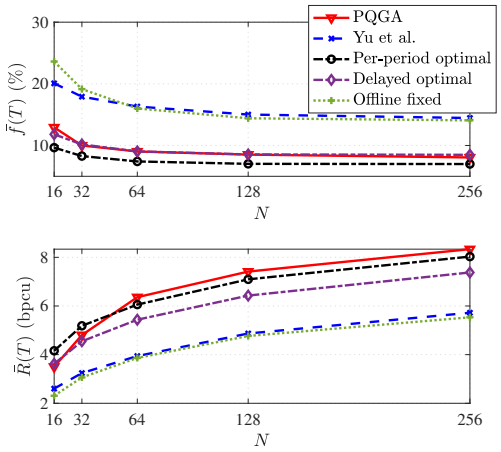


Fig. 6. Performance comparison on $\bar{f}$ and $\bar{R}$ vs. N.

numbers of the aggregated gradient descent steps J. We observe that as J increases, the steady-state value of the time-averaged precoding deviation $\bar{f}(T)$ decreases, and the steady-state value of the time-averaged per-user rate $\bar{R}(T)$ increases. These demonstrate the performance gain brought by performing multi-step gradient descent. We see that the impact of J on the time-averaged transmit power $\bar{P}(T)$ is small. We also observe that the steady-state values of $\bar{f}(T)$ and $\bar{R}(T)$ do not change much when J is close to 8. As such, in the simulation results presented below, we set $J = 8$ as the default parameter for PQGA.

6.4 Performance Comparison

For performance comparison, we consider the following method and performance benchmarks.

- *Yu et al.:* We use the online algorithm from [22] for the InP to compute the precoding matrix $\mathbf{V}_i$ at each update period i. Note that [22] achieves the current best $\mathcal{O}(T^{\frac{1}{2}})$ static regret and $\mathcal{O}(1)$ constraint violation bounds under standard OCO setting with per-time-slot updates. It has also been demonstrated in [22] that this algorithm outperforms the ones in [18] and [19]. In order to apply the algorithm in [22] to the periodic update scenario in our problem setting, we treat each update period of T_i time slots as one super time slot. Besides this, [22] considers only one gradient feedback at each time slot. Therefore, to apply the algorithm to accommodate multiple gradient feedbacks, we treat the averaged gradient as a single gradient feedback. For a fair comparison, we optimize the algorithm parameters of both PQGA and Yu et al. to achieve their respective best performance.
- *Per-period optimal:* At the beginning of each update period i, the InP receives the CSI feedback from the current update period i and uses $\mathbf{V}_i^{\circ}$ in (5).
- *Delayed optimal:* At the beginning of each update period i, the InP collects the delayed CSI feedback from the previous update period $i-1$ and uses$\mathbf{V}_{i-1}^{\circ}$ in (5).
- *Offline fixed:* The InP has the complete CSI over I update periods beforehand, and uses $\mathbf{V}^{\star}$ in (3) at each update period i.

We assume the update periods $\{T_i\}$ keep switching between 8 and 4 time slots. When $T_i = 8$, CSI feedback occurs at the first and fifth time slot, *i.e.*, $S_i = 2$. Otherwise, CSI feedback only occurs at the first time slot, *i.e.*, $T_i = 4$ and $S_i = 1$. Therefore, both the update periods $\{T_i\}$ and the numbers of CSI feedback instances in each update period $\{S_i\}$ are time varying.

