

# Client-Centric Federated Adaptive Optimization

Jianhui Sun

Department of Computer Science  
University of Virginia  
Charlottesville, Virginia, USA  
js9gu@virginia.edu

Heng Huang

Department of Computer Science  
University of Maryland  
College Park, Maryland, USA  
henghuanghh@gmail.com

Xidong Wu

Department of Electrical and Computer Engineering  
University of Pittsburgh  
Pittsburgh, Pennsylvania, USA  
xidong\_wu@outlook.com

Aidong Zhang

Department of Computer Science  
University of Virginia  
Charlottesville, Virginia, USA  
aidong@virginia.edu

## ABSTRACT

Federated Learning (FL) is a distributed learning paradigm where clients collaboratively train a model while keeping their own data private. With an increasing scale of clients and models, FL encounters two key challenges, client drift due to high degree of statistical/system heterogeneity, and lack of adaptivity. However, most existing FL research is based on unrealistic assumptions that virtually ignore system heterogeneity. In this paper, we propose Client-Centric Federated Adaptive Optimization, which is a class of novel federated adaptive optimization approaches. We enable several features in this framework such as arbitrary client participation, asynchronous server aggregation, and heterogeneous local computing, which are ubiquitous in real-world FL systems but are missed in most existing works. We provide a rigorous convergence analysis of our proposed framework for general nonconvex objectives, which is shown to converge with the best known rate. Extensive experiments show that our approaches consistently outperform the baseline by a large margin across benchmarks.

## KEYWORDS

Federated Learning, Federated Averaging, System Heterogeneity, Asynchronous Distributed Computing, Client-Centric Federated Adaptive Optimization

### ACM Reference Format:

Jianhui Sun, Xidong Wu, Heng Huang, and Aidong Zhang. 2024. Client-Centric Federated Adaptive Optimization. In . ACM, New York, NY, USA, 25 pages. <https://doi.org/XXXXXX.XXXXXXX>

## 1 INTRODUCTION

Federated learning (FL) is a distributed learning setting where many clients collaboratively train a machine learning model under the coordination of a central server, while keeping the training data

---

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from [permissions@acm.org](mailto:permissions@acm.org).

*Conference'17, July 2017, Washington, DC, USA*

© 2024 ACM.

ACM ISBN 978-1-4503-XXXX-X/18/06  
<https://doi.org/XXXXXX.XXXXXXX>

decentralized and private [39]. In cross-device FL setting, clients are usually an enormous number of edge devices, which own data that are critical to train a large-scale model on server but are not allowed to share the private data due to privacy concerns or regulatory requirements [8, 19].

Federated Averaging (FedAvg) [51] solves this problem by taking a “computation then aggregation” approach, i.e., having each client train locally with its own data, and the server aggregates the local models every once in a while. FedAvg and its variants [33, 40, 58, 77] enjoy communication efficiency as well as an appealing “linear speedup”, i.e., the convergence accelerates with increasing number of clients and local steps, even with only a small fraction of clients participating in each round [41, 91], and have thus become the most popular FL algorithms.

In spite of the empirical success, FedAvg and its variants face the following two key challenges that may severely destabilize the training and deteriorate its performances,

- **High degrees of heterogeneity.** The *statistical heterogeneity* (i.e., the local data distributions of clients are non *i.i.d.*), and *system heterogeneity* (i.e., the completely different levels of system characteristics such as battery level, computational/memory capacity, network connection), are both ubiquitous in large-scale edge networks. Such high degrees of heterogeneity may result in *client drift*, where local client models move away from globally optimal models [33, 46, 58]. FedAvg converges slowly or even diverges in presence of client drift.

- **Lack of adaptivity.** Despite its simplicity, there is a disadvantage that FedAvg scales its gradient uniformly and applies an identical learning rate to all features throughout the entire training process, which is similar to its non-federated counterpart Stochastic Gradient Descent (SGD). Such isotropic training leads to inferior performances when the feature space is sparse, or there exists a few dominant feature directions, which is quite common in training large-scale overparameterized deep models such as [7, 15, 17, 25].

