Exploring Temporal Similarity for Joint Computation and Communication in Online Distributed Optimization

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

***Abstract*—We consider online distributed optimization in a networked system, where multiple devices assisted by a server collaboratively minimize the accumulation of a sequence of global loss functions that can vary over time. To reduce the amount of communication, the devices send quantized and compressed local decisions to the server, resulting in noisy global decisions. Therefore, there exists a tradeoff between the optimization performance and the communication overhead. Existing works separately optimize computation and communication. In contrast, we jointly consider computation and communication over time, by proactively encouraging temporal similarity in the decision sequence to control the communication overhead. We propose an efficient algorithm, termed Online Distributed Optimization with Temporal Similarity (ODOTS), where the local decisions are both computation- and communication-aware. Furthermore, ODOTS uses a novel tunable virtual queue, which removes the commonly assumed Slater's condition through a modified Lyapunov drift analysis. ODOTS delivers provable performance bounds on both the optimization objective and constraint violation. Furthermore, we consider a variant of ODOTS with multi-step local gradient descent updates, termed ODOTS-MLU, and show that it provides improved performance bounds. As an example application, we apply both ODOTS and ODOTS-MLU to enable communication-efficient federated learning. Our experimental results based on canonical image classification demonstrate that ODOTS and ODOTS-MLU obtain higher classification accuracy and lower communication overhead compared with the current best alternatives for both convex and non-convex loss functions.**

***Index Terms*—Online optimization, federated learning, temporal similarity, long-term constraint, multi-step gradient.**

I. Introduction

Distributed optimization has become an essential tool for modern machine learning applications, which require ample storage, computation, and data. It avoids overburdening any single server and is robust to failures by coordinating multiple local devices to process the machine learning tasks. It can also alleviate privacy concerns by keeping the data local. However, the migration of optimization from the central server to local devices can incur a surge of communication overhead between them [2], [3]. The scarcity of communication resources can thus become a bottleneck for distributed machine learning applications. This calls for *communication-efficient* distributed optimization that integrate techniques from both machine learning and communications [4].

Most existing works on communication-efficient distributed learning consider computation and communication *separately* [5]-[20], *i.e.,* communication designs such as quantization and compression come *after* the machine-learning model parameters are already determined, for example, by standard gradient descent. However, since communication efficiency is strongly dependent on the information being transmitted [21], one can further improve the learning performance by proactively designing the model parameters for both learning accuracy and communication efficiency. In other words, *joint* consideration of computation and communication would take into fuller account the mutual impact between them.

Furthermore, most existing works focus on *offline* optimization, which does not allow time-varying loss functions or account for any long-term constraints. However, in many practical machine learning applications, *e.g.,* network traffic classification [22], dynamic user profiling [23], and real-time video analysis [24], random data samples arrive in a streaming fashion, and consequently the loss functions vary over time. These applications require *online optimization*, where we compute a sequence of optimization decisions that are adaptive to the unpredictable system dynamics over time [25], [26].

In this work, we aim to develop online distributed optimization solutions that jointly consider computation and communication over time. In particular, we are interested in a design that takes into account the interdependence of the optimization decisions over time to reduce the communication overhead, while providing performance guarantees on both the optimization and communication performance metrics. To achieve this goal, we must address several challenges: 1) Since the communication overhead depends on the local decisions transmitted from the devices to the server, when updating the local decisions, we must consider both their optimization performance and communication cost. 2) Lossy quantization substantially reduces the communication overhead but at the same time generates errors in the optimization decisions, and these errors propagate in the iterative computation process over time. 3) Due to the tight coupling between computation and communication, we must properly balance their joint impact

Juncheng Wang was with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 1A1, Canada. He is now with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, China (e-mail: jcwang@comp.hkbu.edu.hk).

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).

Ben Liang is with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 1A1, Canada (e-mail: liang@ece.utoronto.ca).

Gary Boudreau and Ali Afana are with Ericsson Canada, Ottawa, ON K2K 2V6, Canada (email: {gary.boudreau, ali.afana}@ericsson.com)

This work was supported in part by Ericsson Canada, the Natural Sciences and Engineering Research Council (NSERC) of Canada, and the Hong Kong Research Grants Council (RGC) Early Career Scheme (ECS) under grant 22200324.

An early version of this paper was presented at the IEEE International Conference on Computer Communications (INFOCOM), 2023 [1] (DOI: 10.1109/INFOCOM53939.2023.10229086).

on both the optimization performance and the convergence speed. 4) Both computation and communication needs to be properly formulated and designed to account for the unpredictable fluctuations in the environment over time.

Different from standard online distributed optimization that does not consider the communication efficiency, our decision update automatically balances the improvement in optimization and the cost in communication over time. Furthermore, we analyze the mutual impact between computation and communication over time, to provide performance bounds on both the computation and communication metrics for our proposed algorithms. Specifically, the main contributions of this paper are as follows:

- We formulate an online distributed optimization problem where the server computes a sequence of global optimization decisions to minimize the accumulated global loss, by aggregating the quantized and compressed local decisions communicated from the devices. To reduce the communication overhead, we encourage temporal similarity in the computed sequence of local decisions at the devices by enforcing an average long-term decision dis-similarity constraint. Thus, we consider both the optimization and communication performance metrics. To the best of our knowledge, this form of online distributed optimization with joint computation and communication consideration has not been studied in the literature.
- We propose an efficient algorithm to solve this problem, termed Online Distributed Optimization with Temporal Similarity (ODOTS). The local decisions yielded by ODOTS are adaptive to the unpredictable fluctuations of the loss functions while accounting for the decision dissimilarity constraint violation to limit the communication overhead. ODOTS achieves this via a novel tunable virtual queue that requires a modified Lyapunov drift analysis technique. Notably, this eliminates the requirement for Slater's condition, which is commonly assumed in existing virtual-queue-based online optimization algorithms.
- We analyze the tight coupling between computation and communication, and their joint impact on the optimization performance and convergence speed of ODOTS. Our analysis shows that with general local loss functions, for all sequences of time-varying weights on the devices, ODOTS achieves $\mathcal{O}(\max\{T^{\frac{1+\mu}{2}}, T^{\frac{3+\nu}{4}}\})$ performance gap to the centralized per-slot optimal decision sequence and $\mathcal{O}(\max\{T^{\frac{3+\mu}{4}}, T^{\frac{7+\nu}{8}}\})$ violation of the long-term decision dis-similarity constraint over T time slots, where μ represents the growth rate of the centralized per-slot optimizer and the quantization error, and ν measures the accumulated variation of the time-varying weights.
- We further consider a variant of ODOTS with multi-step local updates, termed ODOTS-MLU, to enable multiple steps of local gradient descent at the local devices before performing global decision aggregation at the central server. We analyze the impact of multi-step local gradient descent in ODOTS-MLU, and show that with strongly convex local loss functions, it improves the performance gap and constraint violation to $\mathcal{O}(\max\{T^{\mu}, T^{\frac{1+2\mu+\nu}{4}}\})$ and $\mathcal{O}(\max\{T^{\frac{5+2\mu+\nu}{8}}, T^{\frac{3+\nu}{4}}\})$, respectively.
- As an example application, we apply both ODOTS and ODOTS-MLU to enable communication-efficient federated learning. We study the impact of system parameters on the performance of ODOTS and ODOTS-MLU, by experimenting with canonical image classification datasets. Our experimental results demonstrate that for both convex and non-convex loss functions, ODOTS obtains higher test accuracy with lower communication overhead, compared with the current best alternatives under different scenarios. Performing multi-step local updates in ODOTS-MLU can yield better learning performance while incurring less communication overhead than ODOTS.

The rest of this paper is organized as follows. In Section II, we present the related work. Section III describes the system model and problem formulation. In Section IV, we present ODOTS and its decision updates. Performance bounds of ODOTS are provided in Section V. Then, we discuss the ODOTS-MLU variation with multi-step local gradient descent updates and study its performance in Section VI. The application of ODOTS and ODOTS-MLU to federated learning is presented in Section VII, followed by concluding remarks in Section VIII.

II. Related Work

A. Error-Free Distributed Optimization

Distributed optimization has been widely studied (see [27] and references therein). For example, offline distributed dual averaging and mirror descent algorithms were proposed in [28] and [29]. These two algorithms were respectively extended in [30] and [31] to the online setting. However, these works do not explicitly consider the communication efficiency.

Distributed approximate Newton-typed algorithm and alternating direction method of multipliers algorithm were proposed in [32] and [33] to reduce the number of iterations for efficient communication. Distributed gradient descent with event-triggered communication was considered in [34]. A general communication-efficient distributed dual coordinate ascent framework was proposed in [35], which used local computation in a primal-dual setting for reduced communication. However, the above works all assume error-free communication, and they ignore the opportunity to reduce the communication overhead via information similarity.

B. Communication-Efficient Distributed Learning

The original federated averaging algorithm increases the number of local updates to reduce the communication overhead [5]. An adaptive model aggregation approach was proposed in [6] under communication resource constraints. Quantization schemes have been adopted in distributed learning to reduce the number of transmitted bits by mapping the model parameters to a small set of discrete values. For example, 1-bit and multi-bit quantization methods were developed in [7] and [8]. Some other variations include error compensation [9],

variance reduction [10], and ternary quantization [11]. Sparsification schemes select a portion of the model parameters for communication. For example, threshold-based and top-k selection schemes were proposed in [12] and [13], with improvements in more recent studies such as [14]. Quantization and sparsification have also been applied simultaneously in [15]. However, the above works do not utilize the model similarity for more efficient communication.

Model similarity was utilized in [16] to further reduce the number of transmitted bits via conditional entropy coding. Event-triggered communication after quantization and sparsification was considered in [17], which transmits the decision parameters only if the amount of decision changes surpasses a predefined short-term limit. By using the autoencoder technique originally proposed for image compression, model compression was trained in [18]. Scalable sparsified model compression in combination with error-correction techniques was proposed in [19]. An innovation-based quantization scheme was proposed in [20]. However, the above works have the following fundamental limitations: 1) Their separate consideration of model training and compression overlooks the opportunity to select model parameters that can improve the communication efficiency; 2) Their offline optimization does not fully account for the unpredictable system variations during the learning process.

There is a recent branch of federated learning that utilizes analog communication, where model aggregation can be conducted over the air to reduce latency and communication overhead. For example, the aggregation error caused by noisy channel and model quantization was minimized through power allocation at each iteration in [36]. Online model updating under long-term power constraints was considered in [37], [38]. However, over-the-air model aggregation requires strict symbol-level synchronization among the devices and a large number of subchannels to separately communicate each of the model parameters. It is outside the scope of this work, which is designed for the common digital communication system.

C. Online Convex Optimization and Lyapunov Optimization

Due to the dynamic nature of the iterative computation and communication over time, a part of our solution resembles online convex optimization (OCO) [26], especially distributed constrained OCO with consensus [39]-[44]. However, the OCO framework mainly concerns delayed information feedback with error-free communication, which is inherently different from the joint online computation and communication framework in this work.

Since our work considers online optimization with a long-term constraint, it is also related to Lyapunov optimization [45], which minimizes a weighted sum of the loss and constraint functions at each time. However, directly minimizing the loss function can be difficult; *e.g.,* in distributed learning, it means directly solving for the optimal global model. Furthermore, ODOTS is a gradient-descent-typed algorithm, which substantially differs from Lyapunov optimization.

III. System Model and Problem Formulation

A. Online Distributed Optimization Objective

Consider a networked system consists of N local devices and a server. The system operates in a time-slotted fashion with time indexed by t. Let $f_t^n(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ be the *local* loss function of device n at time t, which may change over time. We are interested in an *online distributed* optimization problem with a *global* loss function $f_t(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ at each time t. It is defined as the weighted average of the local loss functions $\{f_t^n(\mathbf{x})\}$, given by

$$f_t(\mathbf{x}) \triangleq \sum_{n=1}^{N} w_t^n f_t^n(\mathbf{x}) \tag{1}$$

where $w_t^n \geq 0$ is the weight of device n, and satisfies $\sum_{n=1}^{N} w_t^n = 1$. Note that we also allow w_t^n to vary over time. The goal of online distributed optimization is to compute at the server a sequence of global decisions $\{\mathbf{x}_t\}$ that minimizes the accumulated global loss over a finite time horizon T, *i.e.,*

$$\min_{\{\mathbf{x}_t\}} \quad \sum_{t=1}^{T} f_t(\mathbf{x}_t). \tag{2}$$

As an example, in distributed learning, random training data may arrive at the devices over time as a continuous stream. At each time t, each device n collects its *local* dataset denoted by $\mathcal{D}_t^n$. The i-th data sample in $\mathcal{D}_t^n$ is represented by $(\mathbf{u}_t^{n,i}, v_t^{n,i})$, where $\mathbf{u}_t^{n,i}$ is a data feature vector and $v_t^{n,i}$ is its true label. Let $l(\mathbf{x}; \mathbf{u}_t^{n,i}, v_t^{n,i}) : \mathbb{R}^d \to \mathbb{R}$ be a training loss function to indicate how the learning model $\mathbf{x} \in \mathbb{R}^d$ performs on each data sample $(\mathbf{u}_t^{n,i}, v_t^{n,i})$, *e.g.,* it can be defined as the cross-entropy loss for logistic regression (see Section VII-B). In this case, the local loss function $f_t^n(\mathbf{x})$ is the averaged losses of the data samples in $\mathcal{D}_t^n$, given by

$$f_t^n(\mathbf{x}) = \frac{1}{|\mathcal{D}_t^n|} \sum_{i=1}^{|\mathcal{D}_t^n|} l(\mathbf{x}; \mathbf{u}_t^{n,i}, v_t^{n,i}) \tag{3}$$

where $|\mathcal{D}_t^n|$ is the cardinality of $\mathcal{D}_t^n$. When we set the local weight as $w_t^n = \frac{|\mathcal{D}_t^n|}{\sum_{m=1}^{N} |\mathcal{D}_t^m|}$ for each device n, the global loss $f_t(\mathbf{x})$ in (1) is equivalent to the averaged losses incurred by the global dataset $\bigcup_{n=1}^{N} \{\mathcal{D}_t^n\}$. Note that due to the fluctuations of the available computation resources, each device n may process different amounts of data samples over time, leading to a sequence of time-varying weights $\{w_t^n\}$.

B. Local Decision Quantization and Compression

For distributed minimization of the accumulated global loss, each device n generates a sequence of its local decisions $\{\mathbf{x}_t^n\}$. The server aggregates the local decisions into a sequence of global decisions. Transmitting the local decisions $\{\mathbf{x}_t^n\}$ from the N devices to the server can cause a large amount of communication overhead. This can be challenging and time-consuming, *e.g.,* for neural network training in the wireless environment, which can include millions of model parameters in each $\mathbf{x}_t^n$. In practical systems, communicating the local decisions from the devices to the server has been observed to be a significant performance bottleneck [2]-[4].

For efficient communication, the local decisions are usually quantized before transmission to the server. At each time t, after obtaining the local decision $\mathbf{x}_t^n$, each device n generates a quantized local decision $\hat{\mathbf{x}}_t^n$, by projecting each element of $\mathbf{x}_t^n$ to its closest point in a uniformly distributed grid with $s = 2^b$ quantization levels, where b is the quantization bit length.[1] In particular, the i-th element $x_t^{n,i}$ of $\mathbf{x}_t^n$ is quantized as $\hat{x}_t^{n,i}$, given by

$$\hat{x}_t^{n,i} = x_{\max}\text{sign}(x_t^{n,i}) \left\lfloor \frac{|x_t^{n,i}|}{x_{\max}}(s-1) + \frac{1}{2} \right\rfloor \tag{4}$$

where $x_{\max}$ is the maximum decision value, $\text{sign}(x) \in \{-1, 1\}$ returns the sign of x with $\text{sign}(0) = 1$, and $\lfloor a \rfloor$ is the floor function. Note that $x_{\max}$ can be easily enforced to the decision parameters by setting a set of short-term constraints on $\mathbf{x}_t^n$, given by

$$\mathcal{X} \triangleq \{\mathbf{x} : -x_{\max}\mathbf{1} \preceq \mathbf{x} \preceq x_{\max}\mathbf{1}\} \tag{5}$$

with $\mathbf{1}$ being a vector of all 1's.

Communicating the quantized local decisions requires efficient encoding to convert $\hat{\mathbf{x}}_t^n$ into bit streams. There are two common encoding approaches to compress $\hat{\mathbf{x}}_t^n$: 1) simple encoding that does not utilize any correlation in the sequence of decisions, such as Elias coding [46] and entropy coding [47]; and 2) more complicated encoding approach that utilizes the decision similarity, such as Wyner-Ziv coding [48] and conditional entropy coding [49], [50]. For example, consider the ideal conditional entropy coding. Let $H(\hat{\mathbf{x}}_t^n)$ be the marginal entropy of $\hat{\mathbf{x}}_t^n$. It measures the number of bits to communicate $\hat{\mathbf{x}}_t^n$ using entropy coding. Let $H(\hat{\mathbf{x}}_t^n|\hat{\mathbf{x}}_{t-1}^n)$ be the conditional entropy of $\hat{\mathbf{x}}_t^n$ given $\hat{\mathbf{x}}_{t-1}^n$, which represents the number of bits to communicate $\hat{\mathbf{x}}_t^n$ using conditional entropy coding, when $\hat{\mathbf{x}}_{t-1}^n$ is known at the destination. Due to the correlation between $\hat{\mathbf{x}}_{t-1}^n$ and $\hat{\mathbf{x}}_t^n$, their mutual information $H(\hat{\mathbf{x}}_t^n) - H(\hat{\mathbf{x}}_t^n|\hat{\mathbf{x}}_{t-1}^n)$ can be high. Therefore, conditional entropy coding can substantially reduce the communication overhead compared with independent entropy coding [49], [50].

Ideally, the quantized and compressed local decisions are losslessly conveyed to the server through standard channel coding techniques [51], [52]. However, due to lossy quantization, the server can only compute a *noisy* global decision $\hat{\mathbf{x}}_{t+1}$, given by

$$\hat{\mathbf{x}}_{t+1} = \sum_{n=1}^{N} w_t^n \hat{\mathbf{x}}_t^n = \mathbf{x}_{t+1} + \mathbf{n}_{t+1} \tag{6}$$

where $\mathbf{x}_{t+1} = \sum_{n=1}^{N} w_t^n \mathbf{x}_t^n$ is the *noiseless* global decision and $\mathbf{n}_{t+1} = \hat{\mathbf{x}}_{t+1} - \mathbf{x}_{t+1}$ is the global quantization error. The server then broadcasts $\hat{\mathbf{x}}_{t+1}$ to all N devices, and each device uses $\hat{\mathbf{x}}_{t+1}$ and its local loss function at time $t+1$ to compute the next local decision $\mathbf{x}_{t+1}^n$.