In Fig. 5, we compare the steady-state precoding deviation $\bar{f}$ and per-user rate $\bar{R}$ of PQGA with those of other methods for different values of the channel correlation coefficient $\alpha_{\mathbf{h}}$. We see that there is a large performance gap between the per-period optimal method in (5) and the offline fixed method in (3). This indicates that the commonly used static benchmark for OCO may not be a meaningful comparison target for dynamic systems. As $\alpha_{\mathbf{h}}$ increases, the normalized time-averaged precoding deviation $\bar{f}$ yielded by PQGA decreases. This is due to slower channel variation over time and the accumulated system variation decreases. When $\alpha_{\mathbf{h}} > 0.993$, which corresponds to the pedestrian speed 2.5 m/s, $\bar{f}$ yielded by PQGA becomes smaller than that by the delayed optimal method. Note that the per-period optimal method uses the current CSI and has a semi-closed-form solution. In contrast, PQGA only uses the delayed CSI and its solution is in closed-form. We also observe that, as $\alpha_{\mathbf{h}}$ increases, $\bar{f}$ yielded by PQGA approaches that of the per-period optimal method. Furthermore, $\bar{f}$ yielded by PQGA is more robust to channel variation than that of Yu et al. under the periodic update setting. Here, we note that although the time-averaged per-user rate $\bar{R}$ is not the optimization objective of PQGA, $\bar{R}$ yielded by PQGA can still be higher than that of the per-period optimal method when $\alpha_{\mathbf{h}}$ is large.

With the same setting as Fig. 5, Fig. 6 shows the impact of the number of antennas N on the performance of PQGA and the other methods. As N increases, the InP has more degrees of freedom for downlink beamforming to mitigate the inter-SP interference, and thus the precoding deviation from the virtualization demand $\bar{f}$ decreases. As N increases, $\bar{f}$ yielded by PQGA becomes smaller than that of the delayed optimal method; It approaches the per-period optimal method. Furthermore, the per-user rate $\bar{R}$ of PQGA is higher than that of the per-period optimal method when N is large. Finally, we see that PQGA substantially outperforms Yu et al. for both $\bar{f}$ and $\bar{R}$ in a wide range of N values.

7 Conclusions

This paper considers a new constrained OCO problem with periodic updates, where the gradient feedbacks may be partly missing over some time slots and the online decisions are updated once in each update period, which may last for multiple time slots. We present an efficient algorithm termed PQGA, which uses periodic queues together with gradient aggregation to handle the possibly time-varying feedback delay within update periods. We provide bounds on the dynamic regret, static regret, and constraint violation of PQGA. Our analysis takes into account the impact of the new periodic constraint penalty structure and possibly multi-step aggregated gradient descent on the performance guarantees of PQGA. As an application, we apply PQGA to the online network virtualization problem in massive MIMO systems. In addition to the benefits in terms of regret and constraint violation bounds, our simulation results further demonstrate the effectiveness of PQGA in terms of the time-averaged performance.

References

[1] J. Wang, B. Liang, M. Dong, and G. Boudreau, "Online MIMO wireless network virtualization over time-varying channels with periodic updates," in *Proc. IEEE Intel. Workshop on Signal Process. Advances in Wireless Commun. (SPAWC)*, 2020.

[2] N. Cesa-Bianchi and G. Lugosi, *Prediction, Learning, and Games*. Cambridge University Press, 2006.

[3] S. Shalev-Shwartz, "Online learning and online convex optimization," *Found. Trends Mach. Learn.*, vol. 4, pp. 107–194, Feb. 2012.

[4] P. Mertikopoulos and A. L. Moustakas, "Learning in an uncertain world: MIMO covariance matrix optimization with imperfect feedback," *IEEE Trans. Signal Process.*, vol. 64, pp. 5–18, Jan. 2016.

[5] M. Lin, A. Wierman, L. L. H. Andrew, and E. Thereska, "Dynamic right-sizing for power-proportional data centers," *IEEE/ACM Trans. Netw.*, vol. 21, pp. 1378–1391, Oct. 2013.

[6] Y. Zhang, M. H. Hajiesmaili, S. Cai, M. Chen, and Q. Zhu, "Peak-aware online economic dispatching for microgrids," *IEEE Trans. Smart Grid*, vol. 9, pp. 323–335, Jan. 2018.

[7] M. Zinkevich, "Online convex programming and generalized infinitesimal gradient ascent," in *Proc. Intel. Conf. Mach. Learn. (ICML)*, 2003.

[8] 3GPP TS38.300, "3rd Generation Partnership Project Technical Specification Group Radio Access Network; NR; NR and NG-RAN Overall Description; Statge 2 (Release 15)."