Though various attempts have been made to tackle part of the above challenges, most of them belong to a regime which we refer to as **Server-Centric FL**. In server-centric FL, the following unrealistic assumptions on server-client coordination have been implicitly made: (1) each client is available upon the server’s request, and the server determines the sampling scheme of clients; and (2)

all clients conduct a highly synchronized and homogeneous local computing, i.e., all clients synchronize with server at each round and then run an identical number of local epochs. Server-centric FL is an idealized system in which the server orchestrates clients who have minimal independence.

In spite of the wide adoption of server-centric FL in existing literature [1, 5, 10, 27, 28, 33, 40, 41, 51, 52, 58, 64, 67, 75, 77, 78, 80, 85, 86, 91, 96, 101], it is nevertheless an over-simplification of real-world FL deployments. First, clients mostly determine whether to participate subject only to their own conditions and may be unavailable due to reasons that are completely unpredictable by the server e.g., low battery or no Wi-Fi connection. It is therefore impossible for server to dictate the participation scheme as in server-centric FL. Second, synchronization and homogeneity of local computing is inefficient and unnecessary as clients all have various levels of computing capability. Enforcement of synchrony results in “straggler effect”, i.e., slower clients lock the entire training process.

Most existing works which attempt to mitigate *statistical heterogeneity* [27, 40, 41, 57, 91, 102] or add adaptivity [58, 80] are in the server-centric FL regime, which virtually ignore the ubiquitous *system heterogeneity* in large-scale FL deployments.

In light of the limitations of server-centric FL, we propose an FL regime which we refer to as **Client-Centric Federated Learning** (CC-FL) to have a more precise characterization of real-world settings. And to further address the two key challenges FL algorithms encounter, i.e., client drift and lack of adaptivity, we propose **Client-Centric Federated Adaptive Optimization**, which is a general framework that consists of a class of novel federated optimization approaches, such as CC-FedAdam/CC-FedAdagrad/CC-FedAMS, which are formalized in Algorithm 2.

Specifically, in CC-Federated Adaptive Optimization, we enable several unique features which are missed by most existing works: (1) *arbitrarily heterogeneous local computation*, client participates only when it intends to, and each client can self-determine a time-varying and device-dependent number of local epochs; (2) *asynchronous aggregation*, each participating client can work with an asynchronous view of the global model from an outdated timestamp; and (3) *concurrent server-client optimization*, global optimization on server happens concurrently with local update by client, which avoids stragglers from stopping the entire training process. Each client takes an independent role in deciding the participation, and how much computation it carries out in this regime, which is why we refer to it as Client-Centric FL (CC-FL).

One key ingredient we incorporate is the server-side adaptive optimization, which is extremely helpful in mitigating client drift and adding adaptivity when training large-scale overparameterized models. Another critical advantage is that as the design of adaptive optimizers leverages the average of historical gradient information, the global update no longer relies only on current model update as in FedAvg baseline. Note that historical information would effectively regularize the global update from going wild when there is unexpected client distributional drift. Data-driven model training often leads to poor out-of-distribution performance [99].

We then provide the convergence analysis for CC-Federated Adaptive Optimization. Our theory reveals key factors that affect the

performances, and shows that our proposed approach obtains the best known convergence rate. We are unaware of any existing analysis that studies federated adaptive optimization with asynchrony and heterogeneous local computing.

Finally, extensive experiments show that any of the proposed CC-FedAdam/CC-FedAdagrad/CC-FedAMS can outperform FedAvg and its variants by a large margin across benchmarks. The improvement is consistent across various levels of statistical/system heterogeneity. We also carry out an exhaustive hyperparameter sensitivity analysis and our approach turns out to be quite easy to tune to obtain a much better performance than FedAvg.

Our main contributions can be summarized as follow,

- We propose a class of novel FL optimization approaches, which is effective in mitigating the client drift and lack of adaptivity issues in presence of system heterogeneity.
- We show our approach matches a best known convergence rate  $\mathcal{O}\left(\sqrt{\frac{1}{mKT}}\right)$  for general nonconvex objectives. To our best knowledge, this is the first convergence analysis on federated adaptive optimization with both statistical and system heterogeneity<sup>1</sup>.
- Empirical results demonstrate that our approaches consistently outperform widely used baseline by a large margin, across benchmarks and levels of client heterogeneity.