For ease of exposition, we assume the server uses standard channel coding techniques, such that $\hat{\mathbf{x}}_{t+1}$ can be received by all devices in an error-free fashion. However, lossy transmission of $\hat{\mathbf{x}}_{t+1}^n$ can be easily combined with our proposed algorithm and its performance analysis.

[1]Other techniques may be combined to further reduce the communication overhead. For example, each device n can first perform *sparsification* and then quantization to generate $\hat{\mathbf{x}}_t^n$. It will cause additional errors to the global decision $\hat{\mathbf{x}}_{t+1}$ (6). However, these errors can be included in $\mathbf{n}_{t+1}$ and do not impact our performance analysis later.

C. ODOTS Problem Formulation

Our goal is to jointly consider the global loss minimization and the local decision communication overhead over time. However, it is challenging to directly model a temporal-similarity encoding scheme during decision updating, since it depends on the joint probability density of $\hat{\mathbf{x}}_t^n$ and $\hat{\mathbf{x}}_{t-1}^n$. We observe that for different encoding schemes, an importance measure of the coding length is the difference between the information sources, *e.g.,* $\hat{\mathbf{x}}_t^n - \hat{\mathbf{x}}_{t-1}^n$, as it approximates the amount of new information to be encoded. Further note that the quantized local decision $\hat{\mathbf{x}}_t^n$ is generated only *after* computing the local decision $\mathbf{x}_t^n$. That is to say we can only optimize $\mathbf{x}_t^n$ instead of $\hat{\mathbf{x}}_t^n$ during the decision updating process. Therefore, we resort to limiting the amount of decision dis-similarity $\|\mathbf{x}_t^n - \hat{\mathbf{x}}_{t-1}^n\|^2$ to control the communication overhead, where $\|\cdot\|$ represents the Euclidean norm.

We aim at computing a sequence of local decisions $\{\mathbf{x}_t^n \in \mathcal{X}\}$ to minimize the accumulated loss yielded by the noisy global decision sequence $\{\hat{\mathbf{x}}_t\}$, while ensuring that the average long-term decision dis-similarity constraint is satisfied. This leads to the following online distributed optimization problem:

$$\begin{aligned} \mathbf{P1}: \quad \min_{\{\mathbf{x}_t^n \in \mathcal{X}\}} \quad & \sum_{t=1}^{T} f_t(\hat{\mathbf{x}}_t) \\ \text{s.t.} \quad & \frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) \le 0 \end{aligned} \tag{7}$$

where $\hat{\mathbf{x}}_t$ is the noisy global decision in (6) and $g_t^n(\mathbf{x})$ is the constraint function defined as

$$g_t^n(\mathbf{x}) \triangleq \|\mathbf{x} - \hat{\mathbf{x}}_{t-1}^n\|^2 - \epsilon \tag{8}$$

with ϵ being the allowed average decision dis-similarity. The long-term decision dis-similarity constraint (7) controls the total communication cost incurred during the entire optimization process, while allowing the communication cost to be distributed over time and devices. Compared with a strict short-term constraint on the decision dis-similarity at each time or device, the long-term constraint provides more flexibility in decision optimization, which can further reduce the total communication cost.

Note that **P1** is an online optimization problem due to the time-varying loss and constraint functions. In **P1**, the global loss $f_t(\hat{\mathbf{x}}_t)$ is determined by the quantized local decisions $\{\hat{\mathbf{x}}_t^n\}$. The decision dis-similarity constraint $g_t^n(\mathbf{x}_t^n)$ also depends on the quantized local decision $\hat{\mathbf{x}}_{t-1}^n$. Solving **P1** requires simultaneous consideration of computation and communication over time.

Furthermore, compared with the standard error-free optimization problem (2), the additional long-term constraint in (7) of **P1** requires a more complicated *constrained* online distributed optimization algorithm, especially since the local loss functions $\{f_t^n(\mathbf{x})\}$, weights $\{w_t^n\}$, and quantized decisions $\{\hat{\mathbf{x}}_t^n\}$ all can vary over time. It is therefore difficult

to obtain the globally optimal solution to **P1**, which would require *centralized* computation with *a priori* information of $\{f_t^n(\mathbf{x})\}$, $\{w_t^n\}$, and $\{\hat{\mathbf{x}}_t^n\}$ over T time slots.

A commonly used *centralized per-slot optimal* solution benchmark $\{\mathbf{x}_t^{\text{ctr}}\}$ for **P1** is given by [43], [53]-[56][2]

$$\mathbf{x}_t^{\text{ctr}} \in \arg\min_{\mathbf{x}\in\mathcal{X}}\{f_t(\mathbf{x})|g_t^n(\mathbf{x}) \leq 0, \forall n\}. \tag{9}$$

Note that $\mathbf{x}_t^{\text{ctr}}$ is computed without considering any errors, and it requires global information. Furthermore, as explained in Section II-C, directly minimizing $f_t(\mathbf{x})$ as in (9) can be difficult, especially for machine learning tasks. In this work, we aim to develop a constrained online distributed optimization algorithm to compute an online distributed solution sequence $\{\mathbf{x}_t^n\}$ to **P1** with sublinear performance gap to $\{\mathbf{x}_t^{\text{ctr}}\}$, *i.e.,* $\sum_{t=1}^{T}(f_t(\hat{\mathbf{x}}_t) - f_t(\mathbf{x}_t^{\text{ctr}})) = o(T)$ and sublinear constraint violation, *i.e.,* $\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) = o(T)$. Sublinearity in performance gap and constraint violation is important; it implies that the online distributed solution approaches to $\{\mathbf{x}_t^{\text{ctr}}\}$ in terms of its time-averaged performance and the long-term constraint is asymptotically satisfied.

IV. Online Distributed Optimization with Temporal Similarity

In this section, we present details of the ODOTS algorithm at the devices and the server. The local decisions yielded by ODOTS are both computation- and communication-aware, and are in closed forms that can be computed efficiently.

A. Tunable Virtual Queue

We first introduce a novel *tunable* virtual queue Q_t^n at each device n to account for the long-term constraint (7) in **P1**, with the following updating rule:

$$Q_{t+1}^n = \left[(1-\gamma^2)Q_t^n + \gamma\eta g_t^n(\mathbf{x}_t^n)\right]_+ \tag{10}$$

where $\gamma \in (0,1)$ is a tuning factor on the virtual queue, $\eta > 0$ is a weighting factor on the constraint function, and $[a]_+ = \max\{a, 0\}$ is a projection operator.[3] The role of Q_t^n is similar to a Lagrangian multiplier for **P1** or a backlog queue for the constraint violation. It measures the amount of constraint violation and automatically balances the loss minimization and constraint satisfaction over time. The concept of virtual queue was also used in [45] and [53]-[58] for Lyapunov optimization and *centralized* constrained OCO. However, unique to our virtual queue updating rule (10), there is an additional $-\gamma^2 Q_t^n$ term to prevent Q_{t+1}^n from becoming too large, and the constraint violation $g_t^n(\mathbf{x}_t^n)$ is scaled by $\gamma\eta$ to control how fast the virtual queue varies over time.[4]

This new tunable virtual queue updating rule (10) will be shown later in Section V-B to provide a simple upper bound on Q_t^n, which does not require the Slater's condition that is commonly assumed for the virtual-queue-based online optimization algorithms [45], [53]-[58].[5] However, without the Slater's condition, we can no longer directly transfer the virtual queue upper bound to the constraint violation bound. To overcome this technical difficulty, as shown later in Section V-B, we will bound the constraint violation using a new modified Lyapunov drift analysis technique.

B. Decomposition of *P1*

We convert **P1** into a set of *per-device per-slot* optimization problems, one for each device n at each time t, given by

$$\mathbf{P2}^n: \ \min_{\mathbf{x}\in\mathcal{X}} \ \langle \nabla f_t^n(\hat{\mathbf{x}}_t), \mathbf{x} - \hat{\mathbf{x}}_t\rangle + \alpha\|\mathbf{x} - \hat{\mathbf{x}}_t\|^2 + \eta Q_t^n g_t^n(\mathbf{x})$$

where $\alpha > 0$ is a step-size parameter that controls the gradient descent step and $\langle \mathbf{a}, \mathbf{b}\rangle$ represents the inner product of vectors $\mathbf{a}$ and $\mathbf{b}$. We will explain in Section V that $\mathbf{P2}^n$ is equivalent to minimizing an upper bound on a modified drift plus penalty plus violation term (see (29)) to trade off loss minimization and constraint violation over time. We will further bound the performance of the solutions to $\mathbf{P2}^n$ to that of **P1**, in terms of the accumulated loss and constraint violation.

Note that $\mathbf{P2}^n$ is a local optimization problem using the current local loss function $f_t^n(\mathbf{x})$, tunable virtual queue length Q_t^n, and the previous quantized local decision $\hat{\mathbf{x}}_{t-1}^n$. It is under short-term constraints only. Furthermore, the local gradient $\nabla f_t^n(\hat{\mathbf{x}}_t)$ is evaluated using the noisy global decision $\hat{\mathbf{x}}_t$ and the regularization $\|\mathbf{x} - \hat{\mathbf{x}}_t\|^2$ is also on $\hat{\mathbf{x}}_t$ to enable local gradient descent based on $\hat{\mathbf{x}}_t$. Compared with the original **P1**, the long-term decision dis-similarity constraint has been converted into controlling $g_t^n(\mathbf{x}_t^n)$ to maintain the queue stability as shown in the third term of the objective in $\mathbf{P2}^n$. The constraint function $g_t^n(\mathbf{x})$ is convex and the feasible set $\mathcal{X}$ is affine with respect to (w.r.t.) $\mathbf{x}$. Furthermore, the first two terms in the objective of $\mathbf{P2}^n$ are affine and convex w.r.t. $\mathbf{x}$, respectively. Therefore, $\mathbf{P2}^n$ is a convex optimization problem and therefore can be solved efficiently.

[2]The solution benchmark used in [39]-[42] is *fixed* over time.

[3]As will be shown later in Section V-E, η as a constant does not change the growth rate of the performance gap or the constraint violation. However, η can be useful in some numerical experiments as a hyper parameter, especially when the values of the loss and constraint functions differ too much.

[4]The tuning factor γ can be seen as a *virtual* Slater constant, which appears later in the denominator of the virtual queue upper bound (22) in Lemma 2. Note that we require $-\gamma^2 Q_t^n$ instead of $-\gamma Q_t^n$ to maintain a proper bound on Q_{t+1}^n (see the proof of Lemma 2). Furthermore, we remark that this term and the quadratic penalty on the Lagrange multiplier in [59] are two different optimization approaches to accomplish a similar purpose of preventing the virtual queue or the Lagrange multiplier from being too large. The approach in [59] is designed for error-free centralized OCO with fixed long-term constraints, while our approach deals with the online distributed optimization with time-varying constraints. As such, our performance analyses are substantially different from those in [59].

[5]The Slater's condition precludes dealing with equality constraints and can be restrictive to many practical applications. For example, it does not hold if we set $\epsilon = 0$ in the constraint function (8). The virtual-queue-based online optimization algorithm in [54] achieved sublinear performance bounds for *centralized* OCO without the Slater's condition, but it relies on two additional assumptions requiring sublinear variation of the loss functions and of the optimal dual points.

Algorithm 1 ODOTS: Device n's algorithm

1: Initialize $\hat{\mathbf{x}}_1 = \mathbf{0}$ and $Q_1^n = 0$.
2: For each t, do:
3: Update local decision $\mathbf{x}_t^n$ by solving $\mathbf{P2}^n$ via (11).
4: Update local virtual queue Q_{t+1}^n via (10).
5: Update quantized local decision $\hat{\mathbf{x}}_t^n$ via (4).
6: Transmit $\hat{\mathbf{x}}_t^n$ via conditional entropy coding.

C. ODOTS Algorithm

In the following, we provide a closed-form solution to $\mathbf{P2}^n$. It is easy to see that the gradient of the objective function of $\mathbf{P2}^n$ is

$$\nabla f_t^n(\hat{\mathbf{x}}_t) + 2\alpha(\mathbf{x} - \hat{\mathbf{x}}_t) + 2\eta Q_t^n(\mathbf{x} - \hat{\mathbf{x}}_{t-1}^n).$$

Then, the optimal solution to $\mathbf{P2}^n$ can be obtained by setting this gradient to zero and then projecting it onto $\mathcal{X}$. The resulting local decision update is in closed form, given by

$$\mathbf{x}_t^n = \Big[\frac{\alpha}{\alpha+\eta Q_t^n}\Big(\hat{\mathbf{x}}_t + \frac{\eta Q_t^n}{\alpha}\hat{\mathbf{x}}_{t-1}^n - \frac{1}{2\alpha}\nabla f_t^n(\hat{\mathbf{x}}_t)\Big)\Big]_{-x_{\max}\mathbf{1}}^{x_{\max}\mathbf{1}} \quad (11)$$

where $[\mathbf{a}]_{\mathbf{b}}^{\mathbf{c}} = \min\{\mathbf{c}, \max\{\mathbf{a}, \mathbf{b}\}\}$ is an entry-wise projection operator.

Note that the local decision update (11) is scaled by a factor $\frac{\alpha}{\alpha+\eta Q_t^n}$ that depends on the ratio of the tunable virtual queue length Q_t^n and the gradient descent step size α. The values of Q_t^n and α tune the relative weights on the global decision $\hat{\mathbf{x}}_t$ and the previous quantized local decision $\hat{\mathbf{x}}_{t-1}^n$ on the new local decision update. When Q_t^n is small, *i.e.,* the scale on the decision update $\frac{\alpha}{\alpha+\eta Q_t^n}$ is close to 1 and the weight $\frac{\eta Q_t^n}{\alpha}$ on $\hat{\mathbf{x}}_{t-1}$ is close to 0, (11) becomes the standard projected local gradient descent based on the noisy global decision

$$\mathbf{x}_t^n = \Big[\hat{\mathbf{x}}_t - \frac{1}{2\alpha}\nabla f_t^n(\hat{\mathbf{x}}_t)\Big]_{-x_{\max}\mathbf{1}}^{x_{\max}\mathbf{1}} \quad (12)$$

to minimize the loss. Otherwise, when Q_t^n is relatively large compared with α, *i.e.,* $\frac{\alpha}{\alpha+\eta Q_t^n}$ is small and $\frac{\eta Q_t^n}{\alpha}$ is large, the gradient descent is *slowed down* and (11) is close to $\hat{\mathbf{x}}_{t-1}^n$, which reduces the communication overhead due to the resulting high interdependence between $\hat{\mathbf{x}}_t^n$ and $\hat{\mathbf{x}}_{t-1}^n$. The virtual queue Q_t^n gradually adjusts the actual gradient descent step size $\frac{1}{2(\alpha+\eta Q_t^n)}$ based on the decision dis-similarity constraint violation to minimize the loss.[6] Therefore, the local decision update by ODOTS is both computation- and communication aware, *i.e.,* automatically balancing the improvement in optimization and the cost in communication over time.

We summarize the devices' algorithm and the server's algorithm in Algorithms 1 and 2. The choices of algorithm parameters α, γ, and η will be discussed in Section V-E, after we derive the bounds on the performance gap and constraint violation for ODOTS.

Remark 1. The computational complexity of the ODOTS algorithm is mainly determined by the solution to $\mathbf{P2}^n$ in (11). We note that (11) is in closed form, and it contains only a single evaluation of the gradient of $f_t^n(\mathbf{x})$. Therefore, ODOTS is highly efficient, having computational complexity similar to the standard gradient descent algorithm.

[6]When the virtual queue becomes large, it means the decision dis-similarity constraint becomes tight. The gradient descent step has to be small to reduce the decision dis-similarity, and this causes the gradient descent update to slow down, but not necessarily stop.

Algorithm 2 ODOTS: Server's algorithm

1: Initialize and broadcast α, γ, and η.
2: For each t, do:
3: Receive quantized local decisions $\{\hat{\mathbf{x}}_t^n\}$.
4: Update noisy global decision $\hat{\mathbf{x}}_{t+1}$ via (6).
5: Broadcast $\hat{\mathbf{x}}_{t+1}$ to all devices.

V. Performance Bounds of ODOTS

In this section, we further show that ODOTS provides strong performance guarantees in both the optimization objective and the temporal decision dis-similarity constraint. In particular, the unique design of ODOTS requires new analysis techniques to account for the impact of the noisy decision update and the tunable virtual queue.

A. Preliminaries

We make the following standard assumptions in the performance analysis of ODOTS.

Assumption 1. The local loss function $f_t^n(\mathbf{x})$ is convex, *i.e.,*

$$f_t^n(\mathbf{y}) \geq f_t^n(\mathbf{x}) + \langle \nabla f_t^n(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle, \ \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}, \forall n, \forall t. \quad (13)$$

Assumption 2. The local loss function $f_t^n(\mathbf{x})$ has bounded gradient $\nabla f_t^n(\mathbf{x})$: $\exists D > 0$, *s.t.,*

$$\|\nabla f_t^n(\mathbf{x})\| \leq D, \quad \forall \mathbf{x} \in \mathcal{X}, \forall n, \forall t. \quad (14)$$

Assumptions 1 and 2 are common in existing studies on online distributed optimization. Nevertheless, later in Section VII-C, we empirically show that ODOTS also works well for general non-convex loss functions.

The following lemma shows that **P1** satisfies the following properties: 1) The feasible set $\mathcal{X}$ is bounded; 2) The quantization error $\mathbf{n}_t$ is bounded; 3) The constraint function $g_t^n(\mathbf{x})$ is bounded.

Lemma 1. Our formulated **P1** satisfies the following:

$$\|\mathbf{x} - \mathbf{y}\| \leq R, \quad \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}, \quad (15)$$

$$\|\mathbf{n}_t\| \leq \delta, \quad \forall t, \quad (16)$$

$$|g_t^n(\mathbf{x})| \leq G, \quad \forall \mathbf{x} \in \mathcal{X}, \forall n, \forall t. \quad (17)$$

where $R = 2\sqrt{d}x_{\max}$, $\delta = \frac{R}{4(s-1)}$, and $G = \max\{\epsilon, (R+\delta)^2 - \epsilon\}$.