[9] B. Liang, "Mobile edge computing," in *Key Technologies for 5G Wireless Systems*. V. W. S. Wong, R. Schober, D. W. K. Ng, and L.-C. Wang, Eds., Cambridge University Press, 2017.

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

[11] J. Langford, A. J. Smola, and M. Zinkevich, "Slow learners are fast," in *Proc. Adv. Neural Info. Proc. Sys. (NIPS)*, 2009.

[12] K. Quanrud and D. Khashabi, "Online learning with adversarial delays," in *Proc. Adv. Neural Info. Proc. Sys. (NIPS)*, 2015.

[13] E. C. Hall and R. M. Willett, "Online convex optimization in dynamic environments," *IEEE J. Sel. Topics Signal Process.*, vol. 9, pp. 647–662, Jun. 2015.

[14] A. Jadbabaie, A. Rakhlin, S. Shahrampour, and K. Sridharan, "Online optimization : Competing with dynamic comparators," in *Proc. Intel. Conf. Artif. Intell. Statist. (AISTATS)*, 2015.

[15] A. Mokhtari, S. Shahrampour, A. Jababaie, and A. Ribeiro, "Online optimization in dynamic environments: Improved regret rates for strongly convex problems," in *Proc. IEEE Conf. Decision Control (CDC)*, 2016.

[16] L. Zhang, T. Yang, J. Yi, R. Jin, and Z. Zhou, "Improved dynamic regret for non-degenerate functions," in *Proc. Adv. Neural Info. Proc. Sys. (NIPS)*, 2017.

[17] R. Dixit, A. S. Bedi, R. Tripathi, and K. Rajawat, "Online learning with inexact proximal online gradient descent algorithms," *IEEE Trans. Signal Process.*, vol. 67, pp. 1338–1352, 2019.

[18] 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.

[19] R. Jenatton, J. Huang, and C. Archambeau, "Adaptive algorithms for online convex optimization with long-term constraints," in *Proc. Intel. Conf. Mach. Learn. (ICML)*, 2016.

[20] X. Cao, J. Zhang, and H. V. Poor, "Impact of delays on constrained online convex optimization," in *Proc. Asilomar Conf. Signal Sys. Comput.*, 2019.

[21] T. Chen, Q. Ling, and G. B. Giannakis, "An online convex optimization approach to proactive network resource allocation," *IEEE Trans. Signal. Process.*, vol. 65, pp. 6350–6364, Dec. 2017.

[22] H. Yu and M. J. Neely, "A low complexity algorithm with $O(\sqrt{T})$ regret and $O(1)$ constraint violations for online convex optimization with long term constraints," *J. Mach. Learn. Res.*, vol. 21, pp. 1–24, Feb. 2020.

[23] H. Yu, M. J. Neely, and X. Wei, "Online convex optimization with stochastic constraints," in *Proc. Adv. Neural Info. Proc. Sys. (NIPS)*, 2017.

[24] X. Cao, J. Zhang, and H. V. Poor, "A virtual-queue-based algorithm for constrained online convex optimization with applications to data center resource allocation," *IEEE J. Sel. Topics Signal Process.*, vol. 12, pp. 703–716, Aug. 2018.

[25] J. Wang, B. Liang, M. Dong, G. Boudreau, and H. Abou-Zeid, "Delay-tolerant constrained OCO with application to network resource allocation," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2021.

[26] R. Urgaonkar, U. C. Kozat, K. Igarashi, and M. J. Neely, "Dynamic resource allocation and power management in virtualized data centers," in *IEEE/IFIP Netw. Oper. Manage. Symp. (NOMS)*, 2010.

[27] T. Ouyang, Z. Zhou, and X. Chen, "Follow me at the edge: Mobility-aware dynamic service placement for mobile edge computing," *IEEE J. Sel. Areas Commun.*, vol. 36, pp. 2333–2345, Oct. 2018.

[28] Y. Guo, M. Pan, and Y. Fang, "Optimal power management of residential customers in the smart grid," *IEEE Trans. Parallel Distrib. Syst.*, vol. 23, pp. 1593–1606, Sep. 2012.