**Organization.** The rest of the paper is organized as follows. In Section 2, we introduce background that is pertinent to our proposed algorithm. In Section 3, we first discuss the limitations of existing server-centric FL algorithms and then introduce our proposed **Client-Centric Federated Adaptive Optimization**. In Section 4, we provide the convergence analysis of our proposed approach, followed by Section 5, where we provide experimental results that validate the effectiveness of our proposed algorithm. We discuss the related works in Section 6, and conclude our works in Section 7. We defer the proof of our convergence analysis and extra related works/experiments to Appendix due to space limit.

## 2 BACKGROUND

### 2.1 Federated Averaging (FedAvg)

Most Federated Learning problems can be formalized as,

$$\min_{x \in \mathbb{R}^d} f(x) \triangleq \frac{1}{n} \sum_{i=1}^n f_i(x), \quad \text{where } f_i(x) = \mathbb{E}_{\xi \sim \mathcal{D}_i} f_i(x, \xi). \quad (1)$$

where  $n$  is the total number of clients and  $x$  is the model parameter with  $d$  as its dimension. Each client  $i$  is associated with a local data distribution  $\mathcal{D}_i$  and a local objective function  $f_i(x) = \mathbb{E}_{\xi \sim \mathcal{D}_i} f_i(x, \xi)$ . The global objective function is the averaged objective among all clients. We consider the general *non i.i.d.* setting, i.e.,  $\mathcal{D}_i$  can be completely different from  $\mathcal{D}_j$  when  $i \neq j$ .

Federated Averaging (FedAvg) [51] is the first and probably most popular algorithm to optimize (1), formalized in Algorithm 1. Suppose the total number of communication rounds is  $T$ , at each round  $t \in \{1, \dots, T\}$ , FedAvg is composed of **client-level optimization**

---

<sup>1</sup> $\mathcal{O}\left(\sqrt{\frac{1}{mKT}}\right)$  is w.r.t total rounds  $T$ , number of local epochs  $K$ , and number of responsive clients  $m$ . Please refer to Section 4 for details.

and **server-side optimization**. Specifically, at round  $t$ , server randomly samples a subset of clients  $\mathcal{S}_t$  and sends the global model  $x_t$  to each participating client,

- **Client-level Optimization.** Each participating client  $i \in \mathcal{S}_t$  initializes the global model at  $x_{t,0}^i \leftarrow x_t$ , where  $x_{t,k}^i$  denotes the  $i$ -th client's local model at  $k$ -th local step. Each client  $i$  would then conduct  $K$  steps of local SGD, and updates  $x_{t,k+1}^i = x_{t,k}^i - \eta_l g_{t,k}^i$ , where  $\eta_l$  is the local learning rate. Client then computes the model difference  $\Delta_t^i = x_t - x_{t,K}^i$ .
- **Server-side Optimization.** Server collects the model differences from all participating clients  $\{\Delta_t^i\}_{i \in \mathcal{S}_t}$ , and aggregates by averaging, i.e.,  $\Delta_t = \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} \Delta_t^i$ . FedAvg algorithm then simply updates the global model by  $x_{t+1} = x_t - \eta \Delta_t$ , where  $\eta$  is the global learning rate.

Note that in vanilla FedAvg [51],  $\eta = 1$  throughout the entire training process, which is equivalent to directly averaging the local model  $x_{t,K}^i$ . If we regard averaged model difference  $\Delta_t$  as a “pseudo-gradient”, another interpretation of FedAvg’s server-side optimization  $x_{t+1} = x_t - \eta \Delta_t$  is a one-step SGD with constant learning rate 1.

Interpreting FedAvg as a one-step SGD with constant learning rate 1 would immediately inspire us to enable a more flexible optimizer other than SGD, e.g. [91] studies FedAvg with two-sided learning rates (i.e., FedAvg with  $\eta$  that may not be 1), and shows a proper selection of  $\eta$  enables FedAvg to achieve the best known convergence rate for general nonconvex objective functions. FedAvg with Momentum (FedAvgM) [27] is proposed to augment momentum, which is formalized in Algorithm 1. [27, 78] empirically show the convincing performances of FedAvgM, especially when the clients have highly heterogeneous data distributions.