Proof: We first prove (15). For any $\mathbf{x}, \mathbf{y} \in \mathcal{X}$, we have

$$\|\mathbf{x} - \mathbf{y}\| \overset{(a)}{\leq} \|\mathbf{x}\| + \|\mathbf{y}\| \overset{(b)}{\leq} 2\sqrt{d}x_{\max} \quad (18)$$

where (a) is because of the triangle inequality; and (b) follows from the definition of the set of short-term constraints $\mathcal{X}$ in (5) and that $\mathbf{x}$ is of d dimensions, such that $\|\mathbf{x}\|^2 \leq dx_{\max}^2, \forall \mathbf{x} \in \mathcal{X}$.

We now prove (16). From the definition of the quantization error $\mathbf{n}_t$ in (6), we have

$$\|\mathbf{n}_t\| = \|\hat{\mathbf{x}}_t - \mathbf{x}_t\| = \Big\| \sum_{n=1}^{N} w_{t-1}^n(\hat{\mathbf{x}}_{t-1}^n - \mathbf{x}_{t-1}^n)\Big\|$$

$$\overset{(a)}{\leq} \sum_{n=1}^{N} w_{t-1}^n \|\hat{\mathbf{x}}_{t-1}^n - \mathbf{x}_{t-1}^n\| \overset{(b)}{\leq} \sum_{n=1}^{N} w_{t-1}^n \Big(\sqrt{d}\frac{x_{\max}}{2(s-1)}\Big)$$
$$\overset{(c)}{\leq} \sqrt{d}\frac{x_{\max}}{2(s-1)} \overset{(d)}{=} \frac{R}{4(s-1)} \tag{19}$$

where (a) follows from the triangle inequality and $\|\mathbf{ab}\| \leq \|\mathbf{a}\|\|\mathbf{b}\|$; (b) is because $\mathbf{x}_{t-1}^n$ is element-wise quantized as $\hat{\mathbf{x}}_{t-1}^n$ via (6), and each quantized element $\hat{x}_{t-1}^{n,i}$ of $\hat{\mathbf{x}}_{t-1}^n$ in (4) has a maximum quantization error $\frac{x_{\max}}{2(s-1)}$, with s being the number of quantization levels; (c) is because $\sum_{n=1}^{N} w_t^n = 1, \forall t$; and (d) follows from the definition of R.

Finally, we prove (17). For any $\mathbf{x} \in \mathcal{X}$, we have

$$\|\mathbf{x} - \hat{\mathbf{x}}_{t-1}^n\|^2 = \|\mathbf{x} - \mathbf{x}_{t-1}^n + \mathbf{x}_{t-1}^n - \hat{\mathbf{x}}_{t-1}^n\|^2$$
$$\overset{(a)}{\leq} \big(\|\mathbf{x} - \mathbf{x}_{t-1}^n\| + \|\mathbf{x}_{t-1}^n - \hat{\mathbf{x}}_{t-1}^n\|\big)^2 \overset{(b)}{\leq} (R+\delta)^2 \tag{20}$$

where (a) is because of the triangle inequality, and (b) follows from (15) and (16). Further note that

$$|g_t^n(\mathbf{x})| = \big|\|\mathbf{x} - \hat{\mathbf{x}}_{t-1}^n\|^2 - \epsilon\big|$$
$$\leq \max\big\{\epsilon, \max\{\|\mathbf{x} - \hat{\mathbf{x}}_{t-1}^n\|^2\} - \epsilon\big\}, \tag{21}$$

we have (17). ∎

B. Bounds on the Tunable Virtual Queue and Modified Lyapunov Drift

We first provide an upper bound on the tunable virtual queue.

Lemma 2. The tunable virtual queue is upper bounded for any $\mathbf{x}_t^n \in \mathcal{X}$, n, and t by

$$Q_t^n \leq \frac{\eta G}{\gamma}. \tag{22}$$

Proof: We prove by induction. We have $Q_1^n = 0 \leq \frac{\eta G}{\gamma}$ by initialization. Suppose $Q_\tau^n \leq \frac{\eta G}{\gamma}$ for some $\tau \geq 1$. We have

$$Q_{\tau+1}^n \overset{(a)}{\leq} |(1-\gamma^2)Q_\tau^n + \gamma\eta g_\tau^n(\mathbf{x}_\tau^n)|$$
$$\overset{(b)}{\leq} (1-\gamma^2)Q_\tau^n + \gamma\eta|g_\tau^n(\mathbf{x}_\tau^n)|$$
$$\overset{(c)}{\leq} (1-\gamma^2)\frac{\eta G}{\gamma} + \gamma\eta G = \frac{\eta G}{\gamma}$$

where (a) follows directly from the tunable virtual queue updating rule (10); (b) is because of $Q_t^n \geq 0, \forall t$, $\gamma \in (0,1)$, and the triangle inequality; and (c) follows from induction and the bound on $g_t^n(\mathbf{x})$ in (17). ∎

Although our tunable virtual queue updating rule (10) yields a simple upper bound on Q_t^n in (22), unfortunately it also breaks the key connection between the virtual queue bound and the constraint violation bound used by [45], [53]-[57] in their performance analysis. To proceed with our analysis, we define a *modified* Lyapunov drift for each device n as

$$\Theta_t^n = \frac{1}{2\gamma}(Q_{t+1}^n - U)^2 - \frac{1}{2\gamma}(Q_t^n - U)^2. \tag{23}$$

where $U \geq 0$ is a *virtual* regularization factor on the quadratic Lyapunov function. Note that U is introduced only to enable our performance bound analysis, and ODOTS does not require the value of U to run. Using the result in Lemma 2, we provide an upper bound on Θ_t^n, which regains the connection between the tunable virtual queue and the constraint violation.

Lemma 3. The modified Lyapunov drift is upper bounded for any $\mathbf{x}_t^n \in \mathcal{X}$, n, and t by

$$\Theta_t^n \leq \eta Q_t^n g_t^n(\mathbf{x}_t^n) - U\eta g_t^n(\mathbf{x}_t^n) + 2\gamma\eta^2 G^2 + \frac{\gamma}{2}U^2. \tag{24}$$

Proof: From the tunable virtual queue updating rule in (10), and the fact that $|[a]_+ - [b]_+| \leq |a-b|$, we have

$$(Q_{t+1}^n - U)^2 \leq \big((1-\gamma^2)Q_t^n + \gamma\eta g_t^n(\mathbf{x}_t^n) - U\big)^2$$
$$= \big((Q_t^n - U) + \gamma(\eta g_t^n(\mathbf{x}_t^n) - \gamma Q_t^n)\big)^2$$
$$= (Q_t^n - U)^2 + \gamma^2\big(\eta g_t^n(\mathbf{x}_t^n) - \gamma Q_t^n\big)^2 + 2\gamma\eta Q_t^n g_t^n(\mathbf{x}_t^n)$$
$$- 2\gamma\eta U g_t^n(\mathbf{x}_t^n) - 2\gamma^2(Q_t^n - U)Q_t^n. \tag{25}$$

We now bound the terms on the right-hand side (RHS) of (25). From the triangle inequality, the bound on $g_t^n(\mathbf{x})$ in (17), and the bound on Q_t^n in (22), we have

$$|\eta g_t^n(\mathbf{x}_t^n) - \gamma Q_t^n| \leq \eta|g_t^n(\mathbf{x}_t^n)| + \gamma Q_t^n$$
$$\leq \eta G + \gamma\frac{\eta G}{\gamma} = 2\eta G. \tag{26}$$

For the last term on the RHS of (25), we have

$$-2(Q_t^n - U)Q_t^n = U^2 - (Q_t^n)^2 - (Q_t^n - U)^2 \leq U^2. \tag{27}$$

Substituting (26) and (27) into (25), and rearranging terms, we have

$$(Q_{t+1}^n - U)^2 - (Q_t^n - U)^2$$
$$\leq 2\gamma\eta Q_t^n g_t^n(\mathbf{x}_t^n) - 2\gamma\eta U g_t^n(\mathbf{x}_t^n) + \gamma^2(2\eta G)^2 + \gamma^2 U^2. \tag{28}$$

Dividing both sides of (27) by 2γ, and from the definition of the modified Lyapunov drift Θ_t^n in (23), we prove (24). ∎

From the upper bound on Θ_t^n in (24) and noting that $2\gamma\eta^2G^2 + \frac{\gamma}{2}U^2$ in (24) is a constant, we can see that solving $\mathbf{P2}^n$ for each device n is equivalent to minimizing an upper bound on the following modified *drift plus penalty plus violation* term at each time t:

$$\underbrace{\Theta_t^n}_{\text{drift}} + \underbrace{\langle\nabla f_t^n(\hat{\mathbf{x}}_t), \mathbf{x}_t^n - \hat{\mathbf{x}}_t\rangle + \alpha\|\mathbf{x}_t^n - \hat{\mathbf{x}}_t\|^2}_{\text{penalty}} + \underbrace{U\eta g_t^n(\mathbf{x}_t^n)}_{\text{violation}}. \tag{29}$$

This is similar to the Lyapunov optimization approach [45] that minimizes a drift plus penalty term at each time. However, the penalty term in standard Lyapunov optimization is the loss function itself. As explained in Section II-C, for machine learning tasks in distributed learning, this means finding the optimal model within a single time slot and is impossible in general. Instead, we use the penalty term $\langle\nabla f_t^n(\hat{\mathbf{x}}_t), \mathbf{x}_t^n - \hat{\mathbf{x}}_t\rangle + \alpha\|\mathbf{x}_t^n - \hat{\mathbf{x}}_t\|^2$ to enable local gradient descent for the global loss minimization. Note that when the virtual penalty factor U on the quadratic Lyapunov function is nonzero, (29) also includes a violation term $U\eta g_t^n(\mathbf{x}_t^n)$. This is introduced to help bound the constraint violation, since the upper bound (22) on our tunable virtual queue is not directly transferable to the constraint violation bound anymore.

C. Bound on the Performance Gap

Using the results in Lemmas 1-3, the following lemma provides an upper bound on the weighted sum of the per-slot local loss and constraint violation $f_t^n(\hat{\mathbf{x}}_t)+U\eta g_t^n(\mathbf{x}_t^n)$ by ODOTS.

Lemma 4. The weighted sum of the per-slot local loss and constraint violation yielded by ODOTS is upper bounded by

$$\begin{aligned} &f_t^n(\hat{\mathbf{x}}_t)+U\eta g_t^n(\mathbf{x}_t^n)\\ &\le f_t^n(\mathbf{x}_t^{\text{ctr}})+\frac{D^2}{4\alpha}+2\gamma\eta^2G^2+\frac{\gamma}{2}U^2-\Theta_t^n\\ &\quad+\alpha\big(\phi_t+\psi_t^n+\|\mathbf{n}_t\|^2+2R(\|\mathbf{n}_t\|+\pi_t)\big),\ \forall n,\forall t \end{aligned} \tag{30}$$

where $\phi_t \triangleq \|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t\|^2-\|\mathbf{x}_{t+1}^{\text{ctr}}-\mathbf{x}_{t+1}\|^2$, $\psi_t^n \triangleq \|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_{t+1}\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2$, and $\pi_t \triangleq \|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_{t+1}^{\text{ctr}}\|$.

Proof: We require the following lemma, which is copied from Lemma 2.8 in [26].

Lemma 5. ([26, Lemma 2.8]) Let $\mathcal{Z}\in\mathbb{R}^z$ be a nonempty convex set. Let $h(\mathbf{z}):\mathbb{R}^z\to\mathbb{R}$ be a 2ϱ-strongly convex function over $\mathcal{Z}$ w.r.t. any norm $\|\cdot\|'$. Let $\mathbf{w}=\arg\min_{\mathbf{z}\in\mathcal{Z}}h(\mathbf{z})$. Then, for any $\mathbf{u}\in\mathcal{Z}$, we have $h(\mathbf{w})\le h(\mathbf{u})-\varrho\|\mathbf{u}-\mathbf{w}\|'^2$.

The objective function of $\mathbf{P2}^n$ is 2α-strongly convex over $\mathcal{X}$ w.r.t. $\|\cdot\|$ due to the regularization term $\alpha\|\mathbf{x}-\hat{\mathbf{x}}_t\|^2$. Since $\mathbf{x}_t^n$ is the optimal solution to $\mathbf{P2}^n$, from Lemma 5, we have

$$\begin{aligned} &\langle\nabla f_t^n(\hat{\mathbf{x}}_t),\mathbf{x}_t^n-\hat{\mathbf{x}}_t\rangle+\alpha\|\mathbf{x}_t^n-\hat{\mathbf{x}}_t\|^2+\eta Q_t^n g_t^n(\mathbf{x}_t^n)\\ &\le\langle\nabla f_t^n(\hat{\mathbf{x}}_t),\mathbf{x}_t^{\text{ctr}}-\hat{\mathbf{x}}_t\rangle+\eta Q_t^n g_t^n(\mathbf{x}_t^{\text{ctr}})\\ &\quad+\alpha(\|\mathbf{x}_t^{\text{ctr}}-\hat{\mathbf{x}}_t\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2). \end{aligned} \tag{31}$$

We now bound the last term on the RHS of (31). We have

$$\begin{aligned} &\|\mathbf{x}_t^{\text{ctr}}-\hat{\mathbf{x}}_t\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2\\ &=\|\mathbf{x}_t^{\text{ctr}}-\hat{\mathbf{x}}_t\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_{t+1}\|^2\\ &\quad+\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_{t+1}\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2\\ &=\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t+\mathbf{x}_t-\hat{\mathbf{x}}_t\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_{t+1}^{\text{ctr}}+\mathbf{x}_{t+1}^{\text{ctr}}-\mathbf{x}_{t+1}\|^2\\ &\quad+(\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_{t+1}\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2)\\ &\overset{(a)}{\le}\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t\|^2+\|\mathbf{x}_t-\hat{\mathbf{x}}_t\|^2+2\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t\|\|\mathbf{x}_t-\hat{\mathbf{x}}_t\|\\ &\quad-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_{t+1}^{\text{ctr}}\|^2-\|\mathbf{x}_{t+1}^{\text{ctr}}-\mathbf{x}_{t+1}\|^2\\ &\quad+2\|\mathbf{x}_{t+1}^{\text{ctr}}-\mathbf{x}_{t+1}\|\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_{t+1}^{\text{ctr}}\|+\psi_t^n\\ &\overset{(b)}{\le}(\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t\|^2-\|\mathbf{x}_{t+1}^{\text{ctr}}-\mathbf{x}_{t+1}\|^2)+\|\mathbf{n}_t\|^2\\ &\quad+2\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t\|\|\mathbf{n}_t\|+2\|\mathbf{x}_{t+1}^{\text{ctr}}-\mathbf{x}_{t+1}\|\pi_t+\psi_t^n\\ &\overset{(c)}{\le}\phi_t+\|\mathbf{n}_t\|^2+2R\|\mathbf{n}_t\|+2R\pi_t+\psi_t^n. \end{aligned} \tag{32}$$

where (a) follows from $\|\mathbf{a}+\mathbf{b}\|^2\le\|\mathbf{a}\|^2+\|\mathbf{b}\|^2+2\|\mathbf{a}\|\|\mathbf{b}\|$, $-\|\mathbf{a}+\mathbf{b}\|^2\le-\|\mathbf{a}\|^2-\|\mathbf{b}\|^2+2\|\mathbf{a}\|\|\mathbf{b}\|$, and the definition of ψ_t^n; (b) is because of the definitions of $\mathbf{n}_t$ and π_t; and (c) follows from the bound of $\mathcal{X}$ in (15) and the definition of ϕ_t.

Substituting (32) into (31) and rearranging terms, we have

$$\begin{aligned} &-\langle\nabla f_t^n(\hat{\mathbf{x}}_t),\mathbf{x}_t^{\text{ctr}}-\hat{\mathbf{x}}_t\rangle+\eta Q_t^n g_t^n(\mathbf{x}_t^n)\\ &\le-\langle\nabla f_t^n(\hat{\mathbf{x}}_t),\mathbf{x}_t^n-\hat{\mathbf{x}}_t\rangle-\alpha\|\mathbf{x}_t^n-\hat{\mathbf{x}}_t\|^2+\eta Q_t^n g_t^n(\mathbf{x}_t^{\text{ctr}})\\ &\quad+\alpha\big(\phi_t+\psi_t^n+\|\mathbf{n}_t\|^2+2R(\|\mathbf{n}_t\|+\pi_t)\big). \end{aligned} \tag{33}$$

From the convexity of $f_t^n(\mathbf{x})$, we have

$$f_t^n(\hat{\mathbf{x}}_t)-f_t^n(\mathbf{x}_t^{\text{ctr}})\le-\langle\nabla f_t^n(\hat{\mathbf{x}}_t),\mathbf{x}_t^{\text{ctr}}-\hat{\mathbf{x}}_t\rangle. \tag{34}$$

Completing the square and noting that $\nabla f_t^n(\mathbf{x})$ is bounded in (14), we have

$$\begin{aligned} &-\langle\nabla f_t^n(\hat{\mathbf{x}}_t),\mathbf{x}_t^n-\hat{\mathbf{x}}_t\rangle-\alpha\|\mathbf{x}_t^n-\hat{\mathbf{x}}_t\|^2\\ &=-\Big\|\frac{\nabla f_t^n(\hat{\mathbf{x}}_t)}{2\sqrt{\alpha}}+\sqrt{\alpha}(\mathbf{x}_t^n-\hat{\mathbf{x}}_t)\Big\|^2+\frac{1}{4\alpha}\|\nabla f_t^n(\hat{\mathbf{x}}_t)\|^2\\ &\le\frac{1}{4\alpha}\|\nabla f_t^n(\hat{\mathbf{x}}_t)\|^2\le\frac{D^2}{4\alpha}. \end{aligned} \tag{35}$$

Substituting (34) and the bound on $\eta Q_t^n g_t^n(\mathbf{x}_t^n)$ in (24) of Lemma 3 into the left-hand side (LHS) of (33), and (35) into the RHS of (33), we have

$$\begin{aligned} &f_t^n(\hat{\mathbf{x}}_t)-f_t^n(\mathbf{x}_t^{\text{ctr}})+U\eta g_t^n(\mathbf{x}_t^n)\\ &\le-\Theta_t^n+2\gamma\eta^2G^2+\frac{\gamma}{2}U^2+\frac{D^2}{4\alpha}+\eta Q_t^n g_t^n(\mathbf{x}_t^{\text{ctr}})\\ &\quad+\alpha\big(\phi_t+\psi_t^n+\|\mathbf{n}_t\|^2+2R(\|\mathbf{n}_t\|+\pi_t)\big). \end{aligned}$$

Rearranging terms of the above inequality and noting that $\eta Q_t^n g_t^n(\mathbf{x}_t^{\text{ctr}})\le 0$, which is because $Q_t^n\ge 0$ and $g_t^n(\mathbf{x}_t^{\text{ctr}})\le 0$ from the definition of $\mathbf{x}_t^{\text{ctr}}$ in (9), we have (30). ∎

Based on the result in Lemma 4, we provide an upper bound on the performance gap to the centralized per-slot optimal solution sequence $\{\mathbf{x}_t^{\text{ctr}}\}$ for ODOTS in the following theorem.

