Age-of-Information Minimization with Weight Limits for Semi-Asynchronous Online Distributed Optimization

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

***Abstract*—We consider online distributed optimization where a server and multiple devices collaborate to minimize a sequence of time-varying global loss functions. To accommodate slow devices that may require multiple time slots to compute their local decisions, the server uses semi-asynchronous aggregation of the local decisions, which complicates device scheduling and performance optimization. In this work, we first analyze the convergence of semi-asynchronous aggregation in the presence of time-varying local update delays and loss-function weights. Our analysis leads to an online scheduling problem to minimize the accumulated age of information on the local decision updates, subject to individual long-term constraints on the total weights of the scheduled devices. We then design an efficient scheduling policy, termed Age-of-Information Minimization with Weight Limits (AIMWeL), through a modified Lyapunov optimization approach that uses the weighted sum of linear age-of-information values and quadratic virtual queues as a new Lyapunov function. We show that AIMWeL has bounded optimality ratio, via a novel double relaxation approach to handle the unique scheduling-dependent communication indicator with time-varying probabilities of completing local decision update caused by semi-asynchronous aggregation. When AIMWeL is applied to semi-asynchronous federated learning, our simulation results based on standard image classification datasets demonstrate that AIMWeL uses significantly less time to reach the same classification accuracy achieved by the current best alternatives for both convex logistic regression and non-convex convolutional neural networks.**

***Index Terms*—Online distributed optimization, federated learning, semi-asynchronous aggregation, age of information.**

I. Introduction

Modern wireless edge devices generate an enormous amount of data that can be used to train machine learning models. Together with the increasing computational capacity of wireless edge devices, *distributed optimization* has become an essential tool for machine learning applications. The celebrated federated learning (FL) scheme allows multiple local devices to collaboratively optimize a global model based on their local private data, with the assistance of a central server [1]. Most existing works on FL assume *synchronous* aggregation, *i.e.,* the central server waits for all the selected devices to finish updating their local models before aggregating the global model [1]-[7]. However, the local computation time may vary drastically among devices due to the heterogeneity in computational capacity. This leads to the *straggler* issue since the central server needs to wait for the slowest devices [8].

In *asynchronous* FL, the central server performs global model update as soon as it receives *one* local model from a local device, while the remaining devices continue to compute and send their model updates [9]-[12]. Naturally, the slower devices may participate in asynchronous aggregation much less frequently than the fast devices, leading to significant *staleness* of their local models relative to the global model. In *semi-asynchronous* FL [13]-[16], the central server waits for a certain number of local devices before global aggregation commences. It works in a *hybrid* mode between the synchronous and asynchronous modes, mitigating the negative impacts of both straggler and staleness on the learning performance.

All existing works on semi-asynchronous FL focus on *offline* optimization based on fixed datasets, which does not allow streaming data or time-varying loss functions. However, in many practical machine learning applications, *e.g.,* real-time video analysis [17], dynamic user profiling [18], and network traffic classification [19], new data arrive in a streaming fashion, and consequently the loss functions vary over time. These applications require *online* optimization, where decisions are continuously updated to adapt to the unknown system dynamics over time [20]. Furthermore, with limited communication capacity between the local devices and the server in practical systems, not all devices that have finished their local computation can be immediately scheduled by the server to send their model updates. It is crucial to understand how device scheduling impacts the performance of semi-asynchronous aggregation.

The above issues motivate us to pose the following key question: *How to dynamically schedule the local devices over time to improve the performance of semi-asynchronous online distributed optimization?* In particular, we are interested in designing a scheduling policy that takes into account the impacts of both the device information staleness and the time-varying environment on semi-asynchronous aggregation, guided by the aim to provide bounded guarantee on optimization performance. To answer the question above, we must address several main challenges: 1) We need to carefully account for the impacts of random time-varying system behavior including device computation time, partial device participation, and

J. 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).

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

M. 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).

G. Boudreau and A. 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.

loss-function weights on distributed optimization. 2) Semi-asynchronous aggregation among devices with heterogeneous computing speeds leads to complex patterns of device information staleness. 3) Device scheduling is coupled with both computation and communication, creating non-independent device status and decision updating sequences that further complicate optimization design and analysis.

In this context, the contributions of this paper are as follows:

- We extend semi-asynchronous FL to the general online distributed optimization setting, under which both the local loss function and local weight are allowed to change over time. We analyze the performance of the resulting semi-asynchronous online distributed optimization framework, in the presence of time-varying partial device participation and local update delays. This analysis is unique in the literature as far as we are aware. From the derived bound on optimality gap, we observe two key factors that determine the performance of semi-asynchronous aggregation: the local update delays and loss-function weights.
- Motivated by the above analysis, we formulate an online scheduling problem to improve the performance of semi-asynchronous aggregation. We leverage the age-of-information (AoI) metric to represent the local decision staleness, due to their similarity in nature. Our goal is to minimize the time-averaged expected weighted sum of AoI subject to individual long-term weight constraints at the local devices. We propose a new Age-of-Information Minimization with Weight Limits (AIMWeL) scheduling policy, which minimizes an upper bound of a modified Lyapunov drift defined from a new Lyapunov function that uses the weighted sum of linear AoI values and quadratic virtual queues. The resulting scheduling decisions are in closed form with low complexity.
- We analyze the performance of AIMWeL in terms of constraint satisfaction and optimality guarantee. Unique to our semi-asynchronous online distributed optimization framework, the communication indicator of each local device *i.e.,* a flag that indicates whether the device is ready to transmit its computed local decision (see definition in Section III-C), is dependent on the scheduling policy with time-varying probabilities of completing local decision updates, which has not been studied before. We therefore propose a new double relaxation approach to bound the optimality ratio of AIMWeL. In the special case of independent and identically distributed (i.i.d.) communication indicators with a fixed probability, AIMWeL recovers the current best optimality ratio.
- For numerical evaluation, we apply AIMWeL to semi-asynchronous FL. We experiment with standard image classification datasets for both convex logistic regression and non-convex convolutional neural networks. Our simulation results demonstrate that AIMWeL significantly reduces the time to reach the same accuracy achieved by the current best alternatives under various scenarios.

The rest of this paper is organized as follows. In Section II, we present the related work. Section III describes the system model for online distributed optimization with semi-asynchronous aggregation. In Section IV, we present AIMWeL and its performance analysis. The application to semi-asynchronous FL is presented in Section V, followed by concluding remarks in Section VI.

II. Related Work

A. Semi-Asynchronous Federated Learning

Semi-asynchronous FL aims at overcoming the detrimental effects of both the straggler in synchronous FL [1]-[7], and the staleness in asynchronous FL [9]-[12]. In [13], the server waits for a certain number of devices before performing global model aggregation. Similarly, in [14], the server buffers the updates from a minimum number of devices before aggregation commences. The number of devices that the server waits to perform global model aggregation was optimized by minimizing a performance upper bound of semi-asynchronous FL [15]. A multi-armed bandit based approach was proposed in [16] to determine the numbers of model updates at the local devices. However, these works focus on *offline* learning, assuming that the local datasets and weights are fixed during the entire learning process. Furthermore, their *heuristic* scheduling policies lack theoretical insights on how scheduling can improve the performance of semi-asynchronous aggregation and do not provide any scheduling performance guarantee.

B. Age of Information in Federated Learning

AoI measures the time that elapsed since the generation of the information that was delivered to the destination, capturing the information freshness [21]. AoI has been studied in areas such as queueing networks, wireless scheduling, and energy harvesting (see [22] for an overview). In [23], the communication round duration was measured by an age metric called age of update (AoU) to represent the staleness of the global model aggregation in synchronous FL. AoU based scheduling policy was proposed in [24] to greedily minimize the aggregation staleness while considering the channel conditions. The age-optimal number of devices updating their local models and participating in global aggregation was studied in [25]. These works focus on *synchronous* FL and do not provide any theoretical analysis on the impact of AoI on the FL performance.

C. Constrained Age-of-Information Scheduling

More relevant to this work in scheduling theory is constrained AoI scheduling [26]-[30]. In [26], scheduling policies were proposed to minimize the weighted sum of AoI subject to throughput constraints in wireless sensor networks, where multiple sensors share one single interference channel. Lyapunov optimization was used in [27] to minimize the transmit power subject to AoI constraints, and in [28] to minimize the sampling and transmission costs under AoI constraints. Constrained Markov decision techniques were used for AoI minimization with power constraints over time-varying channels [29], and for throughput maximization subject to AoI constraints over fading channels [30]. These works do not consider the impact of AoI on the performance of online

distributed optimization and assume the communication indicator is *independent* of the scheduling policy with a *fixed* probability. In this work, as will be shown in Section III-C, the communication indicator sequence of each device is *dependent* on the scheduling policy with *time-varying* probabilities of completing local decision updates due to semi-asynchronous aggregation. This brings new challenges to the scheduling algorithm design and its performance analysis.

D. Online Optimization and Lyapunov Optimization

Due to the dynamic nature of semi-asynchronous online distributed optimization, a part of our performance analysis resembles online convex optimization (OCO) [20], especially distributed OCO with consensus [31]-[34]. However, the distributed OCO framework mainly concerns *synchronous* consensus over *all* devices at each time, which is inherently different from our *semi-asynchronous* optimization framework with decision aggregation over *time-varying partial* device participation.

AIMWeL is also related to Lyapunov optimization [35], since our online scheduling problem for optimizing the performance of semi-asynchronous aggregation involves long-term constraints. However, different from the standard Lyapunov optimization techniques, we design a new Lyapunov function, which is a weighted sum of *linear* AoI values and *quadratic* virtual queues, to handle the policy-dependent communication indicator sequence that is unique to our system. Furthermore, we use a novel *double* relaxation approach to bound the optimality ratio of AIMWeL, which is substantially different from the standard Lyapunov optimization bounding approach. Specifically, we keep relaxing the original online problem with non-i.i.d. system states until an optimal stationary randomized policy for solving a *relaxed* problem also achieves the optimal objective value of a doubly-relaxed *lower bound* problem. In standard Lyapunov optimization, the system states are assumed to be i.i.d. or Markovian, and an optimal stationary randomized solution to the original optimization problem readily exists.

III. Online Distributed Optimization with Semi-Asynchronous Aggregation

A. Online Distributed Optimization Objective

We consider a distributed system consisting of N local devices and a central server. At each time slot $t \in \mathcal{T} \triangleq \{1, \ldots, T\}$, each local device $n \in \mathcal{N} \triangleq \{1, \ldots, N\}$ observes a *local* loss function $f_t^n(\mathbf{x}_t) : \mathbb{R}^d \to \mathbb{R}$, where $\mathbf{x}_t \in \mathbb{R}^d$ is the decision variable. Under the online optimization setting, the local loss function $f_t^n(\mathbf{x}_t)$ is allowed to vary over time.

In our motivating example of FL, $f_t^n(\mathbf{x}_t)$ can be defined as the average loss incurred by the learning model $\mathbf{x}_t$ with respect to the local dataset $\mathcal{B}_t^n$, given by

$$f_t^n(\mathbf{x}_t) \triangleq \frac{1}{\beta_t^n} \sum_{i \in \mathcal{B}_t^n} l(\mathbf{x}_t; \boldsymbol{\mu}_t^{n,i}, \nu_t^{n,i}) \tag{1}$$

where β_t^n is the cardinality of $\mathcal{B}_t^n$ and $l(\mathbf{x}_t; \boldsymbol{\mu}_t^{n,i}, \nu_t^{n,i}) : \mathbb{R}^d \to \mathbb{R}$ is a sample-wise loss function to represent how the learning model $\mathbf{x}_t$ performs on each data sample $(\boldsymbol{\mu}_t^{n,i}, \nu_t^{n,i})$ in $\mathcal{B}_t^n$, with $\boldsymbol{\mu}_t^{n,i}$ being a data feature vector and $\nu_t^{n,i}$ being its true label. Note that $l(\mathbf{x}; \boldsymbol{\mu}, \nu)$ is generally defined, *e.g.,* it can be the logistic regression loss (see Section V-B) or the neural network loss (see Section V-C) to measure the prediction accuracy. In the case of FL with *streaming* data that sequentially arrive to the devices, $\mathcal{B}_t^n$ represents the local streaming dataset collected by device n at time t. The distribution of $\mathcal{B}_t^n$ may be unknown and vary over time, so that the corresponding loss function $f_t^n(\mathbf{x}_t)$ is also time-varying.

The *global* loss function at time t is defined as the weighted sum of the local loss functions, given by

$$f_t(\mathbf{x}_t) \triangleq \sum_{n \in \mathcal{N}} w_t^n f_t^n(\mathbf{x}_t) \tag{2}$$

where $w_t^n > 0$ is the weight on local device n with $\sum_{n \in \mathcal{N}} w_t^n = 1$. In the FL example, when we set the local weight as $w_t^n = \frac{\beta_t^n}{\beta_t}$ with $\beta_t = \sum_{n \in \mathcal{N}} \beta_t^n$, the global loss is equal to the average loss incurred by the global dataset $\mathcal{B}_t = \bigcup_{n \in \mathcal{N}} \{\mathcal{B}_t^n\}$. Note that due to the unpredictable nature of streaming data or device computation availability, each device n may process different amounts of data samples $\{\beta_t^n\}$ over time, leading to a sequence of time-varying weights $\{w_t^n\}$. We assume w_t^n is mean stationary and let $\bar{w}^n \triangleq \mathbb{E}\{w_t^n\}$.

Let $\mathbf{x}_t^\star \in \arg\min_{\mathbf{x} \in \mathbb{R}^d} f_t(\mathbf{x})$ be the optimal global decision that minimizes $f_t(\mathbf{x})$ at time t. The goal of online distributed optimization is to compute at the central server a sequence of global decisions $\{\mathbf{x}_t\}$, to minimize the difference between the time-averaged global loss yielded by $\{\mathbf{x}_t\}$ and the one by the global optimal solution sequence $\{\mathbf{x}_t^\star\}$ over a finite time horizon T, *i.e.,*

$$\min_{\{\mathbf{x}_t\}} \quad \frac{1}{T} \sum_{t \in \mathcal{T}} \big[f_t(\mathbf{x}_t) - f_t(\mathbf{x}_t^\star) \big]. \tag{3}$$

Solving the above optimization problem is similar to minimizing the *dynamic regret* in the distributed OCO literature [31], [33], [34]. Note that computing a sequence of global decisions $\{\mathbf{x}_t\}$ that satisfies $\lim_{T \to \infty} \frac{1}{T} \sum_{t \in \mathcal{T}} [f_t(\mathbf{x}_t) - f_t(\mathbf{x}_t^\star)] = 0$ is equivalent to achieving *sublinear* dynamic regret, *i.e.,* $\sum_{t \in \mathcal{T}} [f_t(\mathbf{x}_t) - f_t(\mathbf{x}_t^\star)] = o(T)$. As mentioned in Section II-D, the distributed OCO framework [31]-[34] is limited to *synchronous* aggregation over *all* devices at each time. It is a special case of our general *semi-asynchronous* online distributed optimization framework with *time-varying partial* device participation considered in Section III-B.

B. Semi-Asynchronous Aggregation

We extend semi-asynchronous FL [13]-[16], to a general semi-asynchronous online distributed optimization framework, allowing the loss function, local weight, device participation, and update delay to vary over time. For each local device n, if it receives a global decision $\mathbf{x}_t$ from the central server at the *beginning* of time slot t, it performs the following local decision update via local gradient descent:

$$\mathbf{x}_t^n = \mathbf{x}_t - \eta \nabla f_t^n(\mathbf{x}_t) \tag{4}$$

where $\eta > 0$ is the step size (or learning rate).

Most studies in FL assume all the local devices can finish calculating their local gradients within one time slot. However, gradient calculation is computationally costly for high-dimensional functions, and in practical distributed networks, local gradient calculation could take more than one time slot at slow devices. Therefore, in this work we consider a more general setting, so that when device n is computing its local gradient at any time slot t, it completes the computation within the time slot with a probably $0 < p_t^n \leq 1$. This captures the uncertainties both in gradient computation complexity and in the availability of computation resource. We further consider $\{p_t^n\}$ as a random sequence and assume that it is mean stationary and lower bounded, with $\bar{p}^n \triangleq \mathbb{E}\{p_t^n\}$ and $0 < p_{\text{LB}}^n \leq p_t^n, \forall t$.[1] Note that we do *not* require $\{p_t^n\}$ to be an i.i.d. sequence, *i.e.,* we allow the computation time to be *non-memoryless*.