[29] M. J. Neely, *Stochastic Network Optimization with Application on Communication and Queueing Systems*. Morgan & Claypool, 2010.

[30] M. J. Neely, "Dynamic optimization and learning for renewal systems," *IEEE Trans. Automat. Contr.*, vol. 58, pp. 32–46, 2013.

[31] M. Lotfinezhad, B. Liang, and E. S. Sousa, "Optimal control of constrained cognitive radio networks with dynamic population size," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2010.

[32] H. Yu and M. J. Neely, "Dynamic transmit covariance design in MIMO fading systems with unknown channel distributions and inaccurate channel state information," *IEEE Trans. Wireless Commun.*, vol. 16, pp. 3996–4008, Jun. 2017.

[33] J. Wang, M. Dong, B. Liang, and G. Boudreau, "Online precoding design for downlink MIMO wireless network virtualization with imperfect CSI," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2020.

[34] S. Bubeck and C.-B. Nicol, "Regret analysis of stochastic and nonstochastic multi-armed bandit problems," *Found. Trends Mach. Learn.*, vol. 5, pp. 1–122, 2012.

[35] V. Krishnamurthy, *Partially observed Markov decision processes*. Cambridge University Press, 2016.

[36] W. Xia, T. Q. S. Quek, K. Guo, W. Wen, H. H. Yang, and H. Zhu, "Multi-armed bandit-based client scheduling for federated learning," *IEEE Trans. Wireless Commun.*, vol. 19, pp. 7108–7123, Nov. 2020.

[37] X. Chen, Y. Nie, and N. Li, "Online residential demand response via contextual multi-armed bandits," *IEEE Control Syst. Lett.*, vol. 5, no. 2, pp. 433–438, 2021.

[38] J. Pajarinen, A. Hottinen, and J. Peltonen, "Optimizing spatial and temporal reuse in wireless networks by decentralized partially observable markov decision processes," *IEEE Trans. Mobile Comput.*, vol. 13, pp. 866–879, Apr. 2014.

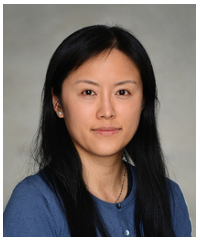
[39] O. Besbes, Y. Gur, and A. Zeevi, "Non-stationary stochastic optimization," *Oper. Res.*, vol. 63, pp. 1227–1244, Sep. 2015.

[40] V. Jumba, S. Parsaeefard, M. Derakhshani, and T. Le-Ngoc, "Resource provisioning in wireless virtualized networks via massive-MIMO," *IEEE Wireless Commun. Lett.*, vol. 4, pp. 237–240, Jun. 2015.

[41] Z. Chang, Z. Han, and T. Ristaniemi, "Energy efficient optimization for wireless virtualized small cell networks with large-scale multiple antenna," *IEEE Trans. Commun.*, vol. 65, pp. 1696–1707, Apr. 2017.

[42] S. Parsaeefard, R. Dawadi, M. Derakhshani, T. Le-Ngoc, and M. Baghani, "Dynamic resource allocation for virtualized wireless networks in massive-MIMO-aided and fronthaul-limited C-RAN," *IEEE Trans. Veh. Technol.*, vol. 66, pp. 9512–9520, Oct. 2017.
[43] D. Tweed and T. Le-Ngoc, "Dynamic resource allocation for uplink MIMO NOMA VWN with imperfect SIC," in *Proc. IEEE Int. Conf. Commun. (ICC)*, 2018.
[44] Y. Liu, M. Derakhshani, S. Parsaeefard, S. Lambotharan, and K. Wong, "Antenna allocation and pricing in virtualized massive MIMO networks via Stackelberg game," *IEEE Trans. Commun.*, vol. 66, pp. 5220–5234, Nov. 2018.
[45] K. Zhu and E. Hossain, "Virtualization of 5G cellular networks as a hierarchical combinatorial auction," *IEEE Trans. Mobile Comput.*, vol. 15, pp. 2640–2654, Oct. 2016.
[46] J. Wang, M. Dong, B. Liang, and G. Boudreau, "Online downlink MIMO wireless network virtualization in fading environments," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, 2019.
[47] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.
[48] H. Holma and A. Toskala, *WCDMA for UMTS - HSPA evolution and LTE*. John Wiely & Sons, 2010.