Theorem 1. Under Assumptions 1 and 2, the performance gap to $\{\mathbf{x}_t^{\text{ctr}}\}$ by ODOTS is upper bounded by

$$\begin{aligned} \sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t)-f_t(\mathbf{x}_t^{\text{ctr}})\big)&\le\frac{D^2T}{4\alpha}+2\gamma\eta^2G^2T+\frac{\eta^2G^2\Omega_T}{2\gamma^3}\\ &\quad+\alpha\big(R^2+\Lambda_{2,T}+2R(\Lambda_T+\Pi_T)\big) \end{aligned} \tag{36}$$

where $\Pi_T\triangleq\sum_{t=1}^{T}\pi_t$, $\Omega_T\triangleq\sum_{t=1}^{T}\sum_{n=1}^{N}(w_{t+1}^n-w_t^n)$, $\Lambda_T\triangleq\sum_{t=1}^{T}\|\mathbf{n}_t\|$, and $\Lambda_{2,T}\triangleq\sum_{t=1}^{T}\|\mathbf{n}_t\|^2$.

Proof: Multiplying both sides of (30) by w_t^n, setting $U=0$, and summing the resulting inequality over n and t, we have

$$\begin{aligned} \sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t)-f_t(\mathbf{x}_t^{\text{ctr}})\big)&\overset{(a)}{\le}\frac{D^2T}{4\alpha}+2\gamma\eta^2G^2T-\sum_{t=1}^{T}\sum_{n=1}^{N}w_t^n\Theta_t^n\\ &\quad+\alpha\sum_{t=1}^{T}\Big(\phi_t+\sum_{n=1}^{N}w_t^n\psi_t^n\Big)+\alpha\big(\Lambda_{2,T}+2R(\Lambda_T+\Pi_T)\big) \end{aligned} \tag{37}$$

where (a) follows from the definitions of $\Lambda_{2,T}$, Λ_T, and Π_T.

We now bound the terms on the RHS of (37). From the definition of the modified Lyapunov drift Θ_t^n in (24), we have

$$\begin{aligned} &-\sum_{t=1}^{T}\sum_{n=1}^{N}w_t^n\Theta_t^n=\frac{1}{2\gamma}\sum_{t=1}^{T}\sum_{n=1}^{N}w_t^n\big((Q_t^n)^2-(Q_{t+1}^n)^2\big)\\ &=\frac{1}{2\gamma}\sum_{t=1}^{T}\sum_{n=1}^{N}\big(w_t^n(Q_t^n)^2-w_{t+1}^n(Q_{t+1}^n)^2\big)\\ &\quad+\frac{1}{2\gamma}\sum_{t=1}^{T}\sum_{n=1}^{N}(w_{t+1}^n-w_t^n)(Q_{t+1}^n)^2\\ &\overset{(a)}{\le}\frac{1}{2\gamma}\sum_{n=1}^{N}\big(w_1^n(Q_1^n)^2-w_{T+1}^n(Q_{T+1}^n)^2\big) \end{aligned}$$

$$+\frac{1}{2\gamma}\sum_{t=1}^{T}\sum_{n=1}^{N}(w_{t+1}^n-w_t^n)\Big(\frac{\eta G}{\gamma}\Big)^2\overset{(b)}{\leq}\frac{\eta^2G^2\Omega_T}{2\gamma^3} \quad (38)$$

where (a) follows from the bound on Q_t^n in (22); and (b) follows from $Q_1^n=0$, $Q_t^n\geq 0,\forall t$, and the definition of Ω_T.

From the definition of ψ_t^n, we have

$$\begin{aligned}\sum_{n=1}^{N}w_t^n\psi_t^n&=\sum_{n=1}^{N}w_t^n\big(\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_{t+1}\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2\big)\\&\overset{(a)}{=}\sum_{n=1}^{N}w_t^n\Big(\Big\|\sum_{m=1}^{N}w_t^m(\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^m)\Big\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2\Big)\\&\overset{(b)}{\leq}\sum_{n=1}^{N}w_t^n\Big(\sum_{m=1}^{N}\big(w_t^m\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^m\|^2\big)-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2\Big)\\&\overset{(c)}{=}\sum_{m=1}^{N}\big(w_t^m\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^m\|^2\big)-\sum_{n=1}^{N}\big(w_t^n\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2\big)\\&=0.\end{aligned} \quad (39)$$

where (a) follows from the definition of the global decision $\mathbf{x}_{t+1}$ in (6), (b) is because of the the separate convexity of the Euclidean norm, and (c) follows from $\sum_{n=1}^{N}w_t^n=1,\forall t$.

Substituting (38) and (39) into (37), and noting that

$$\begin{aligned}\sum_{t=1}^{T}\phi_t&=\sum_{t=1}^{T}\big(\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t\|^2-\|\mathbf{x}_{t+1}^{\text{ctr}}-\mathbf{x}_{t+1}\|^2\big)\\&=\|\mathbf{x}_1^{\text{ctr}}-\mathbf{x}_1\|^2-\|\mathbf{x}_{T+1}^{\text{ctr}}-\mathbf{x}_{T+1}\|^2\leq R^2\end{aligned} \quad (40)$$

we have (36). ∎

D. Bound on the Constraint Violation

We now proceed to provide an upper bound on the constraint violation for ODOTS. The virtual-queue-based online optimization algorithms [45], [53]-[58] bound the constraint violation via the virtual queue bound, which requires Slater's condition (or its relaxed version in [55]). Instead, we resort to bound the constraint violation by properly setting the virtual penalty factor U in the modified Lyapunov drift Θ_t^n (23).

Theorem 2. Under Assumptions 1 and 2, the constraint violation yielded by ODOTS is upper bounded by

$$\begin{aligned}&\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)\\&\leq\Big(\frac{2\gamma^2T+2}{\gamma\eta^2}\Big)^{\frac{1}{2}}\Big(\frac{D^2T}{4\alpha}+2\gamma\eta^2G^2T+D(R+\delta)T\\&\quad+\alpha\big(R^2(1+\Xi_T)+\Lambda_{2,T}+2R(\Lambda_T+\Pi_T)\big)\Big)^{\frac{1}{2}}\end{aligned} \quad (41)$$

where $\Xi_T\triangleq\sum_{t=1}^{T}\sum_{n=1}^{N}(w_t^n-\frac{1}{N})$.

Proof: Summing (30) over n and t, and dividing both sides of the resulting inequality by N, we have

$$\begin{aligned}\frac{U\eta}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)&\leq\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}\big(f_t^n(\mathbf{x}_t^{\text{ctr}})-f_t^n(\hat{\mathbf{x}}_t)\big)\\&+\frac{D^2T}{4\alpha}+2\gamma\eta^2G^2T+\frac{\gamma T}{2}U^2-\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}\Theta_t^n\\&+\alpha\sum_{t=1}^{T}\Big(\phi_t+\sum_{n=1}^{N}\frac{\psi_t^n}{N}\Big)+\alpha\big(\Lambda_{2,T}+2R(\Lambda_T+\Pi_T)\big).\end{aligned} \quad (42)$$

We now bound the terms on the RHS of (42). We have

$$\begin{aligned}f_t^n(\mathbf{x}_t^{\text{ctr}})-f_t^n(\hat{\mathbf{x}}_t)&\overset{(a)}{\leq}\langle\nabla f_t^n(\mathbf{x}_t^{\text{ctr}}),\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t-\mathbf{n}_t\rangle\\&\overset{(b)}{\leq}\|\nabla f_t^n(\mathbf{x}_t^{\text{ctr}})\|(\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t\|+\|\mathbf{n}_t\|)\leq D(R+\delta)\end{aligned} \quad (43)$$

where (a) follows from the convexity of $f_t^n(\mathbf{x})$ in (13) and the definition of the noisy global decision $\hat{\mathbf{x}}_t$ in (6), (b) is because $\langle\mathbf{a},\mathbf{b}\rangle\leq\|\mathbf{a}\|\|\mathbf{b}\|$ and $\|\mathbf{a}+\mathbf{b}\|\leq\|\mathbf{a}\|+\|\mathbf{b}\|$, and (c) follows from the bounds on $\nabla f_t^n(\mathbf{x})$, $\mathcal{X}$, $\|\mathbf{n}_t\|$ in (14), (15), (16).

Similar to the proof of (39), we can show that

$$\begin{aligned}&\sum_{t=1}^{T}\sum_{n=1}^{N}\frac{\psi_t^n}{N}\\&\leq\sum_{t=1}^{T}\sum_{n=1}^{N}\frac{1}{N}\Big(\Big\|\sum_{m=1}^{N}w_t^m(\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^m)\Big\|^2-\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2\Big)\\&\leq\sum_{t=1}^{T}\sum_{n=1}^{N}\big(w_t^n-\frac{1}{N}\big)\|\mathbf{x}_t^{\text{ctr}}-\mathbf{x}_t^n\|^2\overset{(a)}{\leq}R^2\Xi_T.\end{aligned} \quad (44)$$

where (a) follows from the bound on $\mathcal{X}$ in (15) and the definition of Ξ_T.

Also, noting that $Q_1^n=0$ by initialization, we have

$$\begin{aligned}-\sum_{t=1}^{T}\Theta_t^n&=\frac{1}{2\gamma}\sum_{t=1}^{T}\big((Q_t^n-U)^2-(Q_{t+1}^n-U)^2\big)\\&=\frac{1}{2\gamma}\big((Q_1^n-U)^2-(Q_{T+1}^n-U)^2\big)\\&\leq\frac{1}{2\gamma}(Q_1^n-U)^2\leq\frac{U^2}{2\gamma}.\end{aligned} \quad (45)$$

Substituting (40) and (43)-(45) into (42), and rearranging terms, we have

$$\begin{aligned}&\frac{U\eta}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)-\frac{\gamma T}{2}U^2-\frac{1}{2\gamma}U^2\\&\leq\frac{D^2T}{4\alpha}+2\gamma\eta^2G^2T+D(R+\delta)T\\&\quad+\alpha\big(R^2(1+\Xi_T)+\Lambda_{2,T}+2R(\Lambda_T+\Pi_T)\big).\end{aligned} \quad (46)$$

Consider the case $\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)\geq 0$. Set

$$U=\frac{\gamma\eta}{\gamma^2T+1}\Big[\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)\Big]_+ \quad (47)$$

and substitute it into the LHS of (46), we have

$$\begin{aligned}&\frac{\gamma\eta^2}{2\gamma^2T+2}\Big[\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)\Big]_+^2\\&\leq\frac{D^2T}{4\alpha}+2\gamma\eta^2G^2T+D(R+\delta)T\\&\quad+\alpha\big(R^2(1+\Xi_T)+\Lambda_{2,T}+2R(\Lambda_T+\Pi_T)\big).\end{aligned} \quad (48)$$

Taking the square root on both side of the above inequality, we have (41). Further note that for the case $\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)<0$, (41) readily holds, we complete the proof. ∎

E. Discussion on the Performance Bounds

We now discuss the sufficient conditions for ODOTS to yield sublinear performance gap and constraint violation. We define parameters $\mu \in [0,1]$ and $\nu \in [0,1]$ to represent the time variability of the underlying system, such that $\max\{\Pi_T, \Xi_T, \Lambda_{2,T}, \Lambda_T\} = O(T^{\mu})$, and $\Omega_T = \mathcal{O}(T^{\nu})$. Note that Ξ_T and Ω_T are the accumulated variation measures of the time-varying weights $\{w_t^n\}$ on the devices (see Theorems 1 and 2 for definition). An important special case is $w_t^n = \frac{1}{N}, \forall n, \forall t$, *i.e.,* the devices have time-invariant equal weights. From Theorems 1 and 2, we can derive the following corollary regarding the performance gap and constraint violation bounds, depending on whether w_t^n is time-varying.

Corollary 1 (Convex Loss Functions with Single-Step Local Update)**.** Suppose Assumptions 1 and 2 hold.

Time-varying weight: Let $\alpha = T^{\frac{1-\mu}{2}}$, $\gamma = T^{\frac{\nu-1}{4}}$, and $\eta = \mathcal{O}(1)$ in ODOTS. We have

$$\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t) - f_t(\mathbf{x}_t^{\text{ctr}})\big) = \mathcal{O}\big(\max\big\{T^{\frac{1+\mu}{2}}, T^{\frac{3+\nu}{4}}\big\}\big), \quad (49)$$

$$\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) = \mathcal{O}\big(T^{\frac{7+\nu}{8}}\big). \quad (50)$$

Time-invariant equal weight: Suppose $w_t^n = \frac{1}{N}, \forall n, \forall t$ such that $\Xi_T = 0$ and $\Omega_T = 0$. Let $\alpha = T^{\frac{1-\mu}{2}}$, $\gamma = T^{-\frac{1}{2}}$, and $\eta = \mathcal{O}(1)$ in ODOTS. We have

$$\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t) - f_t(\mathbf{x}_t^{\text{ctr}})\big) = \mathcal{O}\big(T^{\frac{1+\mu}{2}}\big), \quad (51)$$

$$\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) = \mathcal{O}\big(T^{\frac{3}{4}}\big). \quad (52)$$

Proof: See Appendix A in the supplementary materials.

In particular, if $\mu < 1$ and $\nu < 1$, *i.e.,* the system variations are sublinear in T, both the performance gap and constraint violation are sublinear in T. Therefore, as T approaches infinity, both the time-averaged performance gap and the constraint violation are guaranteed to converge to zero. We remark here that sublinear system variations is a standard necessary condition (but generally insufficient) for sublinear performance bounds in online optimization with unpredictable dynamics [43], [53]-[57]. Corollary 1 suggests that ODOTS can closely track the underlying system dynamics to return superior performance regardless whether the system variations are sublinear.

VI. Extension to Multi-Step Local Updates

In Section IV, we have proposed ODOTS and derived its performance bounds in Section V for general convex loss functions. The basic version of ODOTS assumes each local device performs only one-step local gradient descent to update its local decision before the global decision aggregation at the central server. In practical computation-communication systems, where the local devices often have high computational capacities while the communication resources are limited, it is beneficial for the local devices to perform multi-step local gradient descent before communicating their local decisions to the central server. In this section, we propose a variation of ODOTS, termed Online Distributed Optimization with Temporal Similarity and Multi-step Local Updates (ODOTS-MLU), to enable multi-step local gradient descent at the local devices for more efficient communication. We further show that ODOTS-MLU provides improved performance bounds for strongly convex loss functions.

A. ODOTS-MLU Algorithm

In the following, we configure ODOTS to enable $M+1$ steps of gradient descent at the local devices. At each time t, after receiving the noisy global decision $\hat{\mathbf{x}}_t$ from the central server, each local device n initializes an auxiliary decision $\tilde{\mathbf{x}}_t^{n,0} = \hat{\mathbf{x}}_t$. Then, each local device n first performs M-step local gradient descent to generate $\tilde{\mathbf{x}}_t^{n,M}$.[7] Specifically, each local device n updates $\tilde{\mathbf{x}}_t^{n,m}, \forall m \in \{1,\ldots,M\}$ by solving the following optimization problem:

$$\mathbf{P3}^{n,m}: \quad \min_{\mathbf{x}\in\mathcal{X}} \quad \langle \nabla f_t^n(\tilde{\mathbf{x}}_t^{n,m-1}), \mathbf{x} - \tilde{\mathbf{x}}_t^{n,m-1}\rangle + \alpha\|\mathbf{x} - \tilde{\mathbf{x}}_t^{n,m-1}\|^2.$$

The optimal solution to $\mathbf{P3}^{n,m}$ is to perform the standard projected local gradient descent update, given by

$$\tilde{\mathbf{x}}_t^{n,m} = \Big[\tilde{\mathbf{x}}_t^{n,m-1} - \frac{1}{2\alpha}\nabla f_t^n(\tilde{\mathbf{x}}_t^{n,m-1})\Big]_{-x_{\max}\mathbf{1}}^{x_{\max}\mathbf{1}} \quad (53)$$

where $\alpha > 0$ is the step-size parameter.

The local decision $\tilde{\mathbf{x}}_t^{n,M}$ obtained after performing M-step local gradient descent in (53) may drift from the previous quantized local decision $\hat{\mathbf{x}}_{t-1}^n$, since $\mathbf{P3}^{n,m}$ is not subject to the decision dis-similarity constraint (7). Similar to $\mathbf{P2}^n$, we use the tunable virtual queue to control the constraint violation. Specifically, after obtaining the *intermediate* local decision $\tilde{\mathbf{x}}_t^{n,M}$, we solve the following optimization problem to update the *final* local decision $\mathbf{x}_t^n$ for time t:

$$\mathbf{P4}^n: \quad \min_{\mathbf{x}\in\mathcal{X}} \quad \langle \nabla f_t^n(\tilde{\mathbf{x}}_t^{n,M}), \mathbf{x} - \tilde{\mathbf{x}}_t^{n,M}\rangle + \alpha\|\mathbf{x} - \tilde{\mathbf{x}}_t^{n,M}\|^2 + \eta Q_t^n g_t^n(\mathbf{x})$$

where $\eta > 0$ is another algorithm parameter and Q_t^n is the tunable virtual queue (10) with tuning parameter $\gamma \in (0,1)$. Compared with $\mathbf{P2}^n$ in the basic form of ODOTS, we replace the noisy global decision $\hat{\mathbf{x}}_t$ with the local decision $\tilde{\mathbf{x}}_t^{n,M}$ after performing the additional M-step local gradient descent to fully utilize the computational capacity at the local devices. Correspondingly, the optimal solution to $\mathbf{P4}^n$ is also in closed form, given by (11) with $\hat{\mathbf{x}}_t$ replaced by $\tilde{\mathbf{x}}_t^{n,M}$. We will show analytically that multi-step local gradient descent will improve the performance bounds by ODOTS for strongly convex loss functions in Section VI-C.