Let $\mathcal{N}_t$ be the set of local devices from which the central server receives local decisions at time t. As explained above, the local decision uploaded by device n at time t may have been delayed over multiple time slots. Let l_t^n be the time when device n last received the global decision $\mathbf{x}_{l_t^n}$ from the central server. The central server performs the following global decision update via decision averaging at the *end* of each time slot t, given by

$$\mathbf{x}_{t+1} = \sum_{n\in\mathcal{N}_t} w_t^n \mathbf{x}_{l_t^n}^n + \sum_{m\in\mathcal{N}\backslash\mathcal{N}_t} w_t^m \mathbf{x}_t. \tag{5}$$

In the first term, $\mathbf{x}_{l_t^n}^n$ is the local decision update from device $n \in \mathcal{N}_t$ based on its last received global decision $\mathbf{x}_{l_t^n}$ at time l_t^n. The second term $\sum_{m\in\mathcal{N}\backslash\mathcal{N}_t} w_t^m \mathbf{x}_t$ in (5) ensures the sum of local weights over all devices $\mathcal{N}$ is 1, to prevent the global decision parameter values from approaching 0 due to partial device participation. The intuition behind (5) is that the central server performs decision averaging over the local decisions $\{\mathbf{x}_{l_t^n}^n\}$ *actually received* from devices $\mathcal{N}_t$, based on the current weights $\{w_t^n\}$ at each time t. For each device $m \in \mathcal{N}\backslash\mathcal{N}_t$ that has not finished updating $\mathbf{x}_{l_t^m}^m$, the central server treats the previous global decision $\mathbf{x}_t$ as a *virtually received* local decision to perform decision averaging in (5). The central server then broadcasts the updated global decision $\mathbf{x}_{t+1}$ to the local devices $\mathcal{N}_t$ at the beginning of time slot $t+1$. Note that device $m \in \mathcal{N}\backslash\mathcal{N}_t$ has not finished calculating its local decision $\mathbf{x}_{l_t^m}^m$ from its received latest global decision $\mathbf{x}_{l_t^m}$ and will continue the same calculation in the next time slot $t+1$.

In Fig. 1, we illustrate the procedure of semi-asynchronous aggregation with $N = 3$ devices. Let τ_t^n be the elapsed time from time slot l_t^n (when device n last received the global decision $\mathbf{x}_{l_t^n}$) to the current time slot t. For example, device 1 receives the global decision $\mathbf{x}_2$ at the beginning of time slot 2, and takes $\tau_3^1 = 2$ time slots to finish updating $\mathbf{x}_2^1$ at time slot $t = 3$. Device 1 then uploads $\mathbf{x}_2^1$, which was updated based on its last received global model $\mathbf{x}_2$ at time slot $l_3^1 = 2$, to the central server at the end of time slot $t = 3$. Similarly, device 3

[1]In practical systems where each device n has *non-zero* computational capacity to perform local decision updates, we have $p_{\text{LB}}^n > 0$. This will be used to bound the impact of time-varying probabilities $\{p_t^n\}$ on the performance of AIMWeL in Sections IV-D and IV-E.

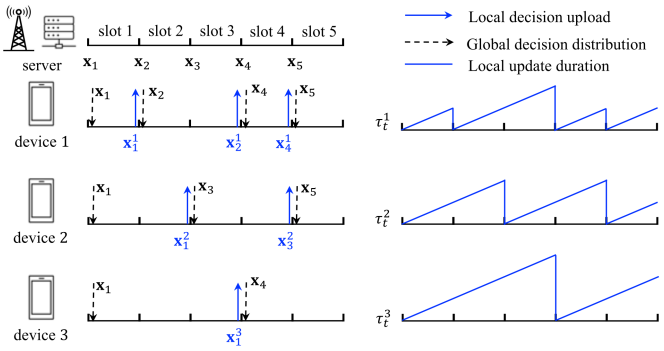


Fig. 1: An illustration of semi-asynchronous online distributed optimization with $N = 3$ devices.

receives the global decision $\mathbf{x}_1$ at the beginning of time slot 1, and takes $\tau_3^3 = 3$ time slots to finish updating $\mathbf{x}_1^3$ at time slot $t = 3$. Device 3 then uploads $\mathbf{x}_1^3$, which was updated based on its last received global model $\mathbf{x}_1$ at time slot $l_3^3 = 1$, to the central server at the end of time slot $t = 3$. After receiving $\mathbf{x}_2^1$ and $\mathbf{x}_1^3$ from devices 1 and 3, the central server performs global decision update to generate $\mathbf{x}_4 = w_3^1\mathbf{x}_2^1 + w_3^3\mathbf{x}_1^3 + w_3^2\mathbf{x}_3$ ($\mathcal{N}_3 = \{1, 3\}$) and distribute it to devices 1 and 3. After receiving $\mathbf{x}_4$ from the central server at the beginning of time slot $t = 4$, devices 1 and 3 then start to generate their new local decisions based on $\mathbf{x}_4$.

C. Device Scheduling Policy

Practical systems have limited communication capacity, *e.g.,* in wireless edge computing. Therefore, we consider the general scenario where the central server can only select up to $K \leq N$ devices to upload their local decisions. Let u_t^n be the *scheduling indicator* such that $u_t^n = 1$ if the central server selects device n to upload its local decision $\mathbf{x}_{l_t^n}^n$, and $u_t^n = 0$ otherwise. A scheduling policy controls the scheduling decisions of the central server at the end of each time slot t, which is represented by $\{u_t^n\}$. We consider non-anticipative scheduling policies Π, *i.e.,* a policy $\pi \in \Pi$ does not utilize any future information in decision making. Denote d_t^n as the *communication indicator* such that $d_t^n = 1$ if device n has finished updating its local decision $\mathbf{x}_{l_t^n}^n$ by the end of time t, and $d_t^n = 0$ otherwise. If device n is ready to upload $\mathbf{x}_{l_t^n}^n$ to the central server, *i.e.,* $d_t^n = 1$, but the central server does not select device n to participate in the global decision update at time t, *i.e.,* $u_t^n = 0$, then device n becomes idle and is also ready to upload at the next time slot $t+1$, *i.e.,* $d_{t+1}^n = 1$. Otherwise, device n finishes its local decision update at time $t+1$ with a probability p_{t+1}^n. Therefore, the evolution of the communication indicator d_t^n follows

$$\mathbb{P}\{d_{t+1}^n = 1\} = \begin{cases} 1, & \text{if } d_t^n = 1, u_t^n = 0, \\ p_{t+1}^n, & \text{o.w.} \end{cases} \tag{6}$$

From (6), we can see that the evolution of d_{t+1}^n depends on u_t^n, and therefore $\mathbb{E}\{d_t^n\}$ is policy π-*dependent*. However, $\mathbb{E}\{d_t^n\}$ is both upper and lower bounded as

$$p_{\text{LB}}^n \leq \mathbb{E}\{d_t^n\} \leq 1. \tag{7}$$

As will be shown later in Section IV-D, the π-dependent communication indicator sequence $\{d_t^n\}$ in (6), with time-varying probabilities $\{p_t^n\}$ of completing local decision updates brought by semi-asynchronous aggregation, requires new techniques to design the scheduling algorithm and to bound its performance. When device n is selected to upload its local decision at time t, the local update delay, *i.e.,* the elapsed time since it last received the global decision at l_t^n, is

$$\tau_t^n = t - l_t^n + 1. \tag{8}$$

Note that τ_t^n depends on both the d_t^n and u_t^n sequences, so it is also policy-dependent. If the central server does not receive a local decision from device n at time t, then $\tau_{t+1}^n = \tau_t^n + 1$. On the other hand, if the central server receives $\mathbf{x}_{l_t^n}^n$ from device n at time t, then the update delay at the next time slot $t+1$ reduces to $\tau_{t+1}^n = 1$. Thus, the evolution of τ_t^n follows

$$\tau_{t+1}^n = \begin{cases} \tau_t^n + 1, & \text{if } d_t^n u_t^n = 0, \\ 1, & \text{o.w.} \end{cases} \tag{9}$$

or equivalently $\tau_{t+1}^n - \tau_t^n = -\tau_t^n d_t^n u_t^n + 1$. We observe that (9) is similar to the evolution of the AoI in nature.[2] Therefore, in the following, we use AoI to refer to τ_t^n. In Table I, we summarize our key notations.

In this work, we aim at finding a scheduling policy to minimize the optimality gap in (3) under semi-asynchronous aggregation. However, it is challenging to directly measure the impact of the scheduling decisions $\{u_t^n\}$ on (3), due to the system dynamics such as time-varying local loss function $f_t^n(\mathbf{x}_t)$, AoI τ_t^n, weight w_t^n, and communication indicator d_t^n. To tackle this challenge, we first derive an upper bound on (3). We then formulate an online scheduling problem to minimize this upper bound.

IV. AoI Minimization with Weight Limits

In this section, we present the Age-of-Information Minimization with Weight Limits (AIMWeL) scheduling policy to optimize the performance of semi-asynchronous aggregation for online distributed optimization.

A. Performance Bound on Semi-Asynchronous Aggregation

We first derive a bound on (3) for the semi-asynchronous online distributed optimization framework. We state the following assumptions required for our performance analysis. These assumptions are common in existing works on FL and distributed optimization [4], [7], [10], [15].

Assumption 1. The optimal global decision $\mathbf{x}_t^\star \in \arg\min_{\mathbf{x}\in\mathbb{R}^d} f_t(\mathbf{x})$ has zero gradient and lower bounded loss, *i.e.,* for any t

$$\nabla f_t(\mathbf{x}_t^\star) = \mathbf{0}, \tag{10}$$

$$f_t(\mathbf{x}_t^\star) > -\infty. \tag{11}$$

[2]The classic AoI evolution commonly assumes the communication indicator d_t^n, *e.g.,* the communication channel on and off indicator, is *i.i.d.* with a *known* mean $\mathbb{E}\{d_t^n\}$, and thus is policy π-*independent*, [26]-[30], [36]. Our model here is more general.

TABLE I: Summary of Key Notations

Notation	Description
$\mathcal{T}$	Set of time slots
$\mathcal{N}$	Set of local devices
T	Total number of time slots
N	Total number of local devices
$f_t^n(\mathbf{x})$	Local loss function of device n at time t
$f_t(\mathbf{x})$	Global loss function at time t
w_t^n	Local function weight of device n at time t
$\mathbf{x}_t^n$	Local decision of device n at time t
$\mathbf{x}_t$	Global decision of central server at time t
$\mathbf{x}_t^\star$	Optimal global decision that minimizes $f_t(\mathbf{x})$
$\mathcal{N}_t$	Set of devices that server receives decisions at time t
η	Gradient descent step size (or learning rate)
l_t^n	Time when device n last received $\mathbf{x}_{l_t^n}$ from server
τ_t^n	AoI (or decision update delay) of device n at time t
u_t^n	Scheduling indicator of device n at time t
d_t^n	Communication indicator of device n at time t
p_t^n	Probability that device n finishes updating at time t
p_{LB}^n	Lower bound constant on p_t^n
μ	Strongly convex constant of $f_t^n(\mathbf{x})$
L	Smoothness constant of $f_t^n(\mathbf{x})$
ϵ	Dissimilarity constant of local gradient $\nabla f_t^n(\mathbf{x})$
Δf_{UB}	Upper bound of global loss function variation
∇f_{UB}	Upper bound of local gradient on optimal decision
τ_{UB}	Upper bound on AoI
w_{LB}	Lower bound on sum weight of scheduled devices
ρ	Contraction constant on global loss
δ	Residual constant on global loss
α^n	Weight on the AoI of device n
q^n	Minimum time-averaged weight limit of device n
K	Maximum number of scheduled devices at time t
π	Scheduling policy
Π	Set of non-anticipative scheduling policies
$\mathbf{P}$	Online scheduling problem with weight limits
$\text{OPT}^\star$	Optimal objective value of $\mathbf{P}$
OPT_π	Objective value of $\mathbf{P}$ achieved by policy π
Q_t^n	Virtual queue of device n at time t
L_t	Modified Lyapunov function at time t
U	Weight on AoI in L_t
Δ_t	Modified Lyapunov drift at time t
S_t	System state at time t
$\mathbf{P}_t$	Per-slot optimization problem that AIMWeL solves
W_t^n	Weight AIMWeL calculates for device n at time t

Assumption 2. The local loss function $f_t^n(\mathbf{x})$ is μ-strongly convex: $\exists \mu > 0$, *s.t.,* for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, n, and t

$$f_t^n(\mathbf{y}) - f_t^n(\mathbf{x}) \geq \langle \nabla f_t^n(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{\mu}{2}\|\mathbf{y} - \mathbf{x}\|^2 \tag{12}$$

where $\langle \mathbf{a}, \mathbf{b} \rangle$ represents the inner product of vectors $\mathbf{a}$ and $\mathbf{b}$, and $\|\cdot\|$ represents the Euclidean norm.

Assumption 3. The local loss function $f_t^n(\mathbf{x})$ is L-smooth: $\exists L > 0$, *s.t.,* for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$, n, and t

$$f_t^n(\mathbf{y}) - f_t^n(\mathbf{x}) \leq \langle \nabla f_t^n(\mathbf{x}), \mathbf{y} - \mathbf{x} \rangle + \frac{L}{2}\|\mathbf{y} - \mathbf{x}\|^2. \tag{13}$$

Assumption 4. The gradient of the local loss function $\nabla f_t^n(\mathbf{x})$ is ϵ-dissimilar to the gradient of the global loss function $\nabla f_t(\mathbf{x})$: $\exists \epsilon > 0$, *s.t.,* for any $\mathbf{x} \in \mathbb{R}^d$, n, and t

$$\langle \nabla f_t^n(\mathbf{x}), \nabla f_t(\mathbf{x}) \rangle \geq \epsilon \|\nabla f_t(\mathbf{x})\|^2. \tag{14}$$

Assumption 4 is weaker than a more common bounded gradient variance assumption that $\exists \gamma \in [0,1)$, *s.t.,* $\|\nabla f_t^n(\mathbf{x}) - \nabla f_t(\mathbf{x})\|^2 \leq \gamma \|\nabla f_t(\mathbf{x})\|^2, \forall \mathbf{x} \in \mathbb{R}^d, n, t$. We can show that Assumption 4 can be derived from setting $\epsilon = \frac{1-\gamma}{2}$.

Unlike many existing works that consider only fixed datasets (or fixed loss functions), we examine the joint impact of time-varying loss function $f_t^n(\mathbf{x})$, local weight w_t^n, partial device participation $\mathcal{N}_t$, and AoI τ_t^n on the performance of semi-asynchronous aggregation. To this end, we need to quantify the amount of variations in the underlying system. Let the variation of the global loss function be upper bounded for any $\mathbf{x} \in \mathbb{R}^d$ and t by

$$|f_t(\mathbf{x}) - f_{t+1}(\mathbf{x})| \leq \Delta f_{\text{UB}}. \tag{15}$$

Furthermore, we need to quantify the heterogeneity in local loss functions. Let the local gradient on the globally optimal decision be upper bounded for any n and t by

$$\|\nabla f_t^n(\mathbf{x}_t^\star)\|^2 \leq \nabla f_{\text{UB}}. \tag{16}$$

The following theorem provides a performance upper bound on semi-asynchronous online distributed optimization.