## 2.2 Federated Adaptive Optimization

The key limitation of SGD and SGD momentum is that they apply an identical learning rate to all features uniformly. When the feature space is sparse or heterogeneous, which is ubiquitous when training large-scale models, the lack of adaptivity may lead to limited training speed and inferior learning performances. In light of such limitations, adaptive optimization approaches propose to scale the gradient by square roots of some form of the average of the squared values of past gradients, to apply an adaptive per-feature learning rate. Examples such as AdaGrad [18] and Adam [35], have long become the default optimizer in many learning tasks, due to fast convergence speed [84].

Inspired by its convincing empirical performances in non-federated settings, existing efforts have been made to combine adaptive optimization approaches with federated learning. [12, 58, 80] propose several federated adaptive optimization approaches, e.g., FedAdam, FedAdagrad, FedYogi, FedAMS, which essentially applies popular non-federated adaptive optimizers Adam [35], Yogi [94], Adagrad [18], and AMSGrad [59] to server-level optimization, respectively.

For expository purposes, we display FedAdam in Algorithm 1 (other variants can be formulated similarly). Specifically, the server regards  $\Delta_t$  as a pseudo-gradient, and updates the global model by Adam optimizer, where  $m_t$  is the ordinary “momentum buffer” as

---

**Algorithm 1:** FedAvg [51], FedAvgM [27], FedAdam [58]

---

**Input:**  
Number of clients  $n$ , objective function  
 $f(x) = \frac{1}{n} \sum_{i=1}^n f_i(x)$ ;  
Number of communication rounds  $T$ , **local** learning rate  $\eta_l$ ,  
**local** updating number  $K$ ;  
Initialization  $x_0$ , **global** learning rate  $\eta$ , and  $\beta, \gamma, \epsilon$ ;

```

1   for  $t \in \{1, \dots, T\}$  do
2     Randomly sample a subset  $\mathcal{S}_t$  of clients;
3     Server send  $x_t$  to subset  $\mathcal{S}_t$  of clients;
4     for each client  $i \in \mathcal{S}_t$  do
5       Initialize  $x_{t,0}^i \leftarrow x_t$ ;
6       for  $k \in \{0, 1, \dots, K-1\}$  do
7         Randomly sample a batch  $\xi_{t,k}^i$ ;
8         Compute  $g_{t,k}^i = \nabla f_i(x_{t,k}^i, \xi_{t,k}^i)$ ;
9         Update  $x_{t,k+1}^i = x_{t,k}^i - \eta_l g_{t,k}^i$ ;
10      end
11       $\Delta_t^i = x_t - x_{t,K}^i$ ;
12    end
13    Server aggregates  $\Delta_t = \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} \Delta_t^i$ ;
14    Update  $x_{t+1} = x_t - \eta \Delta_t$ 
15     $m_t = (1 - \beta) \Delta_t + \beta m_{t-1}$ 
16    Update  $x_{t+1} = x_t - \eta m_t$ 
17     $m_t = (1 - \beta) \Delta_t + \beta m_{t-1}$ 
18     $v_t = (1 - \gamma) \Delta_t^2 + \gamma v_{t-1}$ 
19    Update  $x_{t+1} = x_t - \eta \frac{m_t}{\sqrt{v_t} + \epsilon}$ 
20  end
21  return  $x_T$ 
```

---

in FedAvgM,  $v_t$  stores the accumulated squared values of past gradients, and the hyperparameter  $\epsilon > 0$  controls the *degree of adaptivity*, with smaller  $\epsilon$  representing higher degrees of adaptivity. By updating the global model with  $x_{t+1} = x_t - \eta \frac{m_t}{\sqrt{v_t} + \epsilon}$ , federated adaptive optimization mitigates client drift, and the server could adaptively adjust the learning rate on a per-feature basis, which is shown to achieve superior performances than FedAvg [58].

## 3 CLIENT-CENTRIC FEDERATED ADAPTIVE OPTIMIZATION

In Section 3.1, we first introduce the key limitations of existing FL algorithms, which requires heavy server-client coordination and ignores system heterogeneity. In light of the limitations, we then introduce our proposed algorithm Client-Centric Federated Adaptive Optimization in Section 3.2.