We summarize the devices' algorithm in Algorithm 3. The server uses the same Algorithm 2 as the basic form of ODOTS.

[7] We can easily extend this to allow the local devices perform different steps of local gradient descent. In this case, we define M as the minimum number of gradient descent steps among the devices, and all of our subsequent analysis results hold. Also note that a local device may choose $M = 0$, so that $\tilde{\mathbf{x}}_t^{n,M} = \hat{\mathbf{x}}_t$.

Algorithm 3 ODOTS-MLU: Device n's algorithm

Initialize $\hat{\mathbf{x}}_1 = \mathbf{0}$ and $Q_1^n = 0$. For each t, do:
  Initialize auxiliary local decision $\tilde{\mathbf{x}}_t^{n,0} = \hat{\mathbf{x}}_t$.
  for $m = 1$ **to** M
    Update $\tilde{\mathbf{x}}_t^{n,m}$ by solving $\mathbf{P3}^{n,m}$ via (53).
  end for
  Update local decision $\mathbf{x}_t^n$ by solving $\mathbf{P4}^n$.
  Update local virtual queue Q_{t+1}^n via (10).
  Update quantized local decision $\hat{\mathbf{x}}_t^n$ via (4).
  Transmit $\hat{\mathbf{x}}_t^n$ via conditional entropy coding.

Next, we will derive the bounds on the optimality gap and constraint violation for ODOTS-MLU, which will also give guidelines on the choices of algorithm parameters α, η, and γ.

B. Performance Bounds of ODOTS-MLU

For the analysis of ODOTS-MLU, we require the following assumptions of strong convexity and smoothness on the local loss function.

Assumption 3. The local loss function $f_t^n(\mathbf{x})$ satisfies the following conditions:

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

$$f_t^n(\mathbf{y}) \geq f_t^n(\mathbf{x}) + \langle \nabla f_t^n(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \varrho \|\mathbf{y} - \mathbf{x}\|^2. \quad (54)$$

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

$$f_t^n(\mathbf{y}) \leq f_t^n(\mathbf{x}) + \langle \nabla f_t^n(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + L \|\mathbf{y} - \mathbf{x}\|^2. \quad (55)$$

In many machine learning, system control, and signal processing applications, *e.g.,* support vector machine, softmax classification, and subspace tracking, the loss functions are strongly convex. Furthermore, for broad applications with convex loss functions, adding a regularization term $\varrho\|\mathbf{x}\|^2$ makes the objective function strongly convex without causing much impact on the actual system performance [60].

Remark 2. For unbounded feasible set $\mathcal{X}$, the strong-convexity and bounded gradient assumptions are contradictory [61]. However, we consider *bounded* feasible set $\mathcal{X}$ in **P1**, *i.e.,* $\|\mathbf{x} - \mathbf{y}\| \leq R, \forall \mathbf{x}, \mathbf{y} \in \mathcal{X}$ in (15), such that there exists a constant gradient upper bound $D \geq 4\varrho^2 R^2$ for the two assumptions to hold.

1) Bound on the Performance Gap: Using properties of strongly convex and smooth functions and the results in Lemmas 1-3 and 5, the following lemma bounds the difference between the local decision $\mathbf{x}_t^n$ by ODOTS-MLU and the centralized per-slot optimal decision $\mathbf{x}_t^{\text{ctr}}$.

Lemma 6. For any local gradient descent steps $M > 0$, if $\alpha \geq L$, the difference between the local decision $\mathbf{x}_t^n$ yielded by ODOTS-MLU and the centralized per-slot optimal decision $\mathbf{x}_t^{\text{ctr}}$ is upper bounded by

$$\|\mathbf{x}_t^n - \mathbf{x}_t^{\text{ctr}}\|^2 \leq \rho^M \|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\text{ctr}}\|^2 + \frac{D^2}{2\alpha^2} - \frac{\Theta_t^n}{\alpha} + \frac{2\gamma\eta^2 G^2}{\alpha} + 3\big(\Psi_t^2 + (\Phi_t^n)^2\big) + (4R + 2\delta)\big(\Psi_t + \Phi_t^n\big) \quad (56)$$

where $\rho \triangleq \frac{\alpha - \varrho}{\alpha + \varrho} < 1$, $\Psi_t \triangleq \|\mathbf{x}_t^{\text{ctr}} - \mathbf{x}_t^\star\|$, and $\Phi_t^n \triangleq \|\mathbf{x}_t^{n\star} - \mathbf{x}_t^\star\|$, with $\mathbf{x}_t^\star \in \arg\min_{\mathbf{x} \in \mathcal{X}} f_t(\mathbf{x})$ and $\mathbf{x}_t^{n\star} \in \arg\min_{\mathbf{x} \in \mathcal{X}} f_t^n(\mathbf{x})$ being an optimal global and local decisions under the short-term constraints, respectively.

Proof: The objective function of $\mathbf{P4}^n$ is 2α-strongly convex over $\mathcal{X}$ w.r.t. $\|\cdot\|$ due to the regularization terms $\alpha\|\mathbf{x} - \tilde{\mathbf{x}}_t^{n,M}\|^2$. Since $\mathbf{x}_t^n$ is the optimal solution to $\mathbf{P4}^n$, from the result in Lemma 5, we have

$$\begin{aligned}&\langle \nabla f_t^n(\tilde{\mathbf{x}}_t^{n,M}), \mathbf{x}_t^n - \tilde{\mathbf{x}}_t^{n,M} \rangle + \alpha\|\mathbf{x}_t^n - \tilde{\mathbf{x}}_t^{n,M}\|^2 + \eta Q_t^n g_t^n(\mathbf{x}_t^n) \\ &\quad \leq \langle \nabla f_t^n(\tilde{\mathbf{x}}_t^{n,M}), \mathbf{x}_t^{\text{ctr}} - \tilde{\mathbf{x}}_t^{n,M} \rangle + \eta Q_t^n g_t^n(\mathbf{x}_t^{\text{ctr}}) \\ &\qquad + \alpha\big(\|\mathbf{x}_t^{\text{ctr}} - \tilde{\mathbf{x}}_t^{n,M}\|^2 - \|\mathbf{x}_t^n - \mathbf{x}_t^{\text{ctr}}\|^2\big). \end{aligned} \quad (57)$$

Since $f_t^n(\mathbf{x})$ is $2L$-smooth over $\mathcal{X}$, from (55), we have

$$\begin{aligned}&f_t^n(\mathbf{x}_t^n) - f_t^n(\tilde{\mathbf{x}}_t^{n,M}) - L\|\mathbf{x}_t^n - \tilde{\mathbf{x}}_t^{n,M}\|^2 \\ &\quad \leq \langle \nabla f_t^n(\tilde{\mathbf{x}}_t^{n,M}), \mathbf{x}_t^n - \tilde{\mathbf{x}}_t^{n,M} \rangle. \end{aligned} \quad (58)$$

Since $f_t^n(\mathbf{x})$ is convex over $\mathcal{X}$, from (13), we have

$$\langle \nabla f_t^n(\tilde{\mathbf{x}}_t^{n,M}), \mathbf{x}_t^{\text{ctr}} - \tilde{\mathbf{x}}_t^{n,M} \rangle \leq f_t^n(\mathbf{x}_t^{\text{ctr}}) - f_t^n(\tilde{\mathbf{x}}_t^{n,M}). \quad (59)$$

Note that to use the strong-convexity of $f_t^n(\mathbf{x})$ here would introduce an additional $-\varrho\|\tilde{\mathbf{x}}_t^{n,M} - \mathbf{x}_t^{\text{ctr}}\|^2$ term on the RHS of (59). However, it is the term $\|\tilde{\mathbf{x}}_t^{n,M} - \mathbf{x}_t^{\text{ctr}}\|^2$, instead of the constant (α or $\alpha - \varrho$) in front of it, that limits the performance bounds. Instead, strong-convexity is used later to relate $\|\tilde{\mathbf{x}}_t^{n,M} - \mathbf{x}_t^{\text{ctr}}\|^2$ to $\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\text{ctr}}\|^2$ in (65). Substituting (58) and (59) into the LHS and RHS of (57), respectively, we have

$$\begin{aligned}&f_t^n(\mathbf{x}_t^n) - f_t^n(\tilde{\mathbf{x}}_t^{n,M}) + (\alpha - L)\|\mathbf{x}_t^n - \tilde{\mathbf{x}}_t^{n,M}\|^2 + \eta Q_t^n g_t^n(\mathbf{x}_t^n) \\ &\quad \leq f_t^n(\mathbf{x}_t^{\text{ctr}}) - f_t^n(\tilde{\mathbf{x}}_t^{n,M}) + \eta Q_t^n g_t^n(\mathbf{x}_t^{\text{ctr}}) \\ &\qquad + \alpha\big(\|\mathbf{x}_t^{\text{ctr}} - \tilde{\mathbf{x}}_t^{n,M}\|^2 - \|\mathbf{x}_t^n - \mathbf{x}_t^{\text{ctr}}\|^2\big). \end{aligned} \quad (60)$$

Rearranging terms of (60) and noting that $\eta Q_t^n g_t^n(\mathbf{x}_t^{\text{ctr}}) \leq 0$ and $\alpha \geq L$, we have

$$\begin{aligned}\alpha\|\mathbf{x}_t^n - \mathbf{x}_t^{\text{ctr}}\|^2 &\leq f_t^n(\mathbf{x}_t^{\text{ctr}}) - f_t^n(\mathbf{x}_t^n) - \eta Q_t^n g_t^n(\mathbf{x}_t^n) \\ &\quad + \alpha\|\tilde{\mathbf{x}}_t^{n,M} - \mathbf{x}_t^{\text{ctr}}\|^2. \end{aligned} \quad (61)$$

We now bound the RHS of (61). We have

$$\begin{aligned}f_t^n(\mathbf{x}_t^{\text{ctr}}) - f_t^n(\mathbf{x}_t^n) &\overset{(a)}{\leq} f_t^n(\mathbf{x}_t^{\text{ctr}}) - f_t^n(\mathbf{x}_t^{n\star}) \\ &\overset{(b)}{\leq} \langle \nabla f_t^n(\mathbf{x}_t^{\text{ctr}}), \mathbf{x}_t^{\text{ctr}} - \mathbf{x}_t^{n\star} \rangle \\ &\overset{(c)}{\leq} \frac{\|\nabla f_t^n(\mathbf{x}_t^{\text{ctr}})\|^2}{2\alpha} + \frac{\alpha}{2}\|\mathbf{x}_t^{\text{ctr}} - \mathbf{x}_t^{n\star}\|^2 \\ &\overset{(d)}{\leq} \frac{D^2}{2\alpha} + \frac{\alpha}{2}\|\mathbf{x}_t^{\text{ctr}} - \mathbf{x}_t^{n\star}\|^2 \overset{(e)}{\leq} \frac{D^2}{2\alpha} + \alpha\Psi_t^2 + \alpha(\Phi_t^n)^2 \end{aligned} \quad (62)$$

where (a) follows from $\mathbf{x}_t^{n\star}$ being an optimal local decision for minimizing $f_t^n(\mathbf{x})$ over $\mathcal{X}$, (b) is because of the convexity of $f_t^n(\mathbf{x})$ in (13), (c) follows from $\langle \mathbf{a}, \mathbf{b} \rangle \leq \frac{1}{2\alpha}\|\mathbf{a}\|^2 + \frac{\alpha}{2}\|\mathbf{b}\|^2, \forall \alpha > 0$, (d) is because the gradient $\nabla f_t^n(\mathbf{x})$ being bounded in (14), and (e) follows from $\|\mathbf{a} + \mathbf{b}\|^2 \leq 2\|\mathbf{a}\|^2 + 2\|\mathbf{b}\|^2$ and the definitions of Ψ_t and Φ_t^n.

To bound the last term on the RHS of (61), we require the following property of a 2ϱ-strongly convex and $2L$-smooth function, which is shown in Lemma 1 in [62].

Lemma 7. ([62, 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 $\mathcal{Z}$ as defined in (54) and (55). Let $\mathbf{v} = \arg\min_{\mathbf{z}\in\mathcal{Z}}\{\langle \nabla h(\mathbf{u}), \mathbf{z}-\mathbf{u}\rangle + \alpha\|\mathbf{z}-\mathbf{u}\|^2\}$ and $\mathbf{w} = \arg\min_{\mathbf{z}\in\mathcal{Z}} h(\mathbf{z})$. Then, for any $\alpha \geq L$ and any $\mathbf{u} \in \mathcal{Z}$, we have $\|\mathbf{w}-\mathbf{v}\|^2 \leq \frac{\alpha-\varrho}{\alpha+\varrho}\|\mathbf{w}-\mathbf{u}\|^2$.

Applying the result in Lemma 7 to the update of $\tilde{\mathbf{x}}_t^{n,m}$ in (53), which is the optimal solution to $\mathbf{P3}^{n,m}$, for any $m \in \{1,\ldots,M\}$ and any $\alpha \geq L$, we have

$$\|\tilde{\mathbf{x}}_t^{n,m} - \mathbf{x}_t^{n\star}\|^2 \leq \rho\|\tilde{\mathbf{x}}_t^{n,m-1} - \mathbf{x}_t^{n\star}\|^2 \tag{63}$$

where $\rho \triangleq \frac{\alpha-\varrho}{\alpha+\varrho} < 1$. Combining the above M inequalities and noting that $\tilde{\mathbf{x}}_t^{n,0} = \hat{\mathbf{x}}_t$, we have

$$\|\tilde{\mathbf{x}}_t^{n,M} - \mathbf{x}_t^{n\star}\|^2 \leq \rho^M\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{n\star}\|^2. \tag{64}$$

Also, we have

$$\begin{aligned}
&\|\tilde{\mathbf{x}}_t^{n,M} - \mathbf{x}_t^{\text{ctr}}\|^2 \\
&\leq \|\tilde{\mathbf{x}}_t^{n,M} - \mathbf{x}_t^{\star}\|^2 + \Psi_t^2 + 2R\Psi_t \\
&\leq \|\tilde{\mathbf{x}}_t^{n,M} - \mathbf{x}_t^{n\star}\|^2 + (\Phi_t^n)^2 + \Psi_t^2 + 2R(\Phi_t^n + \Psi_t) \\
&\overset{(a)}{\leq} \rho^M\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{n\star}\|^2 + (\Phi_t^n)^2 + \Psi_t^2 + 2R(\Phi_t^n + \Psi_t) \\
&\overset{(b)}{\leq} \rho^M\big(\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\star}\|^2 + (\Phi_t^n)^2 + 2\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\star}\|\Phi_t^n\big) \\
&\quad + (\Phi_t^n)^2 + \Psi_t^2 + 2R(\Phi_t^n + \Psi_t) \\
&\overset{(c)}{\leq} \rho^M\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\star}\|^2 + 2(\Phi_t^n)^2 + \Psi_t^2 \\
&\quad + (4R+2\delta)\Phi_t^n + 2R\Psi_t \\
&\overset{(d)}{\leq} \rho^M\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\text{ctr}}\|^2 + 2\big((\Phi_t^n)^2 + \Psi_t^2\big) \\
&\quad + (4R+2\delta)(\Phi_t^n + \Psi_t).
\end{aligned} \tag{65}$$

where (a) follows from (64); (b) is due to $\|\mathbf{a}+\mathbf{b}\|^2 \leq \|\mathbf{a}\|^2 + \|\mathbf{b}\|^2 + 2\|\mathbf{a}\|\|\mathbf{b}\|$ and the definition of Φ_t^n; (c) follows from $\rho < 1$, and the bounds on $\mathcal{X}$ and $\mathbf{n}_t$ in (15) and (16) such that $\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\star}\| = \|\mathbf{x}_t - \mathbf{x}_t^{\star} + \mathbf{n}_t\| \leq R + \delta$; and (d) can be proven similar to (b).

Substituting (62), the bound on the modified Lyapunov drift in (24) with $U = 0$, and (65) into (61), we have

$$\begin{aligned}
\alpha\|\mathbf{x}_t^n - \mathbf{x}_t^{\text{ctr}}\|^2 \leq{}& \alpha\rho^M\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\text{ctr}}\|^2 + \frac{D^2}{2\alpha} - \Theta_t^n + 2\gamma\eta^2G^2 \\
&+ 3\alpha\big((\Phi_t^n)^2 + \Psi_t^2\big) + \alpha(4R+2\delta)(\Phi_t^n + \Psi_t).
\end{aligned} \tag{66}$$

Dividing both sizes of (66) by α, we obtain (56). ∎

Based on the result in Lemma 6, we provide an upper bound on the performance gap to the centralized per-slot optimal solution sequence $\{\mathbf{x}_t^{\text{ctr}}\}$ for ODOTS-MLU in the following theorem.

Theorem 3. Under Assumptions 2 and 3, if we choose $\alpha \geq L$, then for any $D \geq 4\varrho^2R^2$, $M > 0$ and $\xi > 0$, the performance gap to $\{\mathbf{x}_t^{\text{ctr}}\}$ by ODOTS-MLU is upper bounded by

$$\begin{aligned}
\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t) - f_t(\mathbf{x}_t^{\text{ctr}})\big) \leq{}& \frac{\Pi_\nabla}{4\xi} + \frac{L+\xi}{1-\rho^M}\Big(\frac{D^2T}{2\alpha^2} + \frac{\eta^2G^2\Omega_T}{2\alpha\gamma^3} \\
&+ \frac{2\gamma\eta^2G^2T}{\alpha} + \Lambda_{2,T} + 2R\Lambda_T + \Pi_{2,T} + 2(R+\delta)\Pi_T \\
&+ 3(\Delta_{2,\mathbf{x}} + \Pi_{2,\mathbf{x}}) + (4R+2\delta)(\Delta_{\mathbf{x}} + \Pi_{\text{x}})\Big)
\end{aligned} \tag{67}$$

where $\Pi_\nabla \triangleq \sum_{t=1}^{T}\sum_{n=1}^{N} w_t^n\|\nabla f_t^n(\mathbf{x}_t^{\text{ctr}})\|^2$, $\Pi_{2,T} \triangleq \sum_{t=1}^{T}\pi_t^2$, $\Pi_{2,\mathbf{x}} \triangleq \sum_{t=1}^{T}\Psi_t^2$, $\Pi_{\mathbf{x}} \triangleq \sum_{t=1}^{T}\Psi_t$, $\Delta_{2,\mathbf{x}} \triangleq \sum_{t=1}^{T}\sum_{n=1}^{N} w_t^n(\Phi_t^n)^2$, and $\Delta_{\mathbf{x}} \triangleq \sum_{t=1}^{T}\sum_{n=1}^{N} w_t^n\Phi_t^n$.