Theorem 1. If the step size is set as $\eta < \frac{2\mu\epsilon}{L^2}$ in the local decision update (4), the final global decision $\mathbf{x}_T$ satisfies

$$f_T(\mathbf{x}_T) - f_T(\mathbf{x}_T^\star) \leq \rho^T \big[f_1(\mathbf{x}_1) - f_1(\mathbf{x}_1^\star)\big] + \delta \tag{17}$$

where $\rho < 1$ is a contraction constant given by

$$\rho \triangleq \big[1 - \eta\big(2\mu\epsilon - \eta L^2\big) w_{\text{LB}}\big]^{\frac{1}{\tau_{\text{UB}}}}, \tag{18}$$

and $\delta \geq 0$ is a residual constant given by

$$\delta \triangleq \frac{2\tau_{\text{UB}} \Delta f_{\text{UB}} + \eta^2 L \nabla f_{\text{UB}}}{\eta\big(2\mu\epsilon - \eta L^2\big) w_{\text{LB}}}, \tag{19}$$

with $\tau_{\text{UB}} \geq \tau_t^n, \forall n, \forall t$ being an upper bound on the AoI and $w_{\text{LB}} \leq \sum_{n \in \mathcal{N}_t} w_t^n, \forall t$ being a lower bound on the sum weights of the scheduled devices.

Proof: See Appendix A.

Theorem 1 provides a performance bound for the final global decision $\mathbf{x}_T$. As T approaches infinity, the final loss is δ-optimal. Note that δ can be small, *e.g.,* in the special case of *offline* distributed optimization that the local loss functions (or datasets in FL) are fixed over time, *i.e.,* $\Delta f_{\text{UB}} = 0$, and the local loss functions are the same among devices, *i.e.,* $\nabla f_{\text{UB}} = 0$, we have $\delta = 0$. The following corollary provides a bound on the optimality gap (3) yielded by the sequence of the global decisions $\{\mathbf{x}_t\}$.

Corollary 1. The time-averaged loss yielded by the sequence of global decisions $\{\mathbf{x}_t\}$ is upper bounded by

$$\frac{1}{T} \sum_{t \in \mathcal{T}} \big[f_t(\mathbf{x}_t) - f_t(\mathbf{x}_t^\star)\big] \leq \frac{f_1(\mathbf{x}_1) - f_1(\mathbf{x}_1^\star)}{(1-\rho)T} + \delta. \tag{20}$$

Proof: See Appendix B.

Corollary 1 implies that the time-averaged loss converges to a δ-neighbourhood of the optimum at an $\mathcal{O}(\frac{1}{T})$ rate.

Remark 1. (*Bounded AoI and Weight*) In practical systems where each device n uploads its local decisions within finite time, its AoI is bounded above, *i.e.,* $\tau_t^n \leq \tau_{\text{UB}} < \infty, \forall n, \forall t$. Also, when as least one device participate in decision averaging at each time t, the sum weights of the scheduled devices is bounded below, *i.e.,* $\sum_{n \in \mathcal{N}_t} w_t^n \geq w_{\text{LB}} > 0, \forall t$. In Section IV-E, we will prove that AIMWeL provides a bounded optimality ratio of the weighted sum AoI. We can then prove by contradiction that AIMWeL guarantees the existence of τ_{UB}. Furthermore, in Section IV-C, we will see that if there is at least one device ready to upload its local decision at each time t, AIMWeL guarantees the existence of w_{LB}. Furthermore, AIMWeL does not need the values of τ_{UB} and w_{LB} to run.

B. AoI Minimization with Weight Limits

From Theorem 1, we can see that the performance of semi-asynchronous aggregation improves as the contraction constant ρ in (18) and the residual constant δ in (19) decrease. We further observe the following two key factors that determine the value of ρ and δ:[3]

- *AoI* τ_t^n: Each device n should have small τ_t^n such that more devices can contribute in-time information to global decision update (5).
- *Function weight* w_t^n*:* More devices with larger weights should be scheduled to participate in global decision update (5), such that $\sum_{n \in \mathcal{N}_t} w_t^n$ is large.

Based on these two observations, we aim at making a sequence of scheduling decisions $\{u_t^n\}$ at the central server to minimize the long-term time-averaged expected weighted sum of AoI, subject to individual long-term weight constraints. This leads to the following online scheduling optimization problem:

$$\mathbf{P}: \quad \text{OPT}^\star = \min_{\pi \in \Pi} \left\{ \lim_{T \to \infty} \frac{1}{T} \sum_{t \in \mathcal{T}} \sum_{n \in \mathcal{N}} \alpha^n \mathbb{E}\{\tau_t^n\} \right\} \tag{21a}$$

$$\text{s.t.} \quad \lim_{T \to \infty} \frac{1}{T} \sum_{t \in \mathcal{T}} \mathbb{E}\{w_t^n d_t^n u_t^n\} \geq q^n, \quad \forall n, \tag{21b}$$

$$\sum_{n \in \mathcal{N}} u_t^n \leq K, \quad \forall t \tag{21c}$$

where $\alpha^n > 0$ is the scaling factor for AoI on device n with $\sum_{n \in \mathcal{N}} \alpha^n = 1$, $q^n > 0$ is the average minimum weight limit, and $K \leq N$ is the maximum number of participating local devices at each t as explained in Section III-C.

For a given network setup $(N, K, \alpha^n, \{w_t^n\}, \{p_t^n\}, q^n)$, let $\pi^\star \in \Pi$ be the optimal online scheduling policy for solving $\mathbf{P}$. Let $\text{OPT}^\star$ be the optimal objective of $\mathbf{P}$ achieved by $\pi^\star$. Similarly, let OPT_π be the objective of $\mathbf{P}$ achieved by some policy $\pi \in \Pi$. The *optimality ratio* of π to $\pi^\star$ is defined as

$$\frac{\text{OPT}_\pi}{\text{OPT}^\star}. \tag{22}$$

[3]The constants Δf_{UB} and ∇f_{UB} in δ are determined by the underlying system and are independent of the scheduling decisions. Scheduling devices to minimize τ_t^n and maximize $\sum_{n \in \mathcal{N}_t} w_t^n$ helps to reduce τ_{UB} and increase w_{LB}, leading to improved performance as seen in Theorem 1.

We assume constraint (21b) in $\mathbf{P}$ is strictly feasible, *i.e.,* there exists a set of scheduling probabilities $\{\tilde{v}^n\}$ that satisfy $\sum_{n\in\mathcal{N}}\tilde{v}^n \leq K$ and

$$\bar{w}^n p^n_{\text{LB}}\tilde{v}^n - q^n \geq \sigma, \quad \forall n \tag{23}$$

where $\bar{w}^n$ and p^n_{LB} are defined in Section III-A and Section III-B, respectively, and $\sigma > 0$ is a Slater's constant for the long-term weight constraints (21b).

C. AIMWeL Scheduling Policy

We now present the design details of AIMWeL for solving $\mathbf{P}$. Different from the standard Lyapunov optimization [35], we introduce a new form of Lyapunov function that is a weighted sum of *linear* AoI values and *quadratic* virtual queues to handle the policy-dependent communication indicator sequence d^n_t with a time-varying probability p^n_t of completing local decision update.

We first introduce a virtual queue Q^n_t at each device n to account for the long-term weight constraints (21b) in $\mathbf{P}$, with the following updating rule:

$$Q^n_{t+1} \triangleq tq^n - \sum_{\tau=1}^{t} w^n_t d^n_t u^n_t, \quad \forall n, \forall t \tag{24}$$

or equivalently $Q^n_{t+1} - Q^n_t = q^n - w^n_t d^n_t u^n_t, \forall n, \forall t$. The concept of virtual queues was also used in [26], [36] for constrained AoI minimization. Unique to our virtual queue in (24), the communication indicator sequence d^n_t is dependent on the scheduling decision sequence u^n_t with a time-varying probability p^n_t. Define $[\cdot]_+ = \max\{\cdot, 0\}$ as a projection operator that computes the positive part of a scalar. From Theorem 2.8 in [35], strong stability of the virtual queue

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t\in\mathcal{T}}\mathbb{E}\big\{[Q^n_t]_+\big\} < \infty, \quad \forall n \tag{25}$$

is sufficient to satisfy (21b) in $\mathbf{P}$.

Let $S_t \triangleq \{\{\tau^n_t\}_{n\in\mathcal{N}}, \{Q^n_t\}_{n\in\mathcal{N}}\}$ denote the system state at time t. Note that it contains both the AoI values and the virtual queues. We define a new Lyapunov function as follows:

$$L_t \triangleq U\sum_{n\in\mathcal{N}}\alpha^n\tau^n_t + \frac{1}{2}\sum_{n\in\mathcal{N}}[Q^n_t]^2_+, \quad \forall t \tag{26}$$

where $U > 0$ is a weight on the AoI. Different from the standard Lyapunov function that is a quadratic function on Q^n_t only, L_t in (26) is a weighted sum of the linear AoI values and the quadratic virtual queues. Define the corresponding *modified* Lyapunov drift as

$$\Delta_t \triangleq L_{t+1} - L_t, \quad \forall t. \tag{27}$$

We provide an upper bound on its conditional expectation in the following lemma.

Lemma 1. The modified Lyapunov drift is upper bounded by

$$\begin{aligned}\mathbb{E}\big\{\Delta_t|S_t\big\} \leq &-\sum_{n\in\mathcal{N}}U\alpha^n\tau^n_t\mathbb{E}\big\{d^n_t u^n_t|S_t\big\}\\ &-\sum_{n\in\mathcal{N}}[Q^n_t]_+\mathbb{E}\big\{w^n_t d^n_t u^n_t|S_t\big\} + B_t\end{aligned} \tag{28}$$

where $B_t \triangleq \sum_{n\in\mathcal{N}}q^n[Q^n_t]_+ + U + \frac{CN}{2}$ with $C \triangleq \max\{(q^{\max})^2, (w^{\max}-q^{\min})^2\}$, $q^{\max} \triangleq \max_n\{q^n\}$, $q^{\min} \triangleq \min_n\{q^n\}$, and $w^{\max} \triangleq \max_{t,n}\{w^n_t\}$.

Proof: Substituting L_t in (26) into Δ_t in (27) and taking conditional expectation over the system state S_t, we have

$$\begin{aligned}\mathbb{E}\{\Delta_t|S_t\} = &U\sum_{n\in\mathcal{N}}\alpha^n\mathbb{E}\big\{\tau^n_{t+1} - \tau^n_t|S_t\big\}\\ &+\frac{1}{2}\sum_{n\in\mathcal{N}}\mathbb{E}\big\{[Q^n_{t+1}]^2_+ - [Q^n_t]^2_+|S_t\big\}.\end{aligned} \tag{29}$$

From the equivalent AoI evolution below (9), we have

$$\mathbb{E}\big\{\tau^n_{t+1} - \tau^n_t|S_t\big\} = -\tau^n_t\mathbb{E}\big\{d^n_t u^n_t|S_t\big\} + 1 \tag{30}$$

From the equivalent virtual queue updating rule below (24), we have

$$\begin{aligned}[Q^n_{t+1}]^2_+ &\leq \big[\max\{[Q^n_t]_+ + q^n - w^n_t d^n_t u^n_t, 0\}\big]^2\\ &\leq \big[[Q^n_t]_+ - (w^n_t d^n_t u^n_t - q^n)\big]^2.\end{aligned} \tag{31}$$

Rearranging the terms of (31), we have

$$\begin{aligned}&[Q^n_{t+1}]^2_+ - [Q^n_t]^2_+\\ &\quad\leq -2[Q^n_t]_+(w^n_t d^n_t u^n_t - q^n) + (w^n_t d^n_t u^n_t - q^n)^2.\end{aligned} \tag{32}$$

Taking conditional expectation over S_t and noting that $(w^n_t d^n_t u^n_t - q^n)^2 \leq C$ yields

$$\begin{aligned}&\mathbb{E}\big\{[Q^n_{t+1}]^2_+ - [Q^n_t]^2_+|S_t\big\}\\ &\quad\leq -2[Q^n_t]_+\big[\mathbb{E}\{w^n_t d^n_t u^n_t|S_t\} - q^n\big] + C.\end{aligned} \tag{33}$$

Substituting (30) and (33) into (29), we have

$$\begin{aligned}\mathbb{E}\{\Delta_t|S_t\} \leq &-U\sum_{n\in\mathcal{N}}\alpha^n\tau^n_t\mathbb{E}\big\{d^n_t u^n_t|S_t\big\} + U\sum_{n\in\mathcal{N}}\alpha^n\\ &-\sum_{n\in\mathcal{N}}[Q^n_t]_+\mathbb{E}\big\{w^n_t d^n_t u^n_t|S_t\big\} + \sum_{n\in\mathcal{N}}q^n[Q^n_t]_+ + \frac{1}{2}\sum_{n\in\mathcal{N}}C\end{aligned}$$

which proves (28). ∎

Note that B_t in (28) is not affected by the scheduling decision u^n_t. We solve the following per-slot optimization problem $\mathbf{P}_t$ to minimize the modified Lyapunov drift upper bound in Lemma 1

$$\begin{aligned}\mathbf{P}_t: \quad \min_{\{u^n_t\}} \quad &-\sum_{n\in\mathcal{N}}U\alpha^n\tau^n_t d^n_t u^n_t + w^n_t[Q^n_t]_+ d^n_t u^n_t\\ \text{s.t.} \quad &\sum_{n\in\mathcal{N}}u^n_t \leq K, \quad \forall t.\end{aligned} \tag{21c}$$

It is easy to see that the solution to $\mathbf{P}_t$ is selecting the top-K local devices with the highest value of

$$W^n_t \triangleq U\alpha^n\tau^n_t d^n_t + w^n_t[Q^n_t]_+ d^n_t. \tag{34}$$

This provides the desired scheduling policy.

We summarize the AIMWeL algorithm in Algorithms 1 and 2 at the server side and the local device side, respectively. We will further show that AIMWeL provides guarantees on both the long-term weight constraint and the optimality ratio in Sections IV-D and IV-E.

Algorithm 1 AIMWeL: central server's algorithm

1: Initialize $U>0$, $\{\alpha^n>0\}$, $\{\tau_1^n=1\}$, $\{Q_1^n=0\}$.
2: Receive $\{w_t^n\}$ and $\{d_t^n\}$ from devices $\mathcal{N}$.
3: Send $\{u_t^n=1\}$ to devices $\mathcal{N}_t$ with K-largest $\{W_t^n\}$ in (34).
4: Receive $\mathbf{x}_{l_t^n}^n$ from devices $\mathcal{N}_t$.
5: Update global decision $\mathbf{x}_{t+1}$ via (5).
6: Broadcast $\mathbf{x}_{t+1}$ to devices $\mathcal{N}_t$.
7: Update $\{\tau_{t+1}^n\}$ for devices $\mathcal{N}$ via (9).
8: Update $\{Q_{t+1}^n\}$ for devices $\mathcal{N}$ via (24).

Algorithm 2 AIMWeL: local device's algorithm

1: Initialize $\mathbf{x}_1=\mathbf{0}$.
2: Send w_t^n, d_t^n to central server.
3: **if** Received $\mathbf{x}_t$ **then**
4: Update local decision $\mathbf{x}_t^n$ via (4).
5: **else** Continue to update $\mathbf{x}_{l_t^n}^n$.
6: **if** $d_t^n u_t^n=1$ **then**
7: Upload $\mathbf{x}_{l_t^n}^n$ to central server.
8: Update d_{t+1}^n via (6).

Remark 2. *(Computational Complexity)* In each time slot, AIMWeL requires only the updating of N virtual queues, closed-form calculation of $\{W_t^n\}$, and finding of top K out of N values, with computational complexity $\mathcal{O}(N)$, $\mathcal{O}(N)$, and $\mathcal{O}(N+K\log K)$, respectively. Therefore, the overall computational complexity of AIMWeL is $\mathcal{O}(N+K\log K)$. It is in the *same* order of computational complexity as some simple huristic AoI minimization methods, *e.g.,* calculating N AoI values and then selecting K devices with the largest AoI.

Remark 3. *(Communication Cost)* In each time slot, the communication cost of AIMWeL is dominated by the uploading of at most K local decisions $\{\mathbf{x}_{l_t^n}^n\}$ from the scheduled local devices $\mathcal{N}_t$, and the receiving of an *equal* number of global decisions $\mathbf{x}_{t+1}$ from the central server. Thus, *both* the uplink and downlink communication cost of AIMWeL is $\mathcal{O}(Kd)$, with d being the number of decision parameters. Note that since d is usually large in modern machine learning applications, *e.g.,* it can be thousands for logistic regression in Section V-B and millions for neural network in Section V-C, the $\mathcal{O}(N)$ communication cost of uploading the function weights $\{w_t^n\}$ and communication indicators $\{d_t^n\}$ is negligible. Furthermore, the scalar values of $\{w_t^n\}$ and $\{d_t^n\}$ can be efficiently communicated over K uplink channels via time division.