### 3.1 Limitations of Server-Centric FL

Most existing federated learning algorithms universally make the following assumption that rarely holds in real world deployment,

**At a given round, all clients synchronize with the same global model and they conduct identical number of local computations.**

Specifically, if we observe the clients' local computation part in Algorithm 1 (i.e. Line 5-9), we would summarize that the following assumptions have been intrinsically made, (a) (**Homogeneous Local Updates**) Clients run an identical number of  $K$  steps; (b) (**Uniform Client Participation**) Each client is sampled by the server in a given round uniformly and independently according to an underlying distribution; (c) (**Synchronous Local Clients**) All participating clients synchronize at any round  $t$ , i.e., they initialize with the global model at current time  $x_t$ .

These assumptions provide a useful yet over-simplified characteristic of real-world system. Due to their simplicity, most existing FL algorithms and their corresponding convergence analysis are proposed and derived based on the above assumptions, see e.g. [1, 27, 33, 40, 41, 51, 58, 64, 77, 78, 80, 91, 96] (please check Section 6 and Appendix ?? for a comprehensive review of existing works).

With the above assumptions, server takes a centric role in the federated learning system, as all clients always synchronize with the server, and the server determines the client sampling scheme as well as number of local updates, we therefore refer to this setting as **Server-Centric Federated Learning System**.

However, we would like to argue that server-centric federated learning is an unrealistic characterization of real-world system. Especially considering that it typically takes thousands of communication rounds to converge in large-scale cross-device deployments, it is impossible to ensure these assumptions hold throughout the entire process. Instead, clients in a realistic cross-device FL system usually have the following characteristics [32, 39],

- (**Heterogeneous Client Capability**) The clients may have completely different computational capability, and faster clients are able to carry out more local computations than slower clients during the same time period. Requesting identical local updates would straggle the training and incur unnecessary energy waste.
- (**Unpredictable Client Availability**) The clients may have arbitrary availability due to various constraints, e.g., unstable network connection, battery outage, or low participation willingness. Thus, each client determines to participate or not in a given communication round purely dependent on its own condition and the decision is completely unpredictable by the server.
- (**Asynchronous Local Model**) Due to the random nature of client participation and capability, a reasonable server does not wait until the slowest client completes local computation, but would instead start the next round as long as a sufficient number of clients respond. The rest of the clients would participate in a future round when they finish and decide to respond. Though this paradigm is more efficient than its synchronous counterpart, the behavior of clients may be more chaotic as clients may compute based on an outdated global model  $x_{t-\tau_{t,i}}$  instead of  $x_t$  (up to a time-varying, device-dependent random delay  $\tau_{t,i}$ ).

In summary, due to their discrepancy against real-world settings, the applicability of these server-centric FL algorithms is questionable. Traditionally, most works that recognize client heterogeneity focus explicitly on statistical heterogeneity [42, 91], i.e., the local data distribution distinguishes with each other, while in reality the system heterogeneity is as ubiquitous as statistical heterogeneity and largely remains unexplored.

### 3.2 Methodology

**Algorithm 2:** A class of **Client-Centric Federated Adaptive Optimization** approaches. CC-FedAdagrad

CC-FedAdam CC-FedAMS

---

**Input:** Number of clients  $n$ , objective  $f(x) = \frac{1}{n} \sum_{i=1}^n f_i(x)$ , initialization  $x_0$ , Number of rounds  $T$ , **local** learning rate  $\eta_L$ , Number of participating clients  $m$ , server-side hyperparameter  $\eta, \beta, \gamma, \epsilon$

```

1 for  $t \in \{1, \dots, T\}$  do
2   At Each Client
3     Self-determine whether to participate in the
4     training, if not, stay idle.
5     Once determine to participate, retrieve a global
6     model  $x_\mu$  from the server ( $\mu$  may not be  $t$ )
7     Trigger Client-Centric Local Computation
8       (Algorithm 3),  $\Delta_\mu^i = \text{CC-Local}(i, t, \mu, \eta_L)$ 
9     Send local update  $\Delta_\mu^i$ 
10    At Server (Concurrently with Client)
11    Collect the local updates  $\{\Delta_{t-\tau_{t,i}}^i, i \in \mathcal{S}_t\}$  returned
12    from a subset of clients  $\mathcal{S}_t$ , where  $\tau_{t,i}$  represents
13    the random delay of the client  $i$ 's local update,
14     $i \in \mathcal{S}_t$ 
15    Aggregate:  $\Delta_t = \frac{1}{m} \sum_{i \in \mathcal{S}_t} \Delta_{t-\tau_{t,i}}^i$ 
16    Update momentum buffer  $m_t = (1 - \beta)\Delta_t + \beta m_{t-1}$ ;
17     $\hat{v}_t = \hat{v}_{t-1} + \Delta_t^2$ 
18     $\hat{v}_t = (1 - \gamma)\hat{v}_t^2 + \gamma\hat{v}_{t-1}$ 
19     $v_t = (1 - \gamma)\hat{v}_t^2 + \gamma v_{t-1}$ 
20     $\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$ 
21    Update  $x_{t+1} = x_t - \eta \frac{m_t}{\sqrt{\hat{v}_t + \epsilon}}$ ;
22  end
23  return  $x_T$ 