Juncheng Wang (Student Member, IEEE) received the B.Eng. degree in electrical engineering from Shanghai Jiao Tong University, Shanghai, China, in 2014, and the M.Sc. degree in electrical and computer engineering from the University of Alberta, Edmonton, AB, Canada, in 2017. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada. His research interests include online learning, stochastic optimization, resource allocation, and networks virtualization.



Min Dong (Senior Member, IEEE) received the B.Eng. degree from Tsinghua University, Beijing, China, in 1998, and the Ph.D. degree in electrical and computer engineering with a minor in applied mathematics from Cornell University, Ithaca, NY, in 2004. From 2004 to 2008, she was with Qualcomm Research, Qualcomm Inc., San Diego, CA. Since 2008, she has been with Ontario Tech University, where she is currently a Professor in the Department of Electrical, Computer and Software Engineering. She also holds a status-only Professor appointment with the Department of Electrical and Computer Engineering at the University of Toronto. Her research interests include wireless communications, statistical signal processing, learning techniques, optimization and control applications in cyber-physical systems.

Dr. Dong received the Early Researcher Award from the Ontario Ministry of Research and Innovation in 2012, the Best Paper Award at IEEE ICCC in 2012, and the 2004 IEEE Signal Processing Society Best Paper Award. She is a co-author of the Best Student Paper at IEEE SPAWC 2021 and the Best Student Paper of Signal Processing for Communications and Networking at IEEE ICASSP 2016. She was an elected member of the Signal Processing for Communications and Networking (SP-COM) Technical Committee of IEEE Signal Processing Society (2013-2018). She served on the Steering Committee of the IEEE TRANSACTIONS ON MOBILE COMPUTING (2019-2021). She is an Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS. She served as an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING (2010-2014) and the IEEE SIGNAL PROCESSING LETTERS (2009-2013). She was the lead Co-Chair of the Communications and Networks to Enable the Smart Grid Symposium at the IEEE International Conference on Smart Grid Communications in 2014.

Ben Liang (Fellow, IEEE) received honors-simultaneous B.Sc. (valedictorian) and M.Sc. degrees in Electrical Engineering from Polytechnic University (now the engineering school of New York University) in 1997 and the Ph.D. degree in Electrical Engineering with a minor in Computer Science from Cornell University in 2001. He was a visiting lecturer and postdoctoral research associate at Cornell University in the 2001 - 2002 academic year. He joined the Department of Electrical and Computer Engineering at the University of Toronto in 2002, where he is now Professor and L. Lau Chair in Electrical and Computer Engineering. His current research interests are in networked systems and mobile communications. He is an associate editor for the IEEE Transactions on Mobile Computing and has served on the editorial boards of the IEEE Transactions on Communications, the IEEE Transactions on Wireless Communications, and the Wiley Security and Communication Networks. He regularly serves on the organizational and technical committees of a number of conferences. He is a Fellow of IEEE and a member of ACM and Tau Beta Pi.

Gary Boudreau (Senior Member, IEEE) received a B.A.Sc. in Electrical Engineering from the University of Ottawa in 1983, an M.A.Sc. in Electrical Engineering from Queens University in 1984 and a Ph.D. in electrical engineering from Carleton University in 1989. From 1984 to 1989 he was employed as a communications systems engineer with Canadian Astronautics Limited and from 1990 to 1993 he worked as a satellite systems engineer for MPR Teltech Ltd. For the period spanning 1993 to 2009 he was employed by Nortel Networks in a variety of wireless systems and management roles within the CDMA and LTE basestation product groups. In 2010 he joined Ericsson Canada where he is currently Director of RAN Architecture and Performance in the North American CTO office. His interests include digital and wireless communications, signal processing and machine learning.