D. Bound on Weight Constraints via Modified Lyapunov Drift

We bound the weight constraint violation by AIMWeL, via a modified Lyapunov drift analysis to provide an upper bound on the virtual queue. We require the following lemma, which states that for any n and t, d_t^n and u_t^n are uncorrelated.

Lemma 2. For any policy $\pi\in\Pi$, we have

$$\mathbb{E}\{d_t^n u_t^n\}=\mathbb{E}\{d_t^n\}\mathbb{E}\{u_t^n\},\quad \forall n,\forall t. \tag{35}$$

Proof: See Appendix C.

In the following theorem, we show AIMWeL guarantees strong stability of the virtual queues in (24) and thus satisfies the individual long-term weight constraints (21b) in **P**.

Theorem 2. The AIMWeL scheduling policy satisfies the individual long-term weight constraints (21b) in **P** for any strictly feasible $\{q^n\}$. Specifically, we have

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}\mathbb{E}\big\{[Q_t^n]_+\big\}\le\frac{1}{\sigma}\Big(U+\frac{CN}{2}\Big)<\infty. \tag{36}$$

Proof: Consider a stationary randomized policy π_{SR} with scheduling probabilities $\{v^n\}$. The modified Lyapunov drift is bounded by

$$\begin{aligned}\mathbb{E}\{\Delta_t|S_t\}&\overset{(a)}{\le}-\sum_{n\in\mathcal{N}}U\alpha^n\tau_t^n\mathbb{E}\{d_t^n v^n|S_t\}\\&\quad-\sum_{n\in\mathcal{N}}[Q_t^n]_+\mathbb{E}\{w_t^n d_t^n v^n-q^n|S_t\}+U+\frac{CN}{2}\\&\overset{(b)}{\le}-U\sum_{n\in\mathcal{N}}p_{\text{LB}}^n v^n\alpha^n\tau_t^n-\sum_{n\in\mathcal{N}}(\bar{w}^n p_{\text{LB}}^n v^n-q^n)[Q_t^n]_+\\&\quad+U+\frac{CN}{2}\end{aligned} \tag{37}$$

where (a) is because AIMWeL greedily minimizes the modified Lyapunov drift upper bound in (28) at every time slot t, and therefore any other policy $\pi\in\Pi$ yields a larger (or equal) right hand side (RHS) of (28); and (b) follows from the lower bound on $\mathbb{E}\{d_t^n\}$ in (7) and (35) in Lemma 2.

Taking the expectation of (37) over S_t, summing it over $t\in\mathcal{T}$, and then dividing it by T, we have

$$\text{LHS}_1+\text{LHS}_2\le-\frac{1}{T}\sum_{t\in\mathcal{T}}\mathbb{E}\{\Delta_t\}+U+\frac{CN}{2} \tag{38}$$

where

$$\text{LHS}_1\triangleq\frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}(\bar{w}^n p_{\text{LB}}^n v^n-q^n)\mathbb{E}\{[Q_t^n]_+\}, \tag{39}$$

and

$$\text{LHS}_2\triangleq\frac{U}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}p_{\text{LB}}^n v^n\alpha^n\mathbb{E}\{\tau_t^n\}. \tag{40}$$

Let $v^n=\tilde{v}^n$, where $\tilde{v}^n$ is defined in Section IV-B, and applying the Slater's condition (23) to LHS_1 in (39), we have

$$\text{LHS}_1\ge\frac{\sigma}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}\mathbb{E}\{[Q_t^n]_+\}. \tag{41}$$

Substituting it into (38), noting that

$$-\frac{1}{T}\sum_{t\in\mathcal{T}}\mathbb{E}\{\Delta_t\}\le\frac{\mathbb{E}\{L_1\}}{T}, \tag{42}$$

$$\text{LHS}_2\ge 0, \tag{43}$$

and taking the limit $T\to\infty$, we have

$$\lim_{T\to\infty}\frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}\mathbb{E}\{[Q_t^n]_+\}\le\lim_{T\to\infty}\frac{\mathbb{E}\{L_1\}}{T\sigma}+\frac{1}{\sigma}\Big(U+\frac{CN}{2}\Big).$$

Further noting that L_1 is upper bounded, we prove strong stability of the virtual queues in (36), which implies (21b) is satisfied. ■

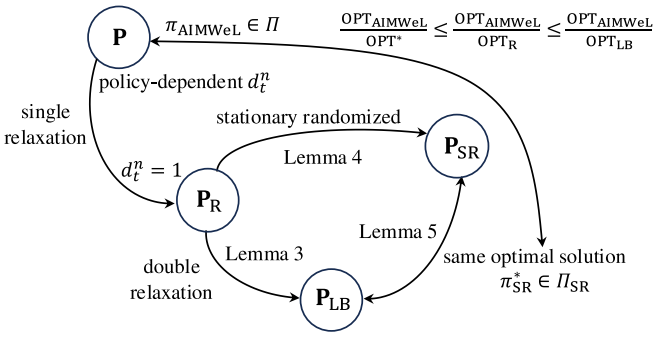


Fig. 2: An illustration of the double relxation approach to bound the optimality ratio of AIMWeL.

E. Bound on Optimality Ratio via Double Relaxation

We bound the optimality ratio of AIMWeL via a novel *double relaxation* approach, to handle the policy-dependent communication indicator sequence d_t^n caused by semi-asynchronous aggregation. As illustrated in Fig. 2, the intuition behind the double relaxation approach is to keep relaxing the original problem $\mathbf{P}$ (21), until the optimal stationary randomized policy $\pi_{\text{SR}}^\star \in \Pi_{\text{SR}}$ for solving a single-relaxed problem $\mathbf{P}_{\text{R}}$ (44) — the solution to the stationary randomized equivalent problem $\mathbf{P}_{\text{SR}}$ (48) — also achieves the optimal objective value of a doubly-relaxed lower bound problem $\mathbf{P}_{\text{LB}}$ (46). This stationary randomized policy $\pi_{\text{SR}}^\star$ builds a bridge, connecting AIMWeL and the optimal policy $\pi^\star$ to bound the optimality ratio.

1) Double Relaxation: Existing constrained AoI scheduling policies [26]-[30], [36], do not consider the case where the communication indicator sequence d_t^n is policy-dependent with a time-varying probability p_t^n of completing local decision update, so we cannot apply their techniques. Instead, we first use the upper bound on $\mathbb{E}\{d_t^n\}$ in (7) to relax $\mathbf{P}$ by setting $d_t^n = 1$, and then relax the deterministic constraint (21c) to the expectation form. This leads to the following relaxed problem of $\mathbf{P}$

$$\mathbf{P}_{\text{R}}: \quad \text{OPT}_{\text{R}} = \min_{\pi\in\Pi} \left\{ \lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \alpha^n \mathbb{E}\{\tau_t^n\} \right\} \tag{44a}$$

$$\text{s.t.} \quad \lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \bar{w}^n \mathbb{E}\{u_t^n\} \ge q^n, \quad \forall n, \tag{44b}$$

$$\sum_{n\in\mathcal{N}} \mathbb{E}\{u_t^n\} \le K. \tag{44c}$$

Let $\pi_{\text{R}} \in \Pi$ be an optimal online policy for $\mathbf{P}_{\text{R}}$, we have

$$\text{OPT}_{\text{R}} \le \text{OPT}^\star. \tag{45}$$

The following lemma provides a lower bound problem to $\mathbf{P}_{\text{R}}$, by relating the scheduling decision u_t^n with the AoI τ_t^n. Our proof extends the results of Theorem 1 in [26], which considers scheduling over a single i.i.d. interference channel, *i.e.,* at most $K = 1$ user can be scheduled at each time, to the more general case that $K \ge 1$ devices can be scheduled at each time.

Lemma 3. Any optimal online policy $\pi_{\text{LB}} \in \Pi$ for the following optimization problem provides a lower bound $\text{OPT}_{\text{LB}} \le \text{OPT}_{\text{R}}$ to $\mathbf{P}_{\text{R}}$:

$$\mathbf{P}_{\text{LB}}: \quad \text{OPT}_{\text{LB}} = \min_{\pi\in\Pi} \left\{ \frac{1}{2} \sum_{n\in\mathcal{N}} \frac{\alpha^n}{c^n} + \frac{1}{2} \right\} \tag{46a}$$

$$\text{s.t.} \quad \bar{w}^n c^n \ge q^n, \quad \forall n, \tag{46b}$$

$$\sum_{n\in\mathcal{N}} \mathbb{E}\{u_t^n\} \le K \tag{44c}$$

where $c^n \triangleq \lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \mathbb{E}\{u_t^n\}$ is the long-term time-averaged expected number of schedules for device n.

Proof: See Appendix D.

Thus, the original scheduling problem $\mathbf{P}$ can be relaxed *twice* to $\mathbf{P}_{\text{LB}}$. We will next show that $\mathbf{P}_{\text{LB}}$ admits a stationary randomized (*i.e.,* AoI-independent) policy that achieves its optimal objective value OPT_{LB}. We require the following lemma on the long-term time-averaged AoI for any stationary randomized policy when the communication indicator d_t^n is always 1. The proof proceeds by noting that the probability of receiving a local decision from device n is v^n and then applying renewal theory [37].

Lemma 4. Consider a stationary randomized policy π_{SR} with scheduling probabilities $\{v^n\}$. When $d_t^n = 1, \forall n, \forall t$, the long-term time-averaged expected AoI is

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \mathbb{E}\{\tau_t^n\} = \frac{1}{v^n}, \quad \forall n. \tag{47}$$

Proof: See Appendix E.

Applying Lemma 4 to problem $\mathbf{P}_{\text{R}}$, we have the following optimization problem $\mathbf{P}_{\text{SR}}$, where Π_{SR} denotes the class of stationary randomized algorithms

$$\mathbf{P}_{\text{SR}}: \quad \text{OPT}_{\text{SR}} = \min_{\pi\in\Pi_{\text{SR}}} \left\{ \sum_{n\in\mathcal{N}} \frac{\alpha^n}{v^n} \right\} \tag{48a}$$

$$\text{s.t.} \quad \bar{w}^n v^n \ge q^n, \quad \forall n, \tag{48b}$$

$$\sum_{n\in\mathcal{N}} v^n \le K. \tag{48c}$$

Let $\pi_{\text{SR}}^\star$ be an optimal stationary randomized policy that solves the above $\mathbf{P}_{\text{SR}}$. The following lemma shows that $\pi_{\text{SR}}^\star$ also achieves the optimal objective value of $\mathbf{P}_{\text{LB}}$. The proof follows from noting that $\pi_{\text{SR}}^\star$ is also a feasible solution to $\mathbf{P}_{\text{LB}}$, and then comparing OPT_{SR} with OPT_{LB} and the objective value of $\mathbf{P}_{\text{LB}}$ achieved by $\pi_{\text{SR}}^\star$.

Lemma 5. The scheduling probabilities of the optimal stationary randomized policy $\pi_{\text{SR}}^\star \in \Pi_{\text{SR}}$ for solving $\mathbf{P}_{\text{SR}}$ are $\{c_{\text{LB}}^n\}_{n\in\mathcal{N}}$, where c_{LB}^n is the long-term time-averaged expected number of schedules for device n under the optimal policy $\pi_{\text{LB}} \in \Pi$ that solves $\mathbf{P}_{\text{LB}}$. Furthermore, $\pi_{\text{SR}}^\star$ achieves the optimal objective value OPT_{LB} of $\mathbf{P}_{\text{LB}}$.

Proof: Consider a stationary randomized policy $\pi_{\text{SR}} \in \Pi_{\text{SR}}$ with scheduling probabilities $\{c_{\text{LB}}^n\}_{n\in\mathcal{N}}$. It follows that policy π_{SR} satisfies constraints (46b) and (44c) in $\mathbf{P}_{\text{LB}}$, and achieves the optimal objective value OPT_{LB} in (46a). Note that policy π_{SR} with scheduling probabilities $\{c_{\text{LB}}^n\}_{n\in\mathcal{N}}$ also satisfies constraints (48b) and (48c) in $\mathbf{P}_{\text{SR}}$, and thus is a feasible solution

to $\mathbf{P}_{\text{SR}}$. Let the objective value of $\mathbf{P}_{\text{SR}}$ achieved by policy π_{SR} be OBJ_{SR}. Comparing the object of $\mathbf{P}_{\text{LB}}$ in (46a) with the object of $\mathbf{P}_{\text{SR}}$ in (48a), we have

$$\frac{\text{OBJ}_{\text{SR}}}{2} < \text{OPT}_{\text{LB}}. \tag{49}$$

We now prove by contradiction that the optimal stationary randomized policy $\pi^{\star}_{\text{SR}}$ for solving $\mathbf{P}_{\text{SR}}$ is $\{c^n_{\text{LB}}\}_{n\in\mathcal{N}}$, *i.e.,*

$$\text{OBJ}_{\text{SR}} = \text{OPT}_{\text{SR}}. \tag{50}$$

Suppose there exists another stationary randomized policy with scheduling probabilities $\{\tilde{v}^n\}_{n\in\mathcal{N}}$ that satisfies constraints (48b) and (48c) in $\mathbf{P}_{\text{SR}}$, and achieves a lower objective value for $\mathbf{P}_{\text{SR}}$ than $\{c^n_{\text{LB}}\}_{n\in\mathcal{N}}$. From (49), it follows that the scheduling policy with $c^n = \tilde{v}^n, \forall n$ satisfies constraints (46b) and (44c) in $\mathbf{P}_{\text{LB}}$, and achieves a lower objective value for $\mathbf{P}_{\text{LB}}$ than π_{LB}. This contradicts to π_{LB} being the optimal scheduling policy for solving $\mathbf{P}_{\text{LB}}$. Therefore, the optimal stationary randomized policy $\pi^{\star}_{\text{SR}}$ for solving $\mathbf{P}_{\text{SR}}$ has scheduling probabilities $\{c^n_{\text{LB}}\}_{n\in\mathcal{N}}$, and thus (50) holds. ∎

2) Optimality Ratio of AIMWeL: The stationary randomized policy $\pi^{\star}_{\text{SR}} \in \Pi_{\text{SR}}$ connects AIMWeL and an optimal policy $\pi^{\star} \in \Pi$ for solving problem $\mathbf{P}$, to bound the optimality ratio of AIMWeL. The following theorem provides an optimality ratio of AIMWeL, where $\text{OPT}_{\text{AIMWeL}}$ is the objective of problem $\mathbf{P}$ achieved by AIMWeL.

Theorem 3. The optimality ratio yielded by AIMWeL is upper bounded by

$$\frac{\text{OPT}_{\text{AIMWeL}}}{\text{OPT}^{\star}} \leq \frac{c^{\max}_{\text{LB}}}{p^{\min}_{\text{LB}} c^{\min}_{\text{LB}}}\Big[2+\frac{CN}{U}\Big]\Big[\frac{(1-p^{\min}_{\text{LB}})q^{\max}}{\sigma}+1\Big] \tag{51}$$

where $c^{\min}_{\text{LB}} \triangleq \min_n\{c^n_{\text{LB}}\}$, $c^{\max}_{\text{LB}} \triangleq \max_n\{c^n_{\text{LB}}\}$, $p^{\min}_{\text{LB}} \triangleq \min_n\{p^n_{\text{LB}}\}$, and $q^{\max} \triangleq \max_n\{q^n\}$.