```

---

In light of the limitations of server-centric federated learning settings and the convincing empirical results of federated adaptive optimization, in this section, we propose a novel framework which we refer to as **Client-Centric Federated Adaptive Optimization**, which is formalized in Algorithm 2.

#### 3.2.1 Workflow of Algorithm 2.

In the Client-Centric FL (CC-FL) framework, each client takes an independent role in deciding the participation, and how much

---

**Algorithm 3: Client-Centric Local Computation**  
**CC-Local** ( $i, t, \mu, \eta_l$ )

---

**Input:** client index  $i$ , current round  $t$ , retrieved global model timestamp  $\mu$ , **local** learning rate  $\eta_l$

- 1 Initialize the local model  $x_{\mu,0}^i = x_\mu$ .
- 2 Determine a number of local steps  $K_{t,i}$ , which can be time-varying and device-dependent based on its own condition.
- 3 **for**  $k \in \{0, 1, \dots, K_{t,i} - 1\}$  **do**
- 4     Randomly sample a batch  $\xi_{\mu,k}^i$
- 5     Compute  $g_{\mu,k}^i = \nabla f_i(x_{\mu,k}^i; \xi_{\mu,k}^i)$
- 6     Update  $x_{\mu,k+1}^i = x_{\mu,k}^i - \eta_l g_{\mu,k}^i$
- 7 **end**
- 8 Compute model difference  $\Delta_\mu^i = x_\mu - x_{\mu,K_{t,i}}^i$
- 9 Normalize w.r.t.  $K_{t,i}$ , and send  $\Delta_\mu^i = \frac{\Delta_\mu^i}{K_{t,i}}$
- 10 return  $\Delta_\mu^i$

---

computation it carries out. Note that CC-Federated Adaptive Optimization is a general and flexible framework in which any adaptive optimizer could serve as a plug-and-play module, and we display only three examples in Algorithm 2, i.e., CC-FedAdagrad/CC-FedAdam/CC-FedAMS. For expository purposes, we walk through the workflow of Client-Centric Federated Adaptive Optimization only with CC-FedAMS.

Suppose  $T$  is the total number of rounds in CC-FedAMS. At round  $t \in \{1, 2, \dots, T\}$ , each client can self-determine whether to participate, and staying idle in the entire training process is allowed. Once it determines to participate, it downloads the global model  $x_\mu$  from the server. Note that since each client may choose to participate at a different timestamp, the global model  $x_\mu$  may not be from the current timestamp  $x_t$ . The client then triggers local computation **CC-Local** (i.e., Algorithm 3). It first initializes  $x_{\mu,0}^i = x_\mu$  locally and carries out  $K_{t,i}$  steps of local SGD (i.e.,  $x_{\mu,k+1}^i = x_{\mu,k}^i - \eta_l g_{\mu,k}^i$  for  $K_{t,i}$  steps). Note that in server-centric FL algorithm,  $K_{t,i}$  is a constant that does not vary with  $t$  and  $i$  (see e.g., Algorithm 1). Here, we allow  $K_{t,i}$  to be time-varying and device-dependent. The client then computes a **normalized** model update by  $\Delta_\mu^i = \frac{x_{\mu,0}^i - x_{\mu,K_{t,i}}^i}{K_{t,i}}$  and sends  $\Delta_\mu^i$  to server. The normalization is to de-bias the global model to avoid favoring clients with more local updates.

Concurrently with the local computation **CC-Local**, the server collects the model updates from responsive clients