Proof: For any n and t, we have

$$\begin{aligned}
f_t^n(\hat{\mathbf{x}}_t) - f_t^n(\mathbf{x}_t^{\text{ctr}}) &\overset{(a)}{\leq} \langle\nabla f_t^n(\mathbf{x}_t^{\text{ctr}}), \hat{\mathbf{x}}_t - \mathbf{x}_t^{\text{ctr}}\rangle + L\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\text{ctr}}\|^2 \\
&\overset{(b)}{\leq} \frac{1}{4\xi}\|\nabla f_t^n(\mathbf{x}_t^{\text{ctr}})\|^2 + (L+\xi)\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\text{ctr}}\|^2
\end{aligned} \tag{68}$$

where (a) follows from the property of smooth function $f_t^n(\mathbf{x})$ in (55), and (b) is because $\langle\mathbf{a},\mathbf{b}\rangle \leq \frac{1}{4\xi}\|\mathbf{a}\|^2 + \xi\|\mathbf{b}\|^2, \forall\xi > 0$. Multiplying both sides of (68) by w_t^n, and summing the resulting inequality over n and t, we have

$$\begin{aligned}
\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t) - f_t(\mathbf{x}_t^{\text{ctr}})\big) \leq{}& \frac{1}{4\xi}\sum_{t=1}^{T}\sum_{n=1}^{N} w_t^n\|\nabla f_t^n(\mathbf{x}_t^{\text{ctr}})\|^2 \\
&+ (L+\xi)\sum_{t=1}^{T}\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\text{ctr}}\|^2.
\end{aligned} \tag{69}$$

We now bound the last term on the RHS of (69). We have

$$\begin{aligned}
\|\hat{\mathbf{x}}_{t+1} - \mathbf{x}_{t+1}^{\text{ctr}}\|^2 &= \|\hat{\mathbf{x}}_{t+1} - \mathbf{x}_t^{\text{ctr}} + \mathbf{x}_t^{\text{ctr}} - \mathbf{x}_{t+1}^{\text{ctr}}\|^2 \\
&\overset{(a)}{\leq} \|\hat{\mathbf{x}}_{t+1} - \mathbf{x}_t^{\text{ctr}}\|^2 + \pi_t^2 + 2(R+\delta)\pi_t \\
&\leq \|\mathbf{x}_{t+1} - \mathbf{x}_t^{\text{ctr}}\|^2 + \|\mathbf{n}_{t+1}\|^2 + 2R\|\mathbf{n}_{t+1}\| \\
&\quad + \pi_t^2 + 2(R+\delta)\pi_t \\
&= \Big\|\sum_{n=1}^{N} w_t^n\mathbf{x}_t^n - \mathbf{x}_t^{\text{ctr}}\Big\|^2 + \|\mathbf{n}_{t+1}\|^2 + 2R\|\mathbf{n}_{t+1}\| \\
&\quad + \pi_t^2 + 2(R+\delta)\pi_t \\
&\overset{(b)}{\leq} \sum_{n=1}^{N} w_t^n\|\mathbf{x}_t^n - \mathbf{x}_t^{\text{ctr}}\|^2 + \|\mathbf{n}_{t+1}\|^2 + 2R\|\mathbf{n}_{t+1}\| \\
&\quad + \pi_t^2 + 2(R+\delta)\pi_t
\end{aligned} \tag{70}$$

where (a) follows from $\|\mathbf{a}+\mathbf{b}\|^2 \leq \|\mathbf{a}\|^2 + \|\mathbf{b}\|^2 + 2\|\mathbf{a}\|\|\mathbf{b}\|$, the bounds on $\mathcal{X}$ and $\mathbf{n}_t$ in (15) and (16), and the definition of π_t; and (b) is because of the separate convexity of the Euclidean norm.

Summing (70) over $t = 1,\ldots,T-1$, we have

$$\begin{aligned}
&\sum_{t=1}^{T-1}\|\hat{\mathbf{x}}_{t+1} - \mathbf{x}_{t+1}^{\text{ctr}}\|^2 \\
&\leq \sum_{t=1}^{T-1}\sum_{n=1}^{N} w_t^n\|\mathbf{x}_t^n - \mathbf{x}_t^{\text{ctr}}\|^2 + \sum_{t=1}^{T-1}\|\mathbf{n}_{t+1}\|^2 + 2R\sum_{t=1}^{T-1}\|\mathbf{n}_{t+1}\| \\
&\quad + \sum_{t=1}^{T-1}\pi_t^2 + 2(R+\delta)\sum_{t=1}^{T-1}\pi_t \\
&\leq \sum_{t=1}^{T}\sum_{n=1}^{N} w_t^n\|\mathbf{x}_t^n - \mathbf{x}_t^{\text{ctr}}\|^2 + \Lambda_{2,T} + 2R\Lambda_T \\
&\quad + \Pi_{2,T} + 2(R+\delta)\Pi_T \\
&\overset{(a)}{\leq} \rho^M\sum_{t=1}^{T}\|\hat{\mathbf{x}}_t - \mathbf{x}_t^{\text{ctr}}\|^2 + \frac{D^2T}{2\alpha^2} - \frac{1}{\alpha}\sum_{t=1}^{T}\sum_{n=1}^{N} w_t^n\Theta_t^n
\end{aligned}$$

$$+\frac{2\gamma\eta^2G^2T}{\alpha}+3(\Delta_{2,\mathbf{x}}+\Pi_{2,\mathbf{x}})+(4R+2\delta)(\Delta_{\mathbf{x}}+\Pi_{\mathrm{x}})+\Lambda_{2,T}+2R\Lambda_T+\Pi_{2,T}+2(R+\delta)\Pi_T$$
$$\overset{(b)}{\leq}\rho^M\sum_{t=1}^{T}\|\hat{\mathbf{x}}_t-\mathbf{x}_t^{\mathrm{ctr}}\|^2+\frac{D^2T}{2\alpha^2}+\frac{\eta^2G^2\Omega_T}{2\alpha\gamma^3}+\frac{2\gamma\eta^2G^2T}{\alpha}+3(\Delta_{2,\mathbf{x}}+\Pi_{2,\mathbf{x}})+(4R+2\delta)(\Delta_{\mathbf{x}}+\Pi_{\mathrm{x}})+\Lambda_{2,T}+2R\Lambda_T+\Pi_{2,T}+2(R+\delta)\Pi_T \tag{71}$$

where (a) follows from substituting the bound on $\|\mathbf{x}_t^n-\mathbf{x}_t^{\mathrm{ctr}}\|^2$ in (56) of Lemma 6, and (b) follows from the bound on $-\sum_{t=1}^{T}\sum_{n=1}^{N}w_t^n\Theta_t$ in (38).

Rearranging terms of (71), we have

$$(1-\rho^M)\sum_{t=1}^{T}\|\hat{\mathbf{x}}_t-\mathbf{x}_t^{\mathrm{ctr}}\|^2\leq-\|\hat{\mathbf{x}}_1-\mathbf{x}_1^{\mathrm{ctr}}\|^2+\frac{D^2T}{2\alpha^2}+\frac{\eta^2G^2\Omega_T}{2\alpha\gamma^3}+\frac{2\gamma\eta^2G^2T}{\alpha}+3(\Delta_{2,\mathbf{x}}+\Pi_{2,\mathbf{x}})+(4R+2\delta)(\Delta_{\mathbf{x}}+\Pi_{\mathrm{x}})+\Lambda_{2,T}+2R\Lambda_T+\Pi_{2,T}+2(R+\delta)\Pi_T. \tag{72}$$

Note that $1-\rho^M>0$ since $\rho<1$, divide both sides of (72) by $1-\rho^M$, and then apply it to the RHS of (69), we complete the proof. ∎

2) Bound on the Constraint Violation: We now provide an upper bound on the constraint violation by ODOTS-MLU in the following theorem.

Theorem 4. Under Assumptions 2 and 3, if we choose $\alpha\geq L$, then for any $D\geq 4\varrho^2R^2$ and $M>0$, the constraint violation yielded by ODOTS-MLU is upper bounded by

$$\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)\leq\Big(\frac{2\gamma^2T+2}{\gamma\eta^2}\Big)^{\frac{1}{2}}\Big(\frac{D^2T}{2\alpha}+2\gamma\eta^2G^2T+\alpha\big(\rho^MR^2(1+\Xi_T)+\rho^M\Lambda_{2,T}+2\rho^MR(\Lambda_T+\Pi_T)+3(\Pi_{2,\mathbf{x}}+\Delta'_{2,\mathbf{x}})+(4R+2\delta)(\Pi_{\mathbf{x}}+\Delta'_{\mathbf{x}})\big)\Big)^{\frac{1}{2}}. \tag{73}$$

where $\Delta'_{\mathbf{x}}\triangleq\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}\Phi_t^n$ and $\Delta'_{2,\mathbf{x}}\triangleq\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}(\Phi_t^n)^2$.

Proof: Substituting the bound on the modified Lyapunov drift (24) and (62) into (61), and rearranging terms, we have

$$U\eta g_t^n(\mathbf{x}_t^n)\leq\frac{D^2}{2\alpha}+\alpha\Psi_t^2+\alpha(\Phi_t^n)^2-\Theta_t^n+2\gamma\eta^2G^2+\frac{\gamma}{2}U^2+\alpha\big(\|\tilde{\mathbf{x}}_t^{n,M}-\mathbf{x}_t^{\mathrm{ctr}}\|^2-\|\mathbf{x}_t^n-\mathbf{x}_t^{\mathrm{ctr}}\|^2\big). \tag{74}$$

We now bound the last term on the RHS of (74). We have

$$\|\tilde{\mathbf{x}}_t^{n,M}-\mathbf{x}_t^{\mathrm{ctr}}\|^2-\|\mathbf{x}_t^n-\mathbf{x}_t^{\mathrm{ctr}}\|^2\overset{(a)}{\leq}\rho^M\|\hat{\mathbf{x}}_t-\mathbf{x}_t^{\mathrm{ctr}}\|^2-\|\mathbf{x}_t^n-\mathbf{x}_t^{\mathrm{ctr}}\|^2+2\big((\Phi_t^n)^2+\Psi_t^2\big)+(4R+2\delta)(\Phi_t^n+\Psi_t)\overset{(b)}{\leq}\rho^M\big(\phi_t+\psi_t^n+\|\mathbf{n}_t\|^2+2R(\|\mathbf{n}_t\|+\pi_t)\big)+2\big((\Phi_t^n)^2+\Psi_t^2\big)+(4R+2\delta)(\Phi_t^n+\Psi_t) \tag{75}$$

where (a) follows from (65), and (b) is from (32).

Substituting (75) into (74), summing the resulting inequality over n and t and then dividing both sides by N, we have

$$\frac{U\eta}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)\leq\frac{D^2T}{2\alpha}+2\gamma\eta^2G^2T+\frac{\gamma}{2}U^2T-\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}\Theta_t^n+\alpha\rho^M\sum_{t=1}^{T}\Big(\phi_t+\sum_{n=1}^{N}\frac{\psi_t^n}{N}\Big)+\alpha\rho^M\big(\Lambda_{2,T}+2R(\Lambda_T+\Pi_T)\big)+3\alpha\sum_{t=1}^{T}\Big(\Psi_t^2+\sum_{n=1}^{N}\frac{(\Phi_t^n)^2}{N}\Big)+\alpha(4R+2\delta)\sum_{t=1}^{T}\Big(\Psi_t+\sum_{n=1}^{N}\frac{\Phi_t^n}{N}\Big). \tag{76}$$

Substituting (40), (44), and (45) into (76), and from the definitions of $\Pi_{\mathbf{x}}$, $\Pi_{2,\mathbf{x}}$, $\Delta'_{\mathbf{x}}$, $\Delta'_{2,\mathbf{x}}$, we have

$$\frac{U\eta}{N}\sum_{t=1}^{T}\sum_{n=1}^{N}g_t^n(\mathbf{x}_t^n)-\frac{\gamma T}{2}U^2-\frac{1}{2\gamma}U^2\leq\frac{D^2T}{2\alpha}+2\gamma\eta^2G^2T+\alpha\big(\rho^MR^2(1+\Xi_T)+\rho^M\Lambda_{2,T}+2\rho^MR(\Lambda_T+\Pi_T)+3(\Pi_{2,\mathbf{x}}+\Delta'_{2,\mathbf{x}})+(4R+2\delta)(\Pi_{\mathbf{x}}+\Delta'_{\mathbf{x}})\big). \tag{77}$$

Similar to the proof of (48) in Theorem 2, from (77), we can show that (73) holds. ∎

C. Improved Performance Bounds

We now discuss the sufficient conditions for ODOTS-MLU to yield sublinear performance gap and constraint violation. We again use parameter $\mu\in[0,1]$ to represent the time variability of the underlying system, such that $\max\{\Pi_\nabla,\Delta_{2,\mathbf{x}},\Delta_{\mathbf{x}},\Pi_{2,\mathbf{x}},\Pi_{\mathrm{x}},\Delta'_{2,\mathbf{x}},\Delta'_{\mathbf{x}}\}=\mathcal{O}(T^\mu)$ (see Theorems 3 and 4 for definition). Note that the accumulated squared gradients Π_∇ can be very small [62]. The accumulated difference between the centralized per-slot optimal solution benchmark $\{\mathbf{x}_t^{\mathrm{ctr}}\}$ and the optimal global solution benchmark $\{\mathbf{x}_t^\star\}$ under short-term constraints only can also be small. In particular, if $\|\mathbf{x}_t^{\mathrm{ctr}}-\mathbf{x}_t^\star\|\propto T^{\mu-1}$, we have $\Pi_{2,\mathbf{x}}=\mathcal{O}(T^\mu)$ and $\Pi_{\mathbf{x}}=\mathcal{O}(T^\mu)$. Similarly, if the accumulated difference between the optimal global and local solution benchmarks satisfy $\|\mathbf{x}_t^\star-\mathbf{x}_t^{n,\star}\|\propto T^{\mu-1}$, we have $\Delta_{2,\mathbf{x}}=\mathcal{O}(T^\mu)$ and $\Delta_{\mathbf{x}}=\mathcal{O}(T^\mu)$. Also note that for the important case of time-invariant equal weights, *i.e.,* $w_t^n=\frac{1}{N},\forall n,\forall t$, we have $\Delta'_{2,\mathbf{x}}=\Delta_{2,\mathbf{x}}$ and $\Delta'_{\mathbf{x}}=\Delta_{\mathbf{x}}$.

From Theorems 3 and 4, we can derive the following corollary regarding the performance gap and constraint violation bounds yielded by ODOTS-MLU, depending on whether the local weight w_t^n is time-varying. It is obtained from substituting the corresponding algorithm parameters α, η, and γ into the bounds in (67) and (73).

Corollary 2 (Strongly-Convex Loss Functions with Multi-Step Local Updates)**.** Suppose Assumptions 2 and 3 hold.

Time-varying weight: Let $\alpha=T^{\frac{1-\mu}{2}}+L$, $\gamma=T^{\frac{\nu-1}{4}}$, and $\eta=\mathcal{O}(1)$ in ODOTS-MLU. Then, for any $M>0$, we have

$$\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t)-f_t(\mathbf{x}_t^{\mathrm{ctr}})\big)=\mathcal{O}\big(\max\{T^\mu,T^{\frac{1+2\mu+\nu}{4}}\}\big), \tag{78}$$

$$\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) = \mathcal{O}\big(\max\{T^{\frac{5+2\mu+\nu}{8}}, T^{\frac{3+\nu}{4}}\}\big). \quad (79)$$

Time-invariant equal weight: Suppose $w_t^n = \frac{1}{N}, \forall n, \forall t$ such that $\Xi_T = 0$, $\Omega_T = 0$, $\Delta_{\mathbf{x}} = \Delta'_{\mathbf{x}}$, and $\Delta_{2,\mathbf{x}} = \Delta'_{2,\mathbf{x}}$. Let $\alpha = T^{\frac{1-\mu}{2}} + L$, $\gamma = T^{-\frac{1}{2}}$, and $\eta = \mathcal{O}(1)$ in ODOTS-MLU. Then, for any $M > 0$, we have

$$\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t) - f_t(\mathbf{x}_t^{\text{ctr}})\big) = \mathcal{O}\big(T^{\mu}\big), \quad (80)$$

$$\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) = \mathcal{O}\big(T^{\frac{2+\mu}{4}}\big). \quad (81)$$

Proof: See Appendix B in the supplementary materials.

In particular, if $\mu < 1$ and $\nu < 1$, *i.e.,* the system variations are sublinear in T, both the performance gap and the constraint violation are sublinear in T. Furthermore, in this case it is easy to see that (78), (79), (80), and (81) represent strict improvements over (49), (50), (51), and (52) in Corollary 1. Thus, under additional assumptions on the strong-convexity and smoothness of the loss functions, ODOTS-MLU can reduce both the performance gap and the constraint violation over ODOTS with multiple steps of local update. To the best of our knowledge, no existing literature has considered performing multi-step local gradient descent to improve the performance bounds for distributed online optimization with long-term constraints.

Remark 3. Performing multiple local gradient descent updates can degrade the performance of distributed optimization if the local loss functions are poorly aligned. For example, this has been observed in recent works on FL with heterogeneous data (see [63] and [64]). The performance gap (67) and the constraint violation (73) yielded by ODOTS-MLU quantify this in terms of the accumulated difference between the optimal local decisions $\{\mathbf{x}_t^{n\star}\}$ and the optimal global decisions $\{\mathbf{x}_t^{\star}\}$. The aforementioned performance benefits of ODOTS-MLU over ODOTS requires the condition $\mu < 1$, *i.e.,* the difference between $\mathbf{x}_t^{n\star}$ and $\mathbf{x}_t^{\star}$ vanishes over time. In Section VII-D, we will further show that the performance of ODOTS-MLU in FL improves as the data heterogeneity is reduced.