Proof: From the definition of LHS_1 (39) in the proof of Theorem 2, we have

$$\begin{aligned}\text{LHS}_1 &= \frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}(\bar{w}^n p^n_{\text{LB}} v^n - q^n)\mathbb{E}\{[Q^n_t]_+\}\\ &= \frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}} p^n_{\text{LB}}(\bar{w}^n v^n - q^n)\mathbb{E}\{[Q^n_t]_+\}\\ &\quad - \frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}(1-p^n_{\text{LB}})q^n\mathbb{E}\{[Q^n_t]_+\}.\end{aligned} \tag{52}$$

Substituting $v^n = c^n_{\text{LB}}$ into (52) and noting that $\bar{w}^n c^n_{\text{LB}} \geq q^n$, we have

$$\begin{aligned}-\text{LHS}_1 &= \frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}(1-p^n_{\text{LB}})q^n\mathbb{E}\{[Q^n_t]_+\}\\ &\quad - \frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}} p^n_{\text{LB}}(\bar{w}^n c^n_{\text{LB}} - q^n)\mathbb{E}\{[Q^n_t]_+\}\\ &\leq \frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}(1-p^n_{\text{LB}})q^n\mathbb{E}\{[Q^n_t]_+\}\\ &\leq \frac{(1-p^{\min}_{\text{LB}})q^{\max}}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}\mathbb{E}\{[Q^n_t]_+\}.\end{aligned} \tag{53}$$

Dividing both sides of (38) in the proof of Theorem 2 by U, taking $T\to\infty$, and rearranging terms, we have

$$\begin{aligned}&\lim_{T\to\infty}\frac{\text{LHS}_2}{U} = \lim_{T\to\infty}\frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}} p^n_{\text{LB}} c^n_{\text{LB}}\alpha^n\mathbb{E}\{\tau^n_t\}\\ &\leq -\lim_{T\to\infty}\frac{\text{LHS}_1}{U} - \lim_{T\to\infty}\frac{1}{UT}\sum_{t\in\mathcal{T}}\mathbb{E}\{\Delta_t\} + 1 + \frac{CN}{2U}\\ &\overset{(a)}{\leq} \frac{(1-p^{\min}_{\text{LB}})q^{\max}}{U}\lim_{T\to\infty}\frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}\mathbb{E}\{[Q^n_t]_+\} + \frac{1}{2}\Big[2+\frac{CN}{U}\Big]\\ &\overset{(b)}{\leq} \frac{(1-p^{\min}_{\text{LB}})q^{\max}}{2\sigma}\Big[2+\frac{CN}{U}\Big] + \frac{1}{2}\Big[2+\frac{CN}{U}\Big]\\ &= \frac{1}{2}\Big[2+\frac{CN}{U}\Big]\Big[\frac{(1-p^{\min}_{\text{LB}})q^{\max}}{\sigma}+1\Big]\end{aligned} \tag{54}$$

where (a) follows from the bound on $-\sum_{t\in\mathcal{T}}\mathbb{E}\{\Delta_t\}$ in (42) and the bound on $-\text{LHS}_1$ in (53), and (b) follows from the virtual queue upper bound in (36). Dividing both sides of (54) by $p^{\min}_{\text{LB}} c^{\min}_{\text{LB}}$, we have

$$\begin{aligned}\text{OPT}_{\text{AIMWeL}} &= \lim_{T\to\infty}\frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}\alpha^n\mathbb{E}\{\tau^n_t\}\\ &\leq \frac{1}{2p^{\min}_{\text{LB}}c^{\min}_{\text{LB}}}\Big[2+\frac{CN}{U}\Big]\Big[\frac{(1-p^{\min}_{\text{LB}})q^{\max}}{\sigma}+1\Big].\end{aligned} \tag{55}$$

From the objective (46a) in $\mathbf{P}_{\text{LB}}$, we have

$$\text{OPT}_{\text{LB}} = \frac{1}{2}\sum_{n\in\mathcal{N}}\frac{\alpha^n}{c^n_{\text{LB}}} + \frac{1}{2} \geq \frac{\sum_{n\in\mathcal{N}}\alpha^n}{2c^{\max}_{\text{LB}}} + \frac{1}{2} \geq \frac{1}{2c^{\max}_{\text{LB}}}. \tag{56}$$

Comparing (55) with (56), we have

$$\frac{\text{OPT}_{\text{AIMWeL}}}{\text{OPT}_{\text{LB}}} \leq \frac{c^{\max}_{\text{LB}}}{p^{\min}_{\text{LB}}c^{\min}_{\text{LB}}}\Big[2+\frac{CN}{U}\Big]\Big[\frac{(1-p^{\min}_{\text{LB}})q^{\max}}{\sigma}+1\Big]. \tag{57}$$

Further noting that $\text{OPT}_{\text{LB}} \leq \text{OPT}_{\text{R}} \leq \text{OPT}^{\star}$, we have

$$\frac{\text{OPT}_{\text{AIMWeL}}}{\text{OPT}^{\star}} \leq \frac{\text{OPT}_{\text{AIMWeL}}}{\text{OPT}_{\text{R}}} \leq \frac{\text{OPT}_{\text{AIMWeL}}}{\text{OPT}_{\text{LB}}}. \tag{58}$$

Combining (56) and (58), we complete the proof. ∎

The following corollary provides an upper bound on the optimality ratio yielded by AIMWeL for the special case where the communication indicator sequence d^n_t is i.i.d. and is independent of the scheduling decision sequence u^n_t.

Corollary 2. For i.i.d. communication indicator sequence d^n_t that is independent of the scheduling decision sequence u^n_t, AIMWeL provides an optimality ratio

$$\frac{\text{OPT}_{\text{AIMWeL}}}{\text{OPT}^{\star}} \leq \frac{\tilde{c}^{\max}_{\text{LB}}}{\tilde{c}^{\min}_{\text{LB}}}\Big[2+\frac{CN}{U}\Big] \tag{59}$$

where $\tilde{c}^{\max}_{\text{LB}} \triangleq \max_n\{\tilde{c}^n_{\text{LB}}\}$ and $\tilde{c}^{\min}_{\text{LB}} \triangleq \min_n\{\tilde{c}^n_{\text{LB}}\}$ with $\tilde{c}^n_{\text{LB}}$ being the long-term time-averaged expected number of schedules for device n achieved by the optimal policy $\tilde{\pi}_{\text{LB}}$ for solving a lower bound problem $\widetilde{\mathbf{P}}_{\text{LB}}$ to $\mathbf{P}$.

Proof: See Appendix F.

As a point of comparison, the drift-plus-penalty (DPP) policy proposed in [26] for device scheduling over a single $K = 1$ i.i.d. interference channel and a policy-independent communication indicator sequence, is shown to be 2-optimal. In Corollary 2, the additional constant $\frac{\tilde{c}^{\max}_{\text{LB}}}{\tilde{c}^{\min}_{\text{LB}}}$ is caused by

scheduling *multiple* $K \geq 1$ devices at each time t. As the network heterogeneity reduces and U is set large enough, AIMWeL becomes 2-optimal.

Remark 4. (*Comparison with Existing Works*) As observed earlier, the assumption of i.i.d. and policy-independent d_t^n is a highly-simplified case that is nevertheless common in existing works [26]-[30], [36]. Specifically, the policies proposed in [26] are for device scheduling over a single i.i.d. interference channel with fixed transmission success probabilites at the devices, *i.e.,* $K = 1$ and $p_t^n = p^n, \forall t$, while we consider scheduling *multiple* $K > 1$ devices with *time-varying* probabilities p_t^n of finishing their local updates at each time t. Furthermore, different from the standard quadratic only Lyapunov function used in [26], we use the weighted sum of *linear* AoI values and *quadratic* virtual queues as a new Lyapunov function, to handle the *policy-dependent* communication indicator sequence. In addition, we propose a novel *double relaxation* approach to bound the optimality ratio of AIMWeL with policy-dependent and thus *non-i.i.d.* communication indicator sequence, which is also different from the bounding approach in [26] for i.i.d. systems. Finally, we emphasize here that AIMWeL together with its performance analysis is applicable to general constrained AoI scheduling problems with *policy-dependent* communication states under *time-varying* state transition probabilities.

V. Application to Semi-Asynchronous FL

As an example to study the performance of AIMWeL, we apply it to semi-asynchronous FL. We present numerical results to demonstrate the performance advantage of AIMWeL over the current best alternatives, based on standard image classification datasets for both logistic regression and neural network training.

A. Simulation Setup

We consider a FL system with $N = 10$ devices and a server. We evaluate our results on the popular MNIST dataset [38] and Fashion-MNIST dataset [39]. Each of their training dataset $\mathcal{D}$ consists of 6×10^4 data samples, and their test dataset $\mathcal{E}$ consists of 1×10^4 data samples. Each data sample $(\boldsymbol{\mu}, \nu)$ represents an image with 28×28 pixels and 10 possible labels, *i.e.,* $\boldsymbol{\mu} \in \mathbb{R}^{784}$ and $\nu \in \{1, \ldots, 10\}$. We study the scenario where the local datasets $\{\mathcal{B}_t^n\}$ contain data samples of different labels among devices, such that the data is *non-i.i.d.* We consider *unbalanced* and *streaming* data at the devices, and set the number of arriving data samples β_t^n as uniformly distributed $U[1,4]$ for $n \in \{1,2\}$, $U[1,5]$ for $n \in \{3,4,5\}$, $U[6,8]$ for $n \in \{6,7\}$, and $U[6,10]$ for $n \in \{8,9,10\}$. We also consider network heterogeneity in computational capacity and set $\bar{p}^n = \frac{3}{4}$ for $n \in \{1, \ldots, 5\}$, and $\bar{p}^n = \frac{1}{4}$ for $n \in \{6, \ldots, 10\}$. We set equal AoI scaling factor $\alpha^n = \frac{1}{10}$ for each device. All programming codes will be provided along with the final version of this paper.

We compare AIMWeL with the following schemes.

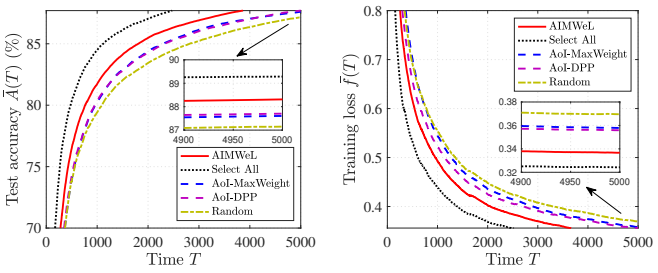
- *Select All*: The server schedules all devices that are ready to upload their local models $\{\mathbf{x}_{l_t^n}^n\}$. It represents the *idealized* aggregation scenario without any limits on the number of participating devices, *i.e.,* $K = N$.
- *AoI-MaxWeight:* The server selects K devices with the largest weight $U\frac{\alpha^n p^n}{2}\tau_t^n(\tau_t^n + 1) + p^n[Q_t^n]_+$. It is a modification of the max weight policy in [26] to handle multi-user scheduling over i.i.d. channels. Also, we use the same virtual queue updating rule in (24) for the max weight policy to take into account the local weights w_t^n.
- *AoI-DPP:* The server selects K devices with the largest DPP value $U\frac{\alpha^n p^n}{2}\tau_t^n + p^n[Q_t^n]_+$. It represents a Lyapunov optimization based scheduling policy similar to the one in [26]. The DPP policy has been extended to also use the same virtual queue updating rule in (24) as AIMWeL.
- *Random:* The server randomly schedules up to K devices that are ready to upload their local models. This policy is commonly adopted by existing works on semi-asynchronous FL [13]-[15].

We detail the adaptations needed to apply the AoI-MaxWeight and AoI-DPP methods of [26] to solve our problem.

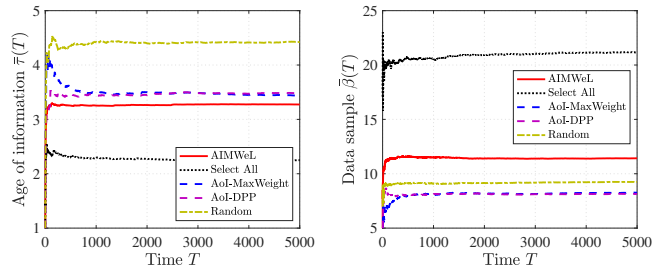
- AoI-MaxWeight and AoI-DPP are designed for scheduling over a single i.i.d. interference channel, *i.e.,* $K = 1$ and d_t^n is i.i.d. In contrast, AIMWeL is for multi-user scheduling, *i.e.,* $K \geq 1$, over policy-dependent and time-varying communication environment, *i.e.,* the communication indicator sequence $\{d_t^n\}$ depends on the scheduling decision sequence $\{u_t^n\}$ and has a time-varying and unknown probability p_t^n of being 1. To run AoI-MaxWeight and AoI-DPP in our simulation, we have extended them to multi-user scheduling and assume $\mathbb{P}\{d_t^n = 1\} = \bar{p}^n$ is constant and known.
- AoI-MaxWeight and AoI-DPP do not consider time-varying weights w_t^n in the long-term constraints. Here we have used the same virtual queue updating rule in (24) as AIMWeL for AoI-MaxWeight and AoI-DPP to take into account the local weights. Note that in this work we have used a novel double relaxation approach to show that AIMWeL provides a bounded optimality ratio of the weighted sum AoI and satisfies the individual long-term weight constraints. In contrast, AoI-MaxWeight and AoI-DPP do not provide any performance guarantee to our problem.

B. Convex Logistic Regression

We consider the cross-entropy loss for multinomial logistic regression, given by $l(\mathbf{x}; \boldsymbol{\mu}, \nu) = -\sum_{j \in \mathcal{V}} 1\{\nu = j\} \log \frac{\exp(\langle \mathbf{x}[j], \boldsymbol{\mu} \rangle)}{\sum_{k \in \mathcal{V}} \exp(\langle \mathbf{x}[k], \boldsymbol{\mu} \rangle)}$, where $\mathbf{x} = [\mathbf{x}[1]^T, \ldots, \mathbf{x}[10]^T]^T$ with $\mathbf{x}[j] \in \mathbb{R}^{784}$ being the model for label j. The entire model is thus of dimension $d = 7,840$. Our computation performance metrics are the time-averaged test accuracy over $\mathcal{E}$ given by $\bar{A}(T) = \frac{1}{|\mathcal{E}|T} \sum_{t \in \mathcal{T}} \sum_{i \in \mathcal{E}} 1\{\arg\max_j \{\log \frac{\exp(\langle \mathbf{x}_t[j], \boldsymbol{\mu}^i \rangle)}{\sum_{k \in \mathcal{V}} \exp(\langle \mathbf{x}_t[k], \boldsymbol{\mu}^i \rangle)}\} = \nu^i\}$ and the time-averaged training loss over $\{\mathcal{B}_t^n\}$ given by $\bar{f}(T) = \frac{1}{T} \sum_{t \in \mathcal{T}} \sum_{n \in \mathcal{N}} \frac{w_t^n}{\beta_t^n} \sum_{i \in \mathcal{B}_t^n} l(\mathbf{x}_t; \boldsymbol{\mu}_t^{n,i}, \nu_t^{n,i})$. Our scheduling performance metrics are the time-averaged weighted sum of AoI given by $\bar{\tau}(T) = \frac{1}{T} \sum_{t \in \mathcal{T}} \sum_{n \in \mathcal{N}} \alpha^n \tau_t^n$

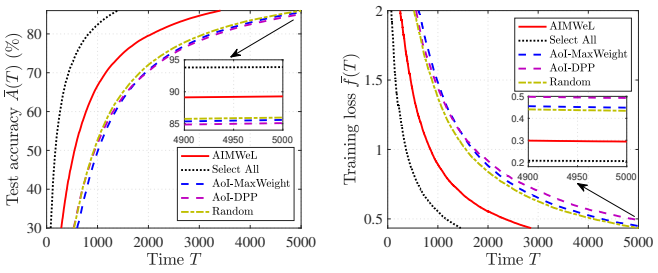


(a) $\bar{A}(T)$ and $\bar{f}(T)$ vs. T.

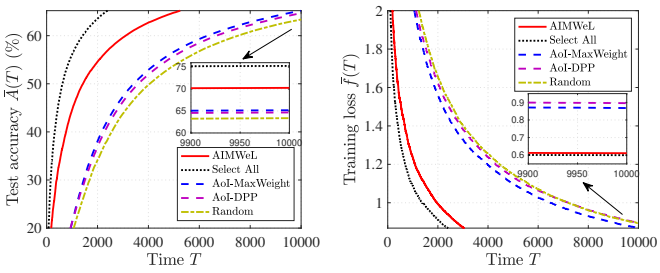


(b) $\bar{\tau}(T)$ and $\bar{\beta}(T)$ vs. T.