VII. Application to Federated Learning

As an example to study the performance of ODOTS and ODOTS-MLU in practical systems, we apply them to federated learning (FL) [5], where multiple local devices cooperate to train a machine-learning model with the assistance of a server. We present numerical results to demonstrate the performance advantage of ODOTS over state-of-the-art alternatives, based on canonical image classification datasets for both convex and non-convex loss functions. Furthermore, we show that performing multi-step local updates in ODOTS-MLU can lead to better learning performance and less communication overhead than ODOTS.

A. Simulation Setup

We consider a FL system with $N = 10$ devices and a server. We define each time slot t as one round of FL, which consists of both the computation time and the communication time. We evaluate our results on the popular MNIST dataset [65]. Its training dataset $\mathcal{D}$ consists of 6×10^4 data samples and its test dataset $\mathcal{E}$ has 1×10^4 data samples. Each data sample $(\mathbf{u}, v)$ represents an image with 28×28 pixels and $V = 10$ possible labels, *i.e.,* $\mathbf{u} \in \mathbb{R}^{784}$ and $v \in \{1, \ldots, V\}$. We study the scenario where each local dataset $\mathcal{D}_t^n$ at device n only contains data samples of label n, such that the data is non-i.i.d. We assume device n randomly selects $|\mathcal{D}_t^n| = 20$ data samples at each time t, such that the devices share the same weight $w_t^n = \frac{1}{N}$. We have also conducted experiments on time-varying weights and different datasets, which show a similar trend as the simulation results in this paper. Due to the page limit, we do not include them. This is to emulate the online FL scenario where data samples arrive at the devices over time.

We compare ODOTS with the following schemes.

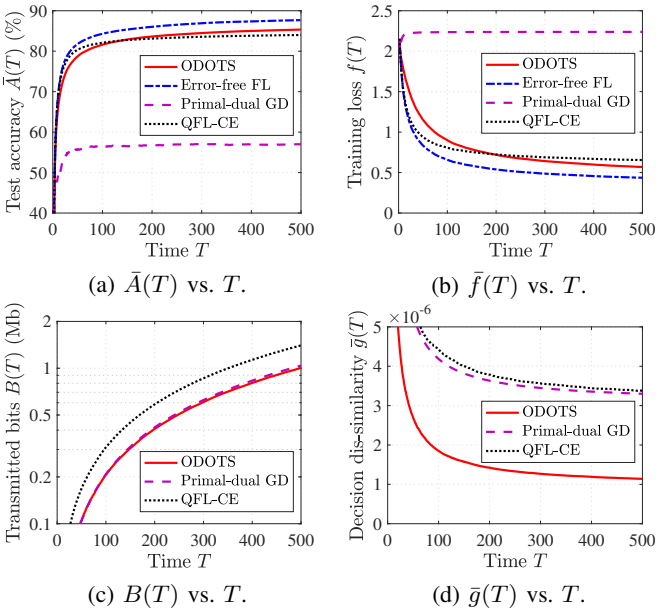
- *Error-free FL:* We alternates local model update $\mathbf{x}_t^n = \mathbf{x}_t - \frac{1}{2\alpha}\nabla f_t^n(\mathbf{x}_t)$ and global model update $\mathbf{x}_{t+1} = \sum_{n=1}^{N} w_t^n \mathbf{x}_t^n$ at each time t. It represents the idealized standard FL algorithm where the communication is error free [5].
- *Primal-dual GD:* The primal-dual gradient descent (GD) algorithm in [43] is the current best solution for distributed constrained online convex optimization with consensus. We implement it to solve **P1**, except using the same current information on the loss and constraint functions as ODOTS.
- *QFL-CE:* We adopt the quantized federated learning (QFL) scheme in [8] by perform local model update (*i.e.,* (11) with $Q_t^n = 0$) and quantization (*i.e.,* (4)) at each time t. We implement the same conditional entropy (CE) coding as ODOTS for QFL.[8] The server then updates its noisy global model (*i.e.,* (6)). This is a state-of-the-art approach where model training and compression are separately designed.

B. Convex Loss: Logistic Regression

We consider the cross-entropy loss for multinomial logistic regression, given by $l(\mathbf{x}; \mathbf{u}, v) = -\sum_{j=1}^{V} 1\{v = j\} \log \frac{\exp(\langle \mathbf{x}[j], \mathbf{u}\rangle)}{\sum_{k=1}^{V}\exp(\langle \mathbf{x}[k], \mathbf{u}\rangle)}$, where $\mathbf{x} = [\mathbf{x}[1]^T, \ldots, \mathbf{x}[V]^T]^T$ with $\mathbf{x}[j] \in \mathbb{R}^{784}$ being the model for label j. The entire model $\mathbf{x}$ is thus of dimension $d = 7840$. Our computation performance metrics are the time-averaged test accuracy $\bar{A}(T) = \frac{1}{|\mathcal{E}|T}\sum_{t=1}^{T}\sum_{i=1}^{|\mathcal{E}|} 1\{\arg\max_j \{\frac{\exp(\langle \hat{\mathbf{x}}_t[j], \mathbf{u}^i\rangle)}{\sum_{k=1}^{V}\exp(\langle \hat{\mathbf{x}}_t[k], \mathbf{u}^i\rangle)}\} = v^i\}$, and the time-averaged training loss $\bar{f}(T) = \frac{1}{T}\sum_{t=1}^{T}\sum_{n=1}^{N} \frac{w_t^n}{|\mathcal{D}_t^n|}\sum_{i=1}^{|\mathcal{D}_t^n|} l(\hat{\mathbf{x}}_t; \mathbf{u}_t^{n,i}, v_t^{n,i})$. Our communication performance metrics are the total number of transmitted bits using the conditional entropy coding $B(T) = \sum_{t=1}^{T}\sum_{n=1}^{N} H(\hat{\mathbf{x}}_t^n | \hat{\mathbf{x}}_{t-1}^n)$ and the time-averaged decision dis-similarity $\bar{g}(T) = \frac{1}{TN}\sum_{t=1}^{T}\sum_{n=1}^{N} \|\mathbf{x}_t^n - \hat{\mathbf{x}}_{t-1}^n\|^2$.[9]

[8] The Elias coding used in [8] does not use any model similarity, and thus incurs more communication overhead compared with the CE coding.

[9] We use the histogram method to estimate the joint probability distribution of $\hat{\mathbf{x}}_t^n$ and $\hat{\mathbf{x}}_{t-1}^n$ and then compute the conditional entropy $H(\hat{\mathbf{x}}_t^n | \hat{\mathbf{x}}_{t-1}^n)$.



(a) $\bar{A}(T)$ vs. T. (b) $\bar{f}(T)$ vs. T.

(c) $B(T)$ vs. T. (d) $\bar{g}(T)$ vs. T.

Fig. 1: Test accuracy $\bar{A}(T)$, training loss $\bar{f}(T)$, transmitted bits $B(T)$, and decision dis-similarity $\bar{g}(T)$ vs. time T.

Fig. 1 shows $\bar{A}(T)$, $\bar{f}(T)$, $B(T)$, and $\bar{g}(T)$ versus T. We set the decision dis-similarity limit $\epsilon = 1e^{-6}$. We set the quantization bit length $b = 5$ for ODOTS and $b = 4$ for Primal-dual GD and QFL-CE. We set the maximum decision limit $x_{\max} = 1 \times 10^{-3}$, step-size $\alpha = 1 \times 10^{5}$, tuning factor $\gamma = 0.5$, and weighting factor $\eta = 5 \times 10^{5}$ in ODOTS. We use the same parameter values for the other schemes if any is used. We note that despite the higher quantization bit length b in ODOTS, due to its inherent communication efficiency, its total number of transmitted bits remains lower than both Primal-dual GD and QFL-CE. We observe that the test accuracy yielded by ODOTS is over 25% higher than Primal-dual GD. This is because Primal-dual GD performs dual gradient descent to control the constraint violation, which can deteriorate its performance when the gradient directions of loss and constraint functions deviate much from each other. Compared with QFL-CE, ODOTS achieves higher test accuracy and incurs abound 30% less communication overhead, thanks to its joint consideration of computation and communication over time. Also, we observe that ODOTS converges slightly slower than QFL-CE at the early training stage, this is because the value of the tunable virtual queue Q_t^n in (10) is relatively large at the beginning to reduce the transmitted bits.

In Fig. 2, we compare the final test accuracy $A(T)$ between ODOTS and QFL-CE under different total transmitted bits $B(T)$. We vary the quantization bit length b in QFL-CE to trade off its computation and communication performance. For ODOTS, we also vary ϵ for any given b value. The final test accuracies yielded by QFL-CE and ODOTS both decrease as b decreases due to the increased quantization errors. However, for any operating point on the QFL-CE curve, we can always find a combination of b and ϵ for ODOTS that achieves higher test accuracy while incurring less communication overhead. Furthermore, their difference in test accuracy grows dramatically as the number of transmitted bits decreases. This suggests

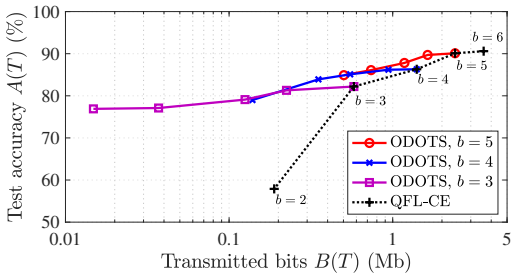


Fig. 2: Final test accuracy $A(T)$ vs. transmitted bits $B(T)$ for convex logistic regression.

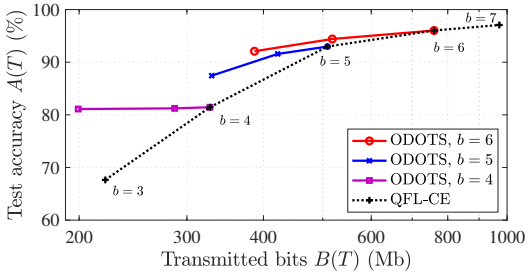


Fig. 3: Final test accuracy $A(T)$ vs. transmitted bits $B(T)$ for non-convex convolutional neural network training.

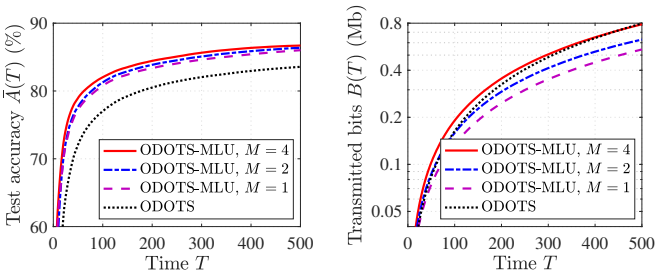
that ODOTS is particularly advantageous in systems with a tight communication budget.

C. Non-Convex Loss: Convolutional Neural Network Training

The performance analysis of ODOTS in Section V requires convex loss functions. To further evaluate the performance of ODOTS for non-convex loss functions, we consider training a convolutional neural network for MNIST classification, with 784 pixels as input, a convolutional layer with 10 filters each of size 9×9, a ReLU hidden layer with 100 neurons, and a softmax output layer with 10 neurons. The total number of model parameters is $d = 101,810$. We set $x_{\max} = 1$, $\alpha = 2$, $\gamma = 0.5$, and $\eta = 0.01$ in ODOTS. Similar to Fig. 2, Fig. 3 compares the performance of ODOTS and QFL-CE in this scenario. Note that the number of transmitted bits is substantially higher due to the larger number of model parameters, compared with the convex logistic regression scenario. We again observe similar trends as in Fig. 2, with ODOTS substantially outperforming QFL-CE especially when the number of transmitted bits is moderate to low.

D. Impact of Multi-Step Local Updates

Fig. 4 shows $\bar{A}(T)$ and $\bar{B}(T)$ versus T on logistic regression for different steps of local updates in ODOTS-MLU with quantization bit length $b = 4$. For this scenario, we uniformly distributed all training data labels among the devices. We set $\epsilon = 1 \times 10^{-6}$. We observe that with only one additional step of local update, the time-averaged test accuracy increases from 83.5% to 86% while the total transmitted bits decreases from 0.8 Mb to 0.54 Mb, thanks to the faster convergence performance of ODOTS-MLU as shown in Corollary 2. As the number of local updates increases, the test accuracy slightly increases at the expense of more transmitted bits. This is



(a) $\bar{A}(T)$ vs. T. (b) $\bar{B}(T)$ vs. T.

Fig. 4: Test accuracy $\bar{A}(T)$ and transmitted bits $\bar{B}(T)$ vs. time T with different steps of local updates M.

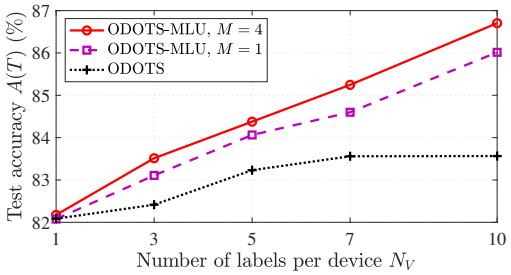


Fig. 5: The impact of heterogeneous data distributions, represented by the number of different data labels N_V each device observes at each time.

because the difference between the updated local decision $\mathbf{x}_t^n$ and the previous quantized local decision $\hat{\mathbf{x}}_{t-1}^n$ increases as the number of local updates M increases, requiring more bits to communicate the quantized decision $\hat{\mathbf{x}}_t^n$ using conditional entropy coding.

In Fig. 5, we study the impact of non-i.i.d. data distribution on multi-step local updates. We change the number of different data labels, denoted by N_V, that each device n observes at each time t to represent the heterogeneity of the data distributions among devices. We observe that as N_V increases, *i.e.,* the data heterogeneity decreases, the learning performance of both ODOTS and ODOTS-MLU improves. Furthermore, the performance gain of ODOTS-MLU over ODOTS is most substantial when $N_V = 10$, *i.e.,* the data distributions among devices are i.i.d. Interestingly, when $N_V = 1$, performing multi-step local updates does not deteriorate the learning performance. This is because the long-term constraint on decision dissimilarity (7) in **P1** prevents the local decision from approaching the local optimal decision even when the data is highly heterogeneous. This serves the same purpose as controlling the *client drift* in [63], [64] for multi-step gradient descent to be effective in FL with heterogeneous data.

VIII. Conclusions

We consider online distributed optimization in networked systems, under a long-term decision dis-similarity constraint to control the communication overhead. We propose efficient ODOTS and ODOTS-MLU algorithms to balance the improvement in optimization and the cost of communication over time via a novel tunable virtual queue. Through a modified Lyapunov drift analysis, we show that both ODOTS and ODOTS-MLU can achieve sublinear performance gap from the centralized per-slot optimizer and sublinear constraint violation simultaneously. With additional steps of local update, ODOTS-MLU can improve over ODOTS in both the performance gap bound and the long-term constraint violation bound. When applying ODOTS and ODOTS-MLU to federated learning, our experimental results demonstrate that ODOTS and ODOTS-MLU can have substantial performance advantage over state-of-the-art approaches, in terms of both improved test accuracy and reduced communication overhead. ODOTS and ODOTS-MLU are advantageous especially in systems with a tight communication budget.

References

[1] J. Wang, B. Liang, M. Dong, G. Boudreau, and A. Afana, "Online distributed optimization for efficient communication via temporal similarity," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2023.

[2] Y. Mao, C. You, J. Zhang, K. Huang, and K. B. Letaief, "A survey on mobile edge computing: The communication perspective," *IEEE Commun. Surveys Tuts.*, vol. 19, pp. 2322–2358, 2017.

[3] Z. Zhou, X. Chen, E. Li, L. Zeng, K. Luo, and J. Zhang, "Edge intelligence: Paving the last mile of artificial intelligence with edge computing," *Proc. IEEE*, vol. 107, pp. 1738–1762, Aug. 2019.

[4] M. I. Jordan, J. D. Lee, and Y. Yang, "Communication-efficient distributed statistical inference," *J. Amer. Stat. Assoc.*, vol. 19, pp. 2322–2358, Feb. 2018.

[5] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. A. Y. Arcas, "Communication-efficient learning of deep networks from decentralized data," in *Proc. Intel. Conf. Artif. Intell. Statist. (AISTATS)*, 2017.

[6] S. Wang *et al.*, "When edge meets learning: Adaptive control for resource-constrained distributed machine learning," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2018.

[7] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu, "1-bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs," in *Proc. Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2014.

[8] D. Alistarh, D. Grubic, J. Li, R. Tomioka, and M. Vojnovic, "QSGD: Communication-efficient SGD via gradient quantization and encoding," in *Proc. Adv. Neural Info. Proc. Sys. (NeurIPS)*, 2017.

[9] J. Wu, W. Huang, J. Huang, and T. Zhang, "Error compensated quantized SGD and its applications to large-scale distributed optimization," in *Proc. Intel. Conf. Mach. Learn. (ICML)*, 2018.

[10] H. Zhang, J. Li, K. Kara, D. Alistarh, J. Liu, and C. Zhang, "ZipML: Training linear models with end-to-end low precision, and a little bit of deep learning," in *Proc. Intel. Conf. Mach. Learn. (ICML)*, 2017.

[11] W. Wen, C. Xu, F. Yan, C. Wu, Y. Wang, Y. Chen, and H. Li, "Terngrad: Ternary gradients to reduce communication in distributed deep learning," in *Proc. Adv. Neural Info. Proc. Sys. (NeurIPS)*, 2017.

[12] A. F. Aji and K. Heafield, "Sparse communication for distributed gradient descent," in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2017.

[13] N. Strom, "Scalable distributed DNN training using commodity GPU cloud computing," in *Proc. Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, 2015.

[14] A. Bereyhi, B. Liang, G. Boudreau, and A. Afana, "Novel gradient sparsification algorithm via Bayesian inference," in *Proc. IEEE Int. Workshop Mach. Learn. Signal Process. (MLSP)*, 2024.

[15] P. Jiang and G. Agrawal, "A linear speedup analysis of distributed deep learning with sparse and quantized communication," in *Proc. Adv. Neural Info. Proc. Sys. (NeurIPS)*, 2018.

[16] A. Abdi and F. Fekri, "Reducing communication overhead via CEO in distributed training," in *Proc. IEEE Intel. Workshop on Signal Process. Advances in Wireless Commun. (SPAWC)*, 2019, pp. 1–5.

[17] N. Singh, D. Data, J. George, and S. Diggavi, "SQuARM-SGD: Communication-efficient momentum SGD for decentralized optimization," *IEEE J. Sel. Areas Inf. Theory*, vol. 2, pp. 954–969, 2021.

[18] L. Abrahamyan, Y. Chen, G. Bekoulis, and N. Deligiannis, "Learned gradient compression for distributed deep learning," *IEEE Trans. Neural Netw. Learn. Syst.*, Jun. 2021.

[19] C.-Y. Chen *et al.*, "Scalecom: Scalable sparsified gradient compression for communication-efficient distributed training," in *Proc. Adv. Neural Info. Proc. Sys. (NeurIPS)*, 2020.

[20] J. Sun, T. Chen, G. Giannakis, and Z. Yang, "Communication-efficient distributed learning via lazily aggregated quantized gradients," in *Proc. Adv. Neural Info. Proc. Sys. (NeurIPS)*, 2019.

[21] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: Wiley, 1991.

[22] M. Soysal and E. G. Schmidt, "Machine learning algorithms for accurate flow-based network traffic classification: Evaluation and comparison," *Perform. Eval.*, vol. 67, pp. 451–467, 2010.

[23] S. Liang, X. Zhang, Z. Ren, and E. Kanoulas, "Dynamic embeddings for user profiling in twitter," in *Proc. ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, 2018.

[24] C. Gutterman *et al.*, "Requet: Real-time QoE metric detection for encrypted YouTube traffic," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 16, pp. 1–28, 2020.

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

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

[27] T. Yang *et al.*, "A survey of distributed optimization," *Annu. Rev. Control*, vol. 46, pp. 278–305, 2019.

[28] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Trans. Automat. Contr.*, vol. 57, no. 3, pp. 592–606, 2012.

[29] M. Raginsky and J. Bouvrie, "Continuous-time stochastic mirror descent on a network: Variance reduction, consensus, convergence," in *IEEE Conf. Decision Control (CDC)*, 2012.

[30] S. Hosseini, A. Chapman, and M. Mesbahi, "Online distributed optimization via dual averaging," in *IEEE Conf. Decision Control (CDC)*, 2013.

[31] S. Shahrampour and A. Jadbabaie, "Distributed online optimization in dynamic environments using mirror descent," *IEEE Trans. Automat. Control*, vol. 63, pp. 714–725, Mar. 2018.

[32] O. Shamir, N. Srebro, and T. Zhang, "Communication-efficient distributed optimization using an approximate Newton-type method," in *Proc. Intel. Conf. Mach. Learn. (ICML)*, 2014.

[33] J. F. C. Mota, J. M. F. Xavier, P. M. Q. Aguiar, and M. Püschel, "D-ADMM: A communication-efficient distributed algorithm for separable optimization," *IEEE Trans. Signal Process.*, vol. 61, pp. 2718–2723, 2013.

[34] C. Liu, H. Li, Y. Shi, and D. Xu, "Distributed event-triggered gradient method for constrained convex minimization," *IEEE Trans. Automat. Contr.*, vol. 65, pp. 778–785, Feb. 2020.

[35] M. Jaggi, V. Smith, M. Takac, J. Terhorst, S. Krishnan, T. Hofmann, and M. I. Jordan, "Communication-efficient distributed dual coordinate ascent," in *Proc. Adv. Neural Info. Proc. Sys. (NeurIPS)*, 2014.

[36] J. Zhang, N. Li, and M. Dedeoglu, "Federated learning over wireless networks: A band-limited coordinated descent approach," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2021.

[37] J. Wang, M. Dong, B. Liang, G. Boudreau, and H. Abou-Zeid, "Online model updating with analog aggregation in wireless edge learning," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2022.

[38] J. Wang, B. Liang, M. Dong, G. Boudreau, and H. Abou-Zeid, "Joint online optimization of model training and analog aggregation for wireless edge learning," *IEEE/ACM Trans. Netw.*, vol. 32, pp. 1212–1228, Oct. 2023.

[39] S. Lee and M. M. Zavlanos, "Distributed primal-dual methods for online constrained optimization," in *Proc. Amer. Control Conf. (ACC)*, 2016.

[40] D. Yuan, D. W. C. Ho, and G.-P. Jiang, "An adaptive primal-dual subgradient algorithm for online distributed constrained optimization," *IEEE Trans. Cybern.*, vol. 48, pp. 3045–3055, Nov. 2018.

[41] D. Yuan, A. Proutiere, and G. Shi, "Distributed online optimization with long-term constraints," *IEEE Trans. Automat. Contr.*, vol. 67, pp. 1089–1104, Mar. 2022.

[42] S. Paternain, S. Lee, M. M. Zavlanos, and A. Ribeiro, "Distributed constrained online learning," *IEEE Trans. Signal Process.*, vol. 68, pp. 3486–3499, Jun. 2020.

[43] P. Sharma, P. Khanduri, L. Shen, D. J. Bucci, and P. K. Varshney, "On distributed online convex optimization with sublinear dynamic regret and fit," in *Proc. Asilomar Conf. Signal Sys. Comput. (ASILOMARSSC)*, 2021.

[44] X. Cao and T. Başar, "Distributed constrained online convex optimization over multiple access fading channels," *IEEE Trans. Signal Process.*, vol. 70, pp. 3468–3483, Jun. 2022.

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

[46] P. Elias, "Predictive coding–I," *IRE Trans. Inf. Theory*, vol. 1, pp. 16–24, 1955.

[47] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proc. IRE*, vol. 40, pp. 1098–1101, 1952.

[48] A. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inf. Theory*, vol. 22, pp. 1–10, 1976.

[49] D. Slepian and J. Wolf, "Noiseless coding of correlated information sources," *IEEE Trans. Inf. Theory*, vol. 19, pp. 471–480, Jul. 1973.

[50] S. Pradhan and K. Ramchandran, "Distributed source coding using syndromes (DISCUS): design and construction," *IEEE Trans. Inf. Theory*, vol. 49, no. 3, pp. 626–643, 2003.

[51] D. J. Costello and G. D. Forney, "Channel coding: The road to channel capacity," *Proc. IEEE*, vol. 95, pp. 1150–1177, 2007.

[52] K. D. Rao, *Channel coding techniques for wireless communications*. Springer, 2015.

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

[54] 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.

[55] X. Wei, H. Yu, and M. J. Neely, "Online primal-dual mirror descent under stochastic constraints," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 4, Jun. 2020.

[56] J. Wang, M. Dong, B. Liang, G. Boudreau, and H. Abou-Zeid, "Delay-tolerant OCO with long-term constraints: Algorithm and its application to network resource allocation," *IEEE/ACM Trans. Netw.*, vol. 31, pp. 147–163, Feb. 2023.

[57] J. Wang, M. Dong, B. Liang, and G. Boudreau, "Periodic updates for constrained OCO with application to large-scale multi-antenna systems," *IEEE Trans. Mobile Comput.*, vol. 22, pp. 6705–6722, Nov. 2023.

[58] 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.

[59] 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.

[60] 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.

[61] L. Nguyen, P. H. Nguyen, M. van Dijk, P. Richtarik, K. Scheinberg, and M. Takac, "SGD and hogwild! Convergence without the bounded gradients assumption," in *Proc. Intel. Conf. Mach. Learn. (ICML)*, 2018.

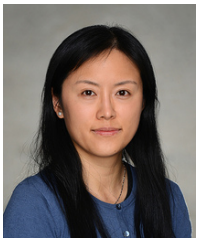
[62] L. Zhang, T. Yang, J. Yi, R. Jin, and Z. Zhou, "Improved dynamic regret for non-degenerate functions," in *Proc. Adv. Neural Info. Proc. Sys. (NeurIPS)*, 2017.

[63] S. P. Karimireddy, S. Kale, M. Mohri, S. Reddi, S. Stich, and A. T. Suresh, "SCAFFOLD: Stochastic controlled averaging for federated learning," in *Proc. Intel. Conf. Mach. Learn. (ICML)*, 2020.

[64] A. Mitra, R. Jaafar, G. J. Pappas, and H. Hassani, "Linear convergence in federated learning: Tackling client heterogeneity and sparse gradients," in *Proc. Adv. Neural Info. Proc. Sys. (NeurIPS)*, 2021.

[65] Y. LeCun, C. Cortes, and C. Burges, "The MNIST database," 1998. [Online]. Available: http://yann.lecun.com/exdb/mnist/

Juncheng Wang (Member, IEEE) received the B.Eng. degree in Electrical Engineering from Shanghai Jiao Tong University, Shanghai, China, in 2014, the M.Sc. degree in Electrical and Computer Engineering from the University of Alberta, Edmonton, AB, Canada, in 2017, and the Ph.D. degree in Electrical and Computer Engineering from the University of Toronto, Toronto, ON, Canada, in 2023. He is currently an Assistant Professor with the Department of Computer Science, Hong Kong Baptist University, Hong Kong, China. His research interests include network artificial intelligence, distributed machine learning, and online optimization.



Min Dong (Fellow, 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, USA, in 2004.

From 2004 to 2008, she was with Qualcomm Research, Qualcomm Inc., San Diego, CA, USA. Since 2008, she has been with Ontario Tech University, where she is currently a Professor with the Department of Electrical, Computer and Software Engineering. She also holds a status-only Professor appointment with the Department of Electrical and Computer Engineering, University of Toronto. Her research interests include wireless communications, statistical signal processing, learning techniques, optimization and control applications in cyber-physical systems. She 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 coauthor 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 is a Senior Area Editor for IEEE TRANSACTIONS ON SIGNAL PROCESSING and an Associate Editor of IEEE OPEN JOURNAL OF SIGNAL PROCESSING. She was an Editor of the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS from 2018 to 2023 and was an Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING from 2010 to 2014 and the IEEE SIGNAL PROCESSING LETTERS from 2009 to 2013. She was also on the Steering Committee of the IEEE TRANSACTIONS ON MOBILE COMPUTING from 2019 to 2021. She was an Elected Member of the Signal Processing for Communications and Networking Technical Committee of IEEE Signal Processing Society from 2013 to 2018.

Ben Liang (Fellow, IEEE) received honors-simultaneous B.Sc. (valedictorian) and M.Sc. degrees in Electrical Engineering from Polytechnic University (now the New York University Tandon School of Engineering) 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 post-doctoral 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 Area Editor for the IEEE *Transactions on Wireless Communications* and has served on the editorial boards of the *IEEE Transactions on Mobile Computing*, the *IEEE Transactions on Communications*, the *IEEE Transactions on Wireless Communications*, and Wiley *Security and Communication Networks*. He is a member of the steering committee for IEEE INFOCOM and 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.

Ali Afana serves as an R&D 5G/6G Wireless Systems Developer at Ericsson, Canada, overseeing collaborative research initiatives between industry and academia in the domains of next-generation radio access networks (RAN) intelligence, low-latency communications, and spectrum sharing technologies.

Within his capacity at Ericsson, Dr Afana spearheads early-phase system designs and algorithm development, actively contributing to the creation of intellectual property through patent filings for the advancement of next-generation RANs. Prior to his tenure at Ericsson, he held instructional and postdoctoral positions at Lakehead University and Memorial University. Dr Afana's research focuses on 5G/6G wireless networking, signal processing for communications, and robust machine learning for networks.

Boasting an impressive portfolio, he co-invented over 20 patent filings and authored over 50 peer-reviewed publications. Dr Afana's contributions to the field have been recognized with the co-recipient title of the IEEE ICC 2022 Best Paper Award. He obtained his PhD. in electrical and computer engineering from Concordia University, Montreal, Canada, in 2014, and his MSc. in communications engineering from Birmingham University, UK, in 2009.

SUPPLEMENTARY MATERIALS

APPENDIX A
PROOF OF COROLLARY 1

Proof: We first prove the case of time-varying weights. Substituting the specified algorithm parameters α, γ, and η into the bound on the performance gap (36) in Theorem 1 and the bound on the constraint violation (41) in Theorem 2 , we have

$$\begin{aligned}\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t)-f_t(\mathbf{x}_t^{\text{ctr}})\big) &\le \mathcal{O}(T^{1-\frac{1-\mu}{2}})+\mathcal{O}(T^{\frac{\nu-1}{4}+1})\\ &\quad+\mathcal{O}(T^{\nu-\frac{3(\nu-1)}{4}})+\mathcal{O}(T^{\frac{1-\mu}{2}+\mu})\\ &=\mathcal{O}(T^{\frac{1+\mu}{2}})+\mathcal{O}(T^{\frac{3+\nu}{4}}),\end{aligned} \tag{82}$$

and

$$\begin{aligned}\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) &\le \Big(\big[\mathcal{O}(T^{\frac{\nu-1}{4}+1})+\mathcal{O}(T^{-\frac{\nu-1}{4}})\big]\\ &\quad\big[\mathcal{O}(T^{1-\frac{1-\mu}{2}})+\mathcal{O}(T^{\frac{\nu-1}{4}+1})+\mathcal{O}(T)+\mathcal{O}(T^{\frac{1-\mu}{2}+\mu})\big]\Big)^{\frac{1}{2}}\\ &\le \Big(\big[\mathcal{O}(T^{\frac{3+\nu}{4}})\big]\big[\mathcal{O}(T)\big]\Big)^{\frac{1}{2}}=\mathcal{O}(T^{\frac{7+\nu}{8}}).\end{aligned} \tag{83}$$

For the case of time-invariant equal weights, we have

$$\begin{aligned}\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t)-f_t(\mathbf{x}_t^{\text{ctr}})\big) &\le \mathcal{O}(T^{1-\frac{1-\mu}{2}})+\mathcal{O}(T^{-\frac{1}{2}+1})+\mathcal{O}(T^{\frac{1-\mu}{2}+\mu})\\ &=\mathcal{O}(T^{\frac{1+\mu}{2}}),\end{aligned} \tag{84}$$

and

$$\begin{aligned}\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) &\le \Big(\big[\mathcal{O}(T^{-\frac{1}{2}+1})+\mathcal{O}(T^{\frac{1}{2}})\big]\\ &\quad\big[\mathcal{O}(T^{1-\frac{1-\mu}{2}})+\mathcal{O}(T^{-\frac{1}{2}+1})+\mathcal{O}(T)+\mathcal{O}(T^{\frac{1-\mu}{2}+\mu})\big]\Big)^{\frac{1}{2}}\\ &\le \Big(\big[\mathcal{O}(T^{\frac{1}{2}})\big]\big[\mathcal{O}(T)\big]\Big)^{\frac{1}{2}}=\mathcal{O}(T^{\frac{3}{4}}).\end{aligned} \tag{85}$$

■

APPENDIX B
PROOF OF COROLLARY 2

Proof: We first prove the case of time-varying weights. Substituting the specified algorithm parameters α, γ, and η into the bound on the performance gap (67) in Theorem 3 and the bound on the constraint violation (73) in Theorem 4 , we have

$$\begin{aligned}\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t)-f_t(\mathbf{x}_t^{\text{ctr}})\big) &\le \mathcal{O}(T^{\mu})+\mathcal{O}(T^{1-(1-\mu)})\\ &\quad+\mathcal{O}(T^{\nu-\frac{1-\mu}{2}-\frac{3(\nu-1)}{4}})+\mathcal{O}(T^{\frac{\nu-1}{4}+1-\frac{1-\mu}{2}})+\mathcal{O}(T^{\mu})\\ &=\mathcal{O}(T^{\mu})+\mathcal{O}(T^{\frac{1+2\mu+\nu}{4}}),\end{aligned} \tag{86}$$

and

$$\begin{aligned}\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) &\le \Big(\big[\mathcal{O}(T^{\frac{\nu-1}{4}+1})+\mathcal{O}(T^{-\frac{\nu-1}{4}})\big]\\ &\quad\big[\mathcal{O}(T^{1-\frac{1-\mu}{2}})+\mathcal{O}(T^{\frac{\nu-1}{4}+1})+\mathcal{O}(T^{\frac{1-\mu}{2}+\mu})\big]\Big)^{\frac{1}{2}}\\ &\le \Big(\big[\mathcal{O}(T^{\frac{3+\nu}{4}})\big]\big[\mathcal{O}(T^{\frac{1+\mu}{2}})+\mathcal{O}(T^{\frac{3+\nu}{4}})\big]\Big)^{\frac{1}{2}}\\ &=\mathcal{O}(T^{\frac{5+\nu+2\mu}{8}})+\mathcal{O}(T^{\frac{3+\nu}{4}}).\end{aligned} \tag{87}$$

For the case of time-invariant equal weights, we have

$$\begin{aligned}\sum_{t=1}^{T}\big(f_t(\hat{\mathbf{x}}_t)-f_t(\mathbf{x}_t^{\text{ctr}})\big) &\le \mathcal{O}(T^{\mu})+\mathcal{O}(T^{1-(1-\mu)})+\mathcal{O}(T^{-\frac{1}{2}+1-\frac{1-\mu}{2}})+\mathcal{O}(T^{\mu})\\ &\le \mathcal{O}(T^{\mu}),\end{aligned} \tag{88}$$

and

$$\begin{aligned}\frac{1}{N}\sum_{t=1}^{T}\sum_{n=1}^{N} g_t^n(\mathbf{x}_t^n) &\le \Big(\big[\mathcal{O}(T^{-\frac{1}{2}+1})+\mathcal{O}(T^{\frac{1}{2}})\big]\\ &\quad\big[\mathcal{O}(T^{1-\frac{1-\mu}{2}})+\mathcal{O}(T^{-\frac{1}{2}+1})+\mathcal{O}(T^{\frac{1-\mu}{2}+\mu})\big]\Big)^{\frac{1}{2}}\\ &\le \Big(\big[\mathcal{O}(T^{\frac{1}{2}})\big]\big[\mathcal{O}(T^{\frac{1+\mu}{2}})\big]\Big)^{\frac{1}{2}}=\mathcal{O}(T^{\frac{2+\mu}{4}}).\end{aligned} \tag{89}$$

■