Fig. 3: Test accuracy $\bar{A}(T)$, training loss $\bar{f}(T)$, AoI $\bar{\tau}(T)$, and data sample $\bar{\beta}(T)$ vs. T for logistic regression on MNIST.



(a) $\bar{A}(T)$ and $\bar{f}(T)$ vs. T on MNIST.



(b) $\bar{A}(T)$ and $\bar{f}(T)$ vs. T on Fashion-MNIST.

Fig. 4: $\bar{A}(T)$ and $\bar{f}(T)$ vs. time T for neural network training on MNIST and Fashion-MNIST. The plots for $\bar{\tau}(T)$ and $\bar{\beta}(T)$ are not included since they are similar to Fig. 3b.

and the time-averaged total number of data samples given by $\bar{\beta}(T) = \frac{1}{T}\sum_{t\in\mathcal{T}}\sum_{n\in\mathcal{N}}\beta_t^n d_t^n u_t^n$.[4]

Fig. 3 shows $\bar{A}(T)$, $\bar{f}(T)$, $\bar{\tau}(T)$, and $\bar{\beta}(T)$ versus time T over MNIST. We set the learning rate $\eta = 1\times 10^{-5}$, $U = 1$, $q^n = \frac{\beta^n p^n K}{N}$, and $K = 2$. We see that the learning performance yielded by AIMWeL outperforms AoI-MaxWeight, AoI-DPP, and Random, and is close to the idealized Select All. AoI-MaxWeight and AoI-DPP achieve nearly the same performance, which is consistent with the simulation results in [26]. We observe that AIMWeL incurs over 20% less time to reach the same 88% accuracy that AoI-MaxWeight and AoI-DPP reach at the end. Furthermore, AIMWeL yields lower averaged AoI at the same time processes more data samples than AoI-MaxWeight and AoI-DPP.

C. Non-Convex Neural Network Training

To further validate the performance of AIMWeL for non-convex loss functions, we train a convolutional neural network, with a convolutional layer with 10 filters each of size 9×9, a ReLU hidden layer with 100 neurons, and a softmax output layer. The total number of model parameters is $d = 101,810$. We set the learning rate $\eta = 0.1$. Under the same settings as Fig. 3, we compare the test accuracy $\bar{A}(T)$ and the training loss $\bar{f}(T)$ among AIMWeL, Select All, AoI-MaxWeight, AoI-DPP, and Random on both MNIST and Fashion-MNIST in Fig. 4. The AoI $\bar{\tau}(T)$ and data sample $\bar{\beta}(T)$ plots are not included since they are similar to the ones in Fig. 3. We see that AoI-MaxWeight and AoI-DPP have nearly the same learning performance as Random. AIMWeL substantially outperforms AoI-MaxWeight, AoI-DPP, and Random, incurring over 25% less time to reach the same accuracy as these policies on MNIST, and over 50% less time on Fashion-MNIST.

[4]As explained in Section III-A, the local weight is commonly set as $w_t^n = \frac{\beta_t^n}{\beta_t}, \forall n\in\mathcal{N}$ for FL. In our simulation, we replace w_t^n with β_t^n, and use the time averaged number of data samples as equivalent constraint for FL.

VI. Conclusions

We consider device scheduling for semi-asynchronous aggregation in online distributed optimization. We propose an efficient AIMWeL scheduling policy via a modified Lyapunov drift design that uses the weighted sum of linear AoI values and quadratic virtual queues as a new Lyapunov function, to minimize the accumulated AoI on the local decision updates, under both individual long-term weight constraints and a number of devices constraint. Through a novel double relaxation approach to decouple the dependency between the communication indicator sequence and the scheduling decision sequence under time-varying probabilities of completing local decision updates due to semi-asynchronous aggregation, we show that AIMWeL provides guaranteed optimality ratio and no long-term weight constraint violation. When applying AIMWeL to semi-asynchronous FL, our experimental results demonstrate substantial performance advantage of AIMWeL over the current best approaches, in terms of both improved final classification accuracy and reduced training time for both convex and non-convex loss functions.

Appendix A
Proof of Theorem 1

Proof: Our proof consists of six major steps.

Step 1: Bound on $D_{t+1} \triangleq f_{t+1}(\mathbf{x}_{t+1}) - f_{t+1}(\mathbf{x}_{t+1}^\star)$. We have

$$D_{t+1} \overset{(a)}{=} f_{t+1}\Big(\sum_{n\in\mathcal{N}_t} w_t^n \mathbf{x}_{l_t^n}^n + \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m \mathbf{x}_t\Big) - f_{t+1}(\mathbf{x}_{t+1}^\star)$$

$$\overset{(b)}{\leq} \sum_{n\in\mathcal{N}_t}\big[w_t^n f_{t+1}(\mathbf{x}_{l_t^n}^n)\big] + \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t}\big[w_t^m f_{t+1}(\mathbf{x}_t)\big] - f_{t+1}(\mathbf{x}_{t+1}^\star)$$
$$\overset{(c)}{=} \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m\big[f_{t+1}(\mathbf{x}_t) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big]$$
$$+ \underbrace{\sum_{n\in\mathcal{N}_t} w_t^n\big[f_{t+1}(\mathbf{x}_{l_t^n}^n) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big]}_{\triangleq A_1}$$
$$\overset{(d)}{=} \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m \underbrace{\big[f_t(\mathbf{x}_t) - f_t(\mathbf{x}_t^\star)\big]}_{=D_t}$$
$$+ \underbrace{\sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m\big[f_{t+1}(\mathbf{x}_t) - f_t(\mathbf{x}_t)\big]}_{\triangleq E_1}$$
$$+ \underbrace{\sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m\big[f_t(\mathbf{x}_t^\star) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big]}_{\triangleq E_2} + A_1$$
$$= \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m D_t + E_1 + E_2 + A_1 \tag{60}$$

where (a) follows from the global decision updating rule in (5) and $l_t^n = t+1-\tau_t^n$ in (8), (b) is because of the convexity of $f_{t+1}(\mathbf{x})$, (c) is because the sum of the local weights satisfies $\sum_{n\in\mathcal{N}} w_t^n = 1$, and (d) follows from

$$f_{t+1}(\mathbf{x}_t) - f_{t+1}(\mathbf{x}_{t+1}^\star) = \big[f_t(\mathbf{x}_t) - f_t(\mathbf{x}_t^\star)\big] + \big[f_{t+1}(\mathbf{x}_t) - f_t(\mathbf{x}_t)\big] + \big[f_t(\mathbf{x}_t^\star) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big].$$

Step 2: Bound on A_1 ***in (60).*** We have

$$A_1 = \sum_{n\in\mathcal{N}_t} w_t^n\big[f_{t+1}(\mathbf{x}_{l_t^n}^n) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big]$$
$$\overset{(a)}{=} \sum_{n\in\mathcal{N}_t} w_t^n \underbrace{\big[f_{l_t^n}(\mathbf{x}_{l_t^n}) - f_{l_t^n}(\mathbf{x}_{l_t^n}^\star)\big]}_{=D_{l_t^n}}$$
$$+ \underbrace{\sum_{n\in\mathcal{N}_t} w_t^n\big[f_{l_t^n}(\mathbf{x}_{l_t^n}^\star) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big]}_{\triangleq E_3}$$
$$+ \sum_{n\in\mathcal{N}_t} w_t^n\big[f_{t+1}(\mathbf{x}_{l_t^n}^n) - f_{l_t^n}(\mathbf{x}_{l_t^n})\big]$$
$$\overset{(b)}{=} \sum_{n\in\mathcal{N}_t} w_t^n D_{l_t^n} + E_3 + \underbrace{\sum_{n\in\mathcal{N}_t} w_t^n\big[f_{t+1}(\mathbf{x}_{l_t^n}^n) - f_{l_t^n}(\mathbf{x}_{l_t^n}^n)\big]}_{\triangleq E_4}$$
$$+ \underbrace{\sum_{n\in\mathcal{N}_t} w_t^n\big[f_{l_t^n}(\mathbf{x}_{l_t^n}^n) - f_{l_t^n}(\mathbf{x}_{l_t^n})\big]}_{\triangleq A_2}$$
$$= \sum_{n\in\mathcal{N}_t} w_t^n D_{l_t^n} + E_3 + E_4 + A_2 \tag{61}$$

where (a) follows from

$$f_{t+1}(\mathbf{x}_{l_t^n}^n) - f_{t+1}(\mathbf{x}_{t+1}^\star) = [f_{l_t^n}(\mathbf{x}_{l_t^n}) - f_{l_t^n}(\mathbf{x}_{l_t^n}^\star)] + [f_{l_t^n}(\mathbf{x}_{l_t^n}^\star) - f_{t+1}(\mathbf{x}_{t+1}^\star)] + \big[f_{t+1}(\mathbf{x}_{l_t^n}^n) - f_{l_t^n}(\mathbf{x}_{l_t^n})\big],$$

and (b) is because

$$f_{t+1}(\mathbf{x}_{l_t^n}^n) - f_{l_t^n}(\mathbf{x}_{l_t^n}) = \big[f_{t+1}(\mathbf{x}_{l_t^n}^n) - f_{l_t^n}(\mathbf{x}_{l_t^n}^n)\big] + \big[f_{l_t^n}(\mathbf{x}_{l_t^n}^n) - f_{l_t^n}(\mathbf{x}_{l_t^n})\big].$$

Step 3: Bound on A_2 ***in (61).*** We have

$$A_2 = \sum_{n\in\mathcal{N}_t} w_t^n\big[f_{l_t^n}(\mathbf{x}_{l_t^n}^n) - f_{l_t^n}(\mathbf{x}_{l_t^n})\big]$$
$$\overset{(a)}{\leq} \sum_{n\in\mathcal{N}_t} w_t^n\big\langle\nabla f_{l_t^n}(\mathbf{x}_{l_t^n}), \mathbf{x}_{l_t^n}^n - \mathbf{x}_{l_t^n}\big\rangle + \frac{L}{2}\sum_{n\in\mathcal{N}_t} w_t^n\|\mathbf{x}_{l_t^n}^n - \mathbf{x}_{l_t^n}\|^2$$
$$\overset{(b)}{=} -\eta\sum_{n\in\mathcal{N}_t} w_t^n\big\langle\nabla f_{l_t^n}(\mathbf{x}_{l_t^n}), \nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n})\big\rangle + \underbrace{\eta^2 L\sum_{n\in\mathcal{N}_t}\frac{w_t^n}{2}\|\nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n})\|^2}_{\triangleq A_3}$$
$$\overset{(c)}{\leq} -\eta\epsilon\sum_{n\in\mathcal{N}_t} w_t^n\|\nabla f_{l_t^n}(\mathbf{x}_{l_t^n})\|^2 + A_3 \tag{62}$$

where (a) follows from $f_t(\mathbf{x})$ being L-smooth in (13), (b) is because the local decision updating rule in (4) that $\mathbf{x}_{l_t^n}^n = \mathbf{x}_{l_t^n} - \eta\nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n})$, and (c) follows from (14) in Assumption 4.

Step 4: Bound on A_3 ***in (62).*** We require the following lemma, which is borrowed from Lemma 1 in [10].

Lemma 6 (Lemma 1, [10])**.** For a μ-strongly convex loss function $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$, if $f(\mathbf{x}^\star) > -\infty$ where $\mathbf{x}^\star \in \arg\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x})$, we have for any $\mathbf{x} \in \mathbb{R}^d$

$$\langle\nabla f(\mathbf{x}), \mathbf{x} - \mathbf{x}^\star\rangle \leq f(\mathbf{x}) - f(\mathbf{x}^\star) \leq \frac{\|\nabla f(\mathbf{x})\|^2}{2\mu}. \tag{63}$$

Also, the co-coercivity of a L-smooth function $f(\mathbf{x}) : \mathbb{R}^d \to \mathbb{R}$ implies that for any $\mathbf{x}, \mathbf{y} \in \mathbb{R}^d$

$$\|\nabla f(\mathbf{x}) - \nabla f(\mathbf{y})\|^2 \leq L\langle\nabla f(\mathbf{x}) - \nabla f(\mathbf{y}), \mathbf{x} - \mathbf{y}\rangle. \tag{64}$$

We now bound A_3 on the RHS of (62). We have

$$A_3 = \eta^2 L\sum_{n\in\mathcal{N}_t}\frac{w_t^n}{2}\|\nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n})\|^2$$
$$\overset{(a)}{\leq} \eta^2 L\sum_{n\in\mathcal{N}_t} w_t^n\|\nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n}) - \nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n}^\star)\|^2 + \underbrace{\eta^2 L\sum_{n\in\mathcal{N}_t} w_t^n\|\nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n}^\star)\|^2}_{\triangleq E_5}$$
$$\overset{(b)}{\leq} \eta^2 L^2\sum_{n\in\mathcal{N}_t} w_t^n\big\langle\nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n}) - \nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n}^\star), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star\big\rangle + E_5$$
$$= \eta^2 L^2\sum_{n\in\mathcal{N}_t} w_t^n\Big[\big\langle\nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n}), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star\big\rangle - \big\langle\nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n}^\star), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star\big\rangle\Big] + E_5$$
$$= \eta^2 L^2\sum_{n\in\mathcal{N}_t} w_t^n\sum_{m\in\mathcal{N}} w_t^m\Big[\big\langle\nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star\big\rangle - \big\langle\nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}^\star), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star\big\rangle\Big] + E_5$$
$$- \eta^2 L^2\sum_{n\in\mathcal{N}_t} w_t^n\sum_{m\in\mathcal{N},m\neq n} w_t^m\Big[\big\langle\nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star\big\rangle$$

$$
\begin{aligned}
&\quad - \big\langle \nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}^\star), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star \big\rangle \Big] \\
&\overset{(c)}{=} \eta^2 L^2 \sum_{n\in\mathcal{N}_t} w_t^n \big\langle \nabla f_{l_t^n}(\mathbf{x}_{l_t^n}), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star \big\rangle \\
&\quad - \eta^2 L^2 \sum_{n\in\mathcal{N}_t} \langle \nabla f_{l_t^n}(\mathbf{x}_{l_t^n}^\star), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star \rangle + E_5 \\
&\quad - \eta^2 L^2 \sum_{n\in\mathcal{N}_t} w_t^n \sum_{m\in\mathcal{N},m\neq n} w_t^m \Big[\big\langle \nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}) \\
&\quad - \nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}^\star), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star \big\rangle \Big] \\
&\overset{(d)}{\leq} \eta^2 L^2 \sum_{n\in\mathcal{N}_t} w_t^n \big\langle \nabla f_{l_t^n}(\mathbf{x}_{l_t^n}), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star \big\rangle \\
&\quad - \eta^2 L^2 \sum_{n\in\mathcal{N}_t} \langle \nabla f_{l_t^n}(\mathbf{x}_{l_t^n}^\star), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star \rangle + E_5 \\
&\overset{(e)}{\leq} \frac{\eta^2 L^2}{2\mu} \sum_{n\in\mathcal{N}_t} w_t^n \|\nabla f_{l_t^n}(\mathbf{x}_{l_t^n})\|^2 + E_5 \qquad (65)
\end{aligned}
$$

where (a) is because $\frac{1}{2}\|\mathbf{a}+\mathbf{b}\|^2 \leq \|\mathbf{a}\|^2 + \|\mathbf{b}\|^2$, (b) follows from the co-coercivity of the L-smooth function in (64), (c) is because $f_t(\mathbf{x}) = \sum_{n\in\mathcal{N}} f_t^n(\mathbf{x})$ in (2), (d) follows from the co-coercivity of $f_t(\mathbf{x})$ in (64) such that

$$
\begin{aligned}
&- L\big\langle \nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}) - \nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}^\star), \mathbf{x}_{l_t^n} - \mathbf{x}_{l_t^n}^\star \big\rangle \\
&\quad \leq -\|\nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}) - \nabla f_{l_t^n}^m(\mathbf{x}_{l_t^n}^\star)\|^2 \leq 0,
\end{aligned}
$$

and (e) follows from applying (63) in Lemma 6 and $\nabla f_t(\mathbf{x}_t^\star) = \mathbf{0}$ in (10) of Assumption 1.

Substituting the bounds on A_1, A_2, A_3 in (61), (62), (65) into the bound on D_{t+1} in (60), we have

$$
\begin{aligned}
D_{t+1} &\leq \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m D_t + \sum_{n\in\mathcal{N}_t} w_t^n D_{l_t^n} \\
&\quad - \Big(\eta\epsilon - \frac{\eta^2 L^2}{2\mu}\Big) \sum_{n\in\mathcal{N}_t} w_t^n \|\nabla f_{l_t^n}(\mathbf{x}_{l_t^n})\|^2 \\
&\quad + E_1 + E_2 + E_3 + E_4 + E_5 \\
&\overset{(a)}{\leq} \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m D_t + \sum_{n\in\mathcal{N}_t} w_t^n \big[1 - \eta(2\mu\epsilon - \eta L^2)\big] D_{l_t^n} \\
&\quad + E_1 + E_2 + E_3 + E_4 + E_5 \qquad (66)
\end{aligned}
$$

where (a) follows from $\eta < \frac{2\mu\epsilon}{L^2}$ such that $\eta\epsilon - \frac{\eta^2 L^2}{2\mu} > 0$ and applying (63) in Lemma 6 again such that

$$
-\|\nabla f_{l_t^n}(\mathbf{x}_{l_t^n})\|^2 \leq -2\mu\big[f_{l_t^n}(\mathbf{x}_{l_t^n}) - f_{l_t^n}(\mathbf{x}_{l_t^n}^\star)\big] = -2\mu D_{l_t^n}.
$$

Step 5: Bound on $E_1 + E_2 + E_3 + E_4 + E_5$ ***in (66).*** We first bound $E_1 + E_4$ as

$$
\begin{aligned}
E_1 + E_4 &= \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m \big[f_{t+1}(\mathbf{x}_t) - f_t(\mathbf{x}_t)\big] \\
&\quad + \sum_{n\in\mathcal{N}_t} w_t^n \big[f_{t+1}(\mathbf{x}_{l_t^n}^n) - f_{l_t^n}(\mathbf{x}_{l_t^n}^n)\big] \\
&= \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m \big[f_{t+1}(\mathbf{x}_t) - f_t(\mathbf{x}_t)\big] \\
&\quad + \sum_{n\in\mathcal{N}_t} w_t^n \sum_{i=1}^{\tau_t^n} \big[f_{t+2-i}(\mathbf{x}_{l_t^n}^n) - f_{t+1-i}(\mathbf{x}_{l_t^n}^n)\big] \\
&\overset{(a)}{\leq} \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m \Delta f_{\text{UB}} + \tau_t^n \sum_{n\in\mathcal{N}_t} w_t^n \Delta f_{\text{UB}} \overset{(b)}{\leq} \tau_{\text{UB}} \Delta f_{\text{UB}} \qquad (67)
\end{aligned}
$$

where (a) follows from the definition of Δf_{UB} in (15) and (b) is because of the definition of τ_{UB} under (19) and $\sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m + \sum_{n\in\mathcal{N}_t} w_t^n = 1$.

We then bound $E_2 + E_3$ as

$$
\begin{aligned}
E_2 + E_3 &= \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m \big[f_t(\mathbf{x}_t^\star) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big] \\
&\quad + \sum_{n\in\mathcal{N}_t} w_t^n \big[f_{l_t^n}(\mathbf{x}_{l_t^n}^\star) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big] \\
&= \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m \big[f_t(\mathbf{x}_t^\star) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big] \\
&\quad + \sum_{n\in\mathcal{N}_t} w_t^n \sum_{i=1}^{\tau_t^n} \big[f_{t+1-i}(\mathbf{x}_{t+1-i}^\star) - f_{t+2-i}(\mathbf{x}_{t+2-i}^\star)\big] \\
&\overset{(a)}{\leq} \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m \Delta f_{\text{UB}} + \tau_t^n \sum_{n\in\mathcal{N}_t} w_t^n \Delta f_{\text{UB}} \overset{(b)}{\leq} \tau_{\text{UB}} \Delta f_{\text{UB}} \qquad (68)
\end{aligned}
$$

where (a) follows from $f_t(\mathbf{x})$ bing convex, $\nabla f_t(\mathbf{x}_t^\star) = \mathbf{0}$ in (10), and the definitions of Δf_{UB} in (15), such that

$$
\begin{aligned}
&f_t(\mathbf{x}_t^\star) - f_{t+1}(\mathbf{x}_{t+1}^\star) \\
&\quad = \big[f_t(\mathbf{x}_t^\star) - f_t(\mathbf{x}_{t+1}^\star)\big] + \big[f_t(\mathbf{x}_{t+1}^\star) - f_{t+1}(\mathbf{x}_{t+1}^\star)\big] \\
&\quad \leq \big\langle \nabla f_t(\mathbf{x}_t^\star), \mathbf{x}_t^\star - \mathbf{x}_{t+1}^\star \big\rangle + \Delta f_{\text{UB}} = \Delta f_{\text{UB}},
\end{aligned}
$$

and (b) is because of the definition of τ_{UB} under (19) and $\sum_{n\in\mathcal{N}} w_t^n = 1$.

For E_5, from the definition of ∇f_{UB} in (16), we have

$$
\begin{aligned}
E_5 &= \eta^2 L \sum_{n\in\mathcal{N}_t} w_t^n \|\nabla f_{l_t^n}^n(\mathbf{x}_{l_t^n}^\star)\|^2 \\
&\leq \eta^2 L \sum_{n\in\mathcal{N}_t} w_t^n \nabla f_{\text{UB}} \leq \eta^2 L \nabla f_{\text{UB}}. \qquad (69)
\end{aligned}
$$

Substituting the bounds on $E_1 + E_4$, $E_2 + E_3$, and E_5 in (67), (68), and (69) into (66), we have

$$
\begin{aligned}
D_{t+1} \leq \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m D_t + \sum_{n\in\mathcal{N}_t} \underbrace{w_t^n \big[1 - \eta(2\mu\epsilon - \eta L^2)\big]}_{\triangleq \lambda_t^n} D_{l_t^n} \\
+ \underbrace{2\tau_{\text{UB}}\Delta f_{\text{UB}} + \eta^2 L \nabla f_{\text{UB}}}_{\triangleq \Delta}. \qquad (70)
\end{aligned}
$$

Let $\theta_t \triangleq \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m + \sum_{n\in\mathcal{N}_t} \lambda_t^n$, where λ_t^n is defined in (70). We have

$$
\begin{aligned}
\theta_t &= \sum_{m\in\mathcal{N}\setminus\mathcal{N}_t} w_t^m + \sum_{n\in\mathcal{N}_t} w_t^n \big[1 - \eta(2\mu\epsilon - \eta L^2)\big] \\
&= \sum_{n\in\mathcal{N}} w_t^n - \eta(2\mu\epsilon - \eta L^2) \sum_{n\in\mathcal{N}_t} w_t^n \\
&\leq \underbrace{1 - \eta(2\mu\epsilon - \eta L^2) w_{\text{LB}}}_{\triangleq \theta_{\text{UB}}} \overset{(a)}{<} 1 \qquad (71)
\end{aligned}
$$

where (a) follows from $\eta < \frac{2\mu\epsilon}{L^2}$ and $w_{\text{LB}} \leq \sum_{n\in\mathcal{N}_t} w_t^n$.

Step 6: Relate D_t ***to*** D_1**.** We prove by induction that the following inequality holds for any t

$$
D_t \leq \rho^t D_1 + \delta \qquad (72)
$$

where $\rho = \theta_{\text{UB}}^{\frac{1}{\tau_{\text{UB}}}}$ and $\delta = \frac{\Delta}{1-\theta_{\text{UB}}}$ are defined in (18) and (19).

Obviously, (72) holds when $t = 1$, *i.e.,* $D_1 \leq \rho D_1 + \delta$, since $\rho < 1$ and $\delta \geq 0$. Suppose (72) holds for $t = 1, \ldots, t'$, we now prove (72) also holds for $t = t' + 1$. From (70) and $l_{t'}^n = t' + 1 - \tau_{t'}^n$, we have

$$\begin{aligned} D_{t'+1} &\leq \sum_{m\in\mathcal{N}\setminus\mathcal{N}_{t'}} w_{t'}^m D_{t'} + \sum_{n\in\mathcal{N}_{t'}} \lambda_{t'}^n D_{l_{t'}^n} + \Delta \\ &\overset{(a)}{\leq} \sum_{m\in\mathcal{N}\setminus\mathcal{N}_{t'}} w_{t'}^m \big[\rho^{t'} D_1 + \delta\big] + \sum_{n\in\mathcal{N}_{t'}} \lambda_{t'}^n \big[\rho^{l_{t'}^n} D_1 + \delta\big] + \Delta \\ &= \bigg(\sum_{m\in\mathcal{N}\setminus\mathcal{N}_{t'}} w_{t'}^m + \sum_{n\in\mathcal{N}_{t'}} \lambda_{t'}^n \rho^{1-\tau_{t'}^n}\bigg)\rho^{t'} D_1 \\ &\quad + \bigg(\sum_{m\in\mathcal{N}\setminus\mathcal{N}_{t'}} w_{t'}^m + \sum_{n\in\mathcal{N}_{t'}} \lambda_{t'}^n\bigg)\delta + \Delta \end{aligned} \tag{73}$$

where (a) follows from the induction.

We now bound the RHS of (73). Note that

$$\begin{aligned} &\sum_{m\in\mathcal{N}\setminus\mathcal{N}_{t'}} w_{t'}^m + \sum_{n\in\mathcal{N}_{t'}} \lambda_{t'}^n \rho^{1-\tau_{t'}^n} \\ &\overset{(a)}{\leq} \sum_{m\in\mathcal{N}\setminus\mathcal{N}_{t'}} w_{t'}^m + \sum_{n\in\mathcal{N}_{t'}} \lambda_{t'}^n \rho^{1-\tau_{\text{UB}}} \\ &\overset{(b)}{=} \sum_{m\in\mathcal{N}\setminus\mathcal{N}_{t'}} w_{t'}^m + \sum_{n\in\mathcal{N}_{t'}} \lambda_{t'}^n (\theta_{\text{UB}})^{-\frac{\tau_{\text{UB}}-1}{\tau_{\text{UB}}}} \\ &\overset{(c)}{\leq} \bigg(\sum_{m\in\mathcal{N}\setminus\mathcal{N}_{t'}} w_{t'}^m + \sum_{n\in\mathcal{N}_{t'}} \lambda_{t'}^n\bigg)(\theta_{\text{UB}})^{-\frac{\tau_{\text{UB}}-1}{\tau_{\text{UB}}}} \\ &\overset{(d)}{\leq} \theta_{\text{UB}}(\theta_{\text{UB}})^{-\frac{\tau_{\text{UB}}-1}{\tau_{\text{UB}}}} = (\theta_{\text{UB}})^{\frac{1}{\tau_{\text{UB}}}} = \rho \end{aligned} \tag{74}$$

where (a) follows from $\tau_t^n \leq \tau_{\text{UB}}$, (b) is because $\rho = \theta_{\text{UB}}^{\frac{1}{\tau_{\text{UB}}}}$, (c) is because $\theta_{\text{UB}} < 1$ and $\tau_{\text{UB}} \geq 1$ such that $(\theta_{\text{UB}})^{-\frac{\tau_{\text{UB}}-1}{\tau_{\text{UB}}}} > 1$, and (d) follows from the definition of θ_{UB}.

Also, we have

$$\begin{aligned} &\bigg(\sum_{m\in\mathcal{N}\setminus\mathcal{N}_{t'}} w_{t'}^m + \sum_{n\in\mathcal{N}_{t'}} \lambda_{t'}^n\bigg)\delta + \Delta \\ &\overset{(a)}{\leq} \theta_{\text{UB}}\delta + \Delta \overset{(b)}{=} \theta_{\text{UB}}\delta + (1-\theta_{\text{UB}})\delta = \delta \end{aligned} \tag{75}$$

where (a) follows from the definition of θ_{UB} and (b) is because $\delta = \frac{\Delta}{1-\theta_{\text{UB}}}$.

Substituting (74) and (75) into (73), we have

$$D_{t'+1} \leq \rho^{t'+1} D_1 + \delta, \tag{76}$$

which proves that (72) holds for $t = t' + 1$.

Therefore, by induction, we have (72) holds for any t. It then follows that (17) holds. ∎

Appendix B
Proof of Corollary 1

Summing (72) in the proof of Theorem 1 over $t \in \mathcal{T}$ and dividing both sides by T, we have

$$\begin{aligned} \frac{1}{T}\sum_{t\in\mathcal{T}} \big[f_t(\mathbf{x}_t) - f_t(\mathbf{x}_t^\star)\big] &= \frac{1}{T}\sum_{t\in\mathcal{T}} D_t \\ &\overset{(a)}{\leq} \frac{1}{T}\sum_{t\in\mathcal{T}} \big[\rho^t D_1 + \delta\big] = \frac{\big[1-\rho^T\big]\big[f_1(\mathbf{x}_1) - f_1(\mathbf{x}_1^\star)\big]}{(1-\rho)T} + \delta \\ &\leq \frac{f_1(\mathbf{x}_1) - f_1(\mathbf{x}_1^\star)}{(1-\rho)T} + \delta. \end{aligned} \tag{77}$$

where (a) follows from (72) in the proof of Theorem 1. ∎

Appendix C
Proof of Lemma 2

Proof: From the iterated law of expectation and by conducting two case studies on u_t^n, we have

$$\begin{aligned} &\mathbb{E}\{d_t^n u_t^n\} = \mathbb{E}\big\{\mathbb{E}\{d_t^n u_t^n | u_t^n\}\big\} \\ &= \mathbb{E}\{d_t^n u_t^n | u_t^n = 0\}\mathbb{P}\{u_t^n = 0\} + \mathbb{E}\{d_t^n u_t^n | u_t^n = 1\}\mathbb{P}\{u_t^n = 1\} \\ &\overset{(a)}{=} \mathbb{E}\{d_t^n | u_t^n = 1\}\mathbb{P}\{u_t^n = 1\} \\ &\overset{(b)}{=} \mathbb{E}\{d_t^n\}\mathbb{P}\{u_t^n = 1\} \overset{(c)}{=} \mathbb{E}\{d_t^n\}\mathbb{E}\{u_t^n\}. \end{aligned} \tag{78}$$

where (a) follows from $d_t^n u_t^n = 1$ with probability 0 when $u_t^n = 0$ such that $\mathbb{E}\{d_t^n u_t^n | u_t^n = 0\} = 0$, and (b) is because $d_t^n = 1$ with probability $\mathbb{E}\{d_t^n\}$ when $u_t^n = 1$ such that $\mathbb{E}\{d_t^n | u_t^n = 1\} = \mathbb{E}\{d_t^n\}$. ∎

Appendix D
Proof of Lemma 3

Proof: Consider a policy $\pi \in \Pi$ that satisfies (44b) and (44c). Let $D_T^n \triangleq \sum_{t\in\mathcal{T}} u_t^n$ be the number of times that device n is scheduled over T time slots. Let I_m^n be the number of slots between the $m-1$-th and the m-th schedule of device n, for any $m \in \{1, \ldots, D_T^n\}$. Let the number of remaining time slots be R^n after the D_T^n-th schedule of device n. The AoI area associated with the m-th schedule is $\sum_{i=1}^{I_m^n} i = \frac{(I_m^n+1)I_m^n}{2}$. From this equality, the time-averaged AoI of each device n can be expressed as

$$\begin{aligned} &\frac{1}{T}\sum_{t\in\mathcal{T}} \tau_t^n = \frac{1}{T}\bigg[\sum_{m=1}^{D_T^n} \frac{(I_m^n+1)I_m^n}{2} + \frac{(R^n+1)R^n}{2}\bigg] \\ &= \frac{1}{2T}\bigg[\sum_{m=1}^{D_T^n} (I_m^n)^2 + (R^n)^2 + \sum_{m=1}^{D_T^n} I_m^n + R^n\bigg] \\ &\overset{(a)}{=} \frac{1}{2}\bigg[\frac{D_T^n}{T}\Big(\frac{1}{D_T^n}\sum_{m=1}^{D_T^n} (I_m^n)^2\Big) + \frac{(R^n)^2}{T} + 1\bigg] \\ &\overset{(b)}{\geq} \frac{1}{2}\bigg[\frac{D_T^n}{T}\Big(\frac{1}{D_T^n}\sum_{m=1}^{D_T^n} I_m^n\Big)^2 + \frac{(R^n)^2}{T} + 1\bigg] \\ &\overset{(c)}{=} \frac{1}{2}\bigg[\frac{(T-R^n)^2}{TD_T^n} + \frac{(R^n)^2}{T} + 1\bigg] \overset{(d)}{\geq} \frac{1}{2}\bigg[\frac{T}{D_T^n+1} + 1\bigg] \end{aligned} \tag{79}$$

where (a) follows from $T = \sum_{m=1}^{D_T^n} I_m^n + R^n, \forall n$, (b) is because of the Jensen's inequality, (c) is because $\sum_{m=1}^{D_T^n} I_m^n = T - R^n$, and (d) follows from setting $R^n = \frac{T}{D_T^n+1}$ to minimize $\frac{(T-R^n)^2}{D_T^n} + (R^n)^2$.

Taking expectation on both sides of (79), we have

$$\frac{1}{T}\sum_{t\in\mathcal{T}} \mathbb{E}\{\tau_t^n\} \geq \frac{1}{2}\bigg[\frac{1}{\frac{1}{T}\mathbb{E}\{D_T^n\} + \frac{1}{T}} + 1\bigg]. \tag{80}$$

Multiplying both sides of (80) by α^n, summing over $n \in \mathcal{N}$, and take the limit $T \to \infty$, we have

$$\lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \alpha^n \mathbb{E}\{\tau_t^n\} \overset{(a)}{\geq} \frac{1}{2} \sum_{n\in\mathcal{N}} \frac{\alpha^n}{\lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \mathbb{E}\{u_t^n\}} + \frac{1}{2} \tag{81}$$

where (a) follows from $D_T^n = \sum_{t\in\mathcal{T}} u_t^n$ and $\sum_{n\in\mathcal{N}} \alpha^n = 1$.

Further noting that $c^n = \lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \mathbb{E}\{u_t^n\}$, the RHS of (81) is the objective of $\mathbf{P}_{\text{LB}}$, and (46b) is equivalent to (44b), we have that $\mathbf{P}_{\text{LB}}$ provides a lower bound to $\mathbf{P}_{\text{R}}$. ■

Appendix E
Proof of Lemma 4

Proof: At any time slot t, device n is scheduled with probability v^n. When $d_t^n = 1, \forall t$, the number of time slots I_m^n between the $m-1$-th and the m-th schedule of device n are i.i.d. with $\mathbb{P}\{I_m^n = r\} = v^n(1-v^n)^{r-1}$. The sequence of decision uploads at each device n is therefore a renewal process. From the generalization of the elementary renewal theorem for renewal process in [37], we have $\lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \mathbb{E}\{\tau_t^n\} = \frac{\mathbb{E}\{(I_m^n)^2\}}{\mathbb{E}\{I_m^n\}} + \frac{1}{2} = \frac{1}{v^n}, \forall n$. ■

Appendix F
Proof of Corollary 2

Proof: For i.i.d. communication indicator sequence d_t^n, we do not need the double relaxation approach in Section IV-E to find a lower bound problem for $\mathbf{P}$. Let $\bar{d}^n = \mathbb{P}\{d_t^n = 1\}$. Similar to the proof of Lemma 3, we can show that the following optimization problem provides a lower bound $\widetilde{\text{OPT}}_{\text{LB}} \leq \text{OPT}^\star$ to the original scheduling problem $\mathbf{P}$ when d_t^n is i.i.d.:

$$\widetilde{\mathbf{P}}_{\text{LB}}: \quad \widetilde{\text{OPT}}_{\text{LB}} = \min_{\pi\in\Pi} \left\{ \frac{1}{2} \sum_{n\in\mathcal{N}} \frac{\alpha^n}{\bar{d}^n c^n} + \frac{1}{2} \right\} \tag{82a}$$

$$\text{s.t.} \quad \bar{w}^n \bar{d}^n c^n \geq q^n, \quad \forall n, \tag{82b}$$

$$\sum_{n\in\mathcal{N}} \mathbb{E}\{u_t^n\} \leq K \tag{44c}$$

where c^n is defined below $\mathbf{P}_{\text{LB}}$. Let $\tilde{c}_{\text{LB}}^n$ be the long-term time-averaged expected number of schedules of device n under the optimal policy $\tilde{\pi}_{\text{LB}}$ that solves $\widetilde{\mathbf{P}}_{\text{LB}}$.

Similar to the proof of Lemma 4, we can show that for i.i.d. d_t^n, the long-term time-averaged expected AoI achieved by a stationary randomized policy with scheduling probabilities $\{v^n\}$ becomes $\lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \mathbb{E}\{\tau_t^n\} = \frac{1}{\bar{d}^n v^n}$. Applying the above inequality to $\mathbf{P}$, we have the following equivalent optimization problem $\widetilde{\mathbf{P}}_{\text{SR}}$ over Π_{SR}:

$$\widetilde{\mathbf{P}}_{\text{SR}}: \quad \widetilde{\text{OPT}}_{\text{SR}} = \min_{\pi\in\Pi_{\text{SR}}} \left\{ \sum_{n\in\mathcal{N}} \frac{\alpha^n}{\bar{d}^n v^n} \right\} \tag{83a}$$

$$\text{s.t.} \quad \bar{w}^n \bar{d}^n v^n \geq q^n, \quad \forall n, \tag{83b}$$

$$\sum_{n\in\mathcal{N}} v^n \leq K. \tag{83c}$$

Following the proof of (37) in Theorem 2 and noting that $\mathbb{E}\{d_t^n\} = \bar{d}^n$, we have

$$\mathbb{E}\{\Delta_t | S_t\} \leq -U \sum_{n\in\mathcal{N}} \bar{d}^n v^n \alpha^n \tau_t^n - \sum_{n\in\mathcal{N}} (\bar{w}^n \bar{d}^n v^n - q^n)[Q_t^n]_+ + U + \frac{CN}{2}. \tag{84}$$

Taking the expectation of (84) over S_t, summing it over $t \in \mathcal{T}$, and then dividing it by T, we have

$$\widetilde{\text{LHS}}_1 + \widetilde{\text{LHS}}_2 \leq -\frac{1}{T} \sum_{t\in\mathcal{T}} \mathbb{E}\{\Delta_t\} + U + \frac{CN}{2} \tag{85}$$

where

$$\widetilde{\text{LHS}}_1 \triangleq \frac{1}{T} \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} (\bar{w}^n \bar{d}^n v^n - q^n) \mathbb{E}\{[Q_t^n]_+\}, \tag{86}$$

$$\widetilde{\text{LHS}}_2 \triangleq \frac{U}{T} \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \bar{d}^n v^n \alpha^n \mathbb{E}\{\tau_t^n\}. \tag{87}$$

Let $v^n = \tilde{c}_{\text{LB}}^n$ in (85) and note that $\bar{w}^n \bar{d}^n \tilde{c}_{\text{LB}}^n - q^n \geq 0$ from (82b), we have $\widetilde{\text{LHS}}_1 \geq 0$. Dividing both sides of (85) by U and taking $T \to \infty$, we have

$$\lim_{T\to\infty} \frac{\widetilde{\text{LHS}}_2}{U} = \lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \bar{d}^n \tilde{c}_{\text{LB}}^n \alpha^n \mathbb{E}\{\tau_t^n\} \leq -\lim_{T\to\infty} \frac{1}{UT} \sum_{t\in\mathcal{T}} \mathbb{E}\{\Delta_t\} + \frac{1}{2}\Big[2 + \frac{CN}{U}\Big]. \tag{88}$$

Dividing both sides of (88) by $d^{\min} \tilde{c}_{\text{LB}}^{\min}$, where $d^{\min} \triangleq \min_n\{\bar{d}^n\}$ and $\tilde{c}_{\text{LB}}^{\min} \triangleq \min_n\{\tilde{c}_{\text{LB}}^n\}$, noting that $d^{\min} \leq 1$, and from the bound on $-\frac{1}{T} \sum_{t\in\mathcal{T}} \mathbb{E}\{\Delta_t\}$ in (42), we have

$$\text{OPT}_{\text{AIMWeL}} = \lim_{T\to\infty} \frac{1}{T} \sum_{t\in\mathcal{T}} \sum_{n\in\mathcal{N}} \alpha^n \mathbb{E}\{\tau_t^n\} \leq \frac{1}{2\tilde{c}_{\text{LB}}^{\min}} \Big[2 + \frac{CN}{U}\Big]. \tag{89}$$

Substituting $\tilde{c}_{\text{LB}}^n$ into $\widetilde{\mathbf{P}}_{\text{LB}}$, we have

$$\widetilde{\text{OPT}}_{\text{LB}} = \frac{1}{2} \sum_{n\in\mathcal{N}} \frac{\alpha^n}{\bar{d}^n \tilde{c}_{\text{LB}}^n} + \frac{1}{2} \geq \frac{\sum_{n\in\mathcal{N}} \alpha^n}{2 d^{\max} \tilde{c}_{\text{LB}}^{\max}} + \frac{1}{2} \geq \frac{1}{2\tilde{c}_{\text{LB}}^{\max}} \tag{90}$$

where $d^{\max} \triangleq \max_n\{\bar{d}^n\}$ and $\tilde{c}_{\text{LB}}^{\max} \triangleq \max_n\{\tilde{c}_{\text{LB}}^n\}$.

Comparing (89) with (90), we have

$$\frac{\text{OPT}_{\text{AIMWeL}}}{\widetilde{\text{OPT}}_{\text{LB}}} \leq \frac{\tilde{c}_{\text{LB}}^{\max}}{\tilde{c}_{\text{LB}}^{\min}} \Big[2 + \frac{CN}{U}\Big]. \tag{91}$$

Further note that $\widetilde{\text{OPT}}_{\text{LB}} \leq \text{OPT}^\star$, we complete the proof. ■

References

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

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

[3] 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.

[4] D. Alistarh, T. Hoefler, M. Johansson, N. Konstantinov, S. Khirirat, and C. Renggli, "The convergence of sparsified gradient methods," in *Proc. Adv. Neural Info. Proc. Sys. (NeurIPS)*, 2018.

[5] T. Lin, S. U. Stich, and M. Jaggi, "Don't use large mini-batches, use local SGD," in *Proc. Intel. Conf. Learn. Represent. (ICLR)*, 2020.

[6] 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.

[7] S. U. Stich, "Local SGD converges fast and communicates little," in *Proc. Intel. Conf. Learn. Represent. (ICLR)*, 2019.

[8] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proc. Mach. Learn. Syst. (MLSys)*, 2020.

[9] Y. Chen, X. Sun, and Y. Jin, "Communication-efficient federated deep learning with layerwise asynchronous model update and temporally weighted aggregation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, pp. 4229–4238, 2020.

[10] Y. Chen, Y. Ning, M. Slawski, and H. Rangwala, "Asynchronous online federated learning for edge devices with non-iid data," in *Proc. IEEE Int. Conf. Big Data (BigData)*, 2020.

[11] C.-H. Hu, Z. Chen, and E. G. Larsson, "Device scheduling and update aggregation policies for asynchronous federated learning," in *Proc. IEEE Intel. Workshop on Signal Process. Advances in Wireless Commun. (SPAWC)*, 2021.

[12] C. Xie, S. Koyejo, and I. Gupta, "Asynchronous federated optimization," in *Proc. NeurIPS Workshop Optim. Mach. Learn. (OPT)*, 2020.

[13] W. Wu, L. He, W. Lin, R. Mao, C. Maple, and S. Jarvis, "SAFA: A semi-asynchronous protocol for fast federated learning with low overhead," *IEEE Trans. Comput.*, vol. 70, pp. 655–665, 2021.

[14] J. Nguyen, K. Malik, H. Zhan, A. Yousefpour, M. Rabbat, M. Malek, and D. Huba, "Federated learning with buffered asynchronous aggregation," in *Proc. Intel. Conf. Mach. Learn. (ICML)*, 2021.

[15] Q. Ma, Y. Xu, H. Xu, Z. Jiang, L. Huang, and H. Huang, "FedSA: A semi-asynchronous federated learning mechanism in heterogeneous edge computing," *IEEE J. Sel. Areas Commun.*, vol. 39, pp. 3654–3672, 2021.

[16] S. Chen, X. Wang, P. Zhou, W. Wu, W. Lin, and Z. Wang, "Heterogeneous semi-asynchronous federated learning in internet of things: A multi-armed bandit approach," *IEEE Trans. Emerg. Topics Comput. Intell.*, vol. 6, pp. 1113–1124, 2022.

[17] C. Gutterman, K. Guo, S. Arora, X. Wang, L. Wu, E. Katz-Bassett, and G. Zussman, "Requet: Real-time QoE metric detection for encrypted YouTube traffic," *ACM Trans. Multimedia Comput. Commun. Appl.*, vol. 16, pp. 1–28, 2020.

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

[19] 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.

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

[21] S. Kaul, R. Yates, and M. Gruteser, "Real-time status: How often should one update?" in *Porc. IEEE Conf. Comput. Commun. (INFOCOM) Mini Conf.*, 2012.

[22] Y. Sun, I. Kadota, R. Talak, and E. Modiano, "Age of information: A new metric for information freshness," *Synthesis Lectures Commun. Netw.*, vol. 12, pp. 1–224, 2019.

[23] H. H. Yang, A. Arafa, T. Q. S. Quek, and H. V. Poor, "Age-based scheduling policy for federated learning in mobile edge networks," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP)*, 2020.

[24] W. Xia, W. Wen, K.-K. Wong, T. Q. S. Quek, J. Zhang, and H. Zhu, "Federated-learning-based client scheduling for low-latency wireless communications," *IEEE Wireless Commun.*, vol. 28, pp. 32–38, 2021.

[25] B. Buyukates and S. Ulukus, "Timely communication in federated learning," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM) Workshops*, 2021.

[26] I. Kadota, A. Sinha, and E. Modiano, "Scheduling algorithms for optimizing age of information in wireless networks with throughput constraints," *IEEE/ACM Trans. Netw.*, vol. 27, pp. 1359–1372, 2019.

[27] M. Moltafet, M. Leinonen, M. Codreanu, and N. Pappas, "Power minimization in wireless sensor networks with constrained AoI using stochastic optimization," in *Proc. Asilomar Conf. Signal Sys. Comput.*, 2019.

[28] E. Fountoulakis, N. Pappas, M. Codreanu, and A. Ephremides, "Optimal sampling cost in wireless networks with age of information constraints," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM) Workshops*, 2020.

[29] H. Tang, J. Wang, L. Song, and J. Song, "Minimizing age of information with power constraints: Multi-user opportunistic scheduling in multi-state time-varying channels," *IEEE J. Sel. Areas Commun.*, vol. 38, pp. 854–868, 2020.

[30] R. V. Bhat, R. Vaze, and M. Motani, "Throughput maximization with an average age of information constraint in fading channels," *IEEE Trans. Wireless Commun.*, vol. 20, pp. 481–494, 2021.

[31] N. Eshraghi and B. Liang, "Distributed online optimization over a heterogeneous network with any-batch mirror descent," in *Proc. Intel. Conf. Mach. Learn. (ICML)*, 2020.

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

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

[34] Y. Zhang, R. J. Ravier, M. M. Zavlanos, and V. Tarokh, "A distributed online convex optimization algorithm with improved dynamic regret," in *Proc. IEEE Conf. Decision Control (CDC)*, 2019.

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

[36] B. Li, "Efficient learning-based scheduling for information freshness in wireless networks," in *Proc. IEEE Conf. Comput. Commun. (INFOCOM)*, 2021.

[37] R. Gallager, *Stochastic Processes: Theory for Applications*. Cambridge University Press, 2013.

[38] Y. LeCun, C. Cortes, and C. Burges, "The MNIST database," 1998.

[39] H. Xiao, K. Rasul, and R. Vollgraf, "Fashion-mnist: A novel image dataset for benchmarking machine learning algorithms," 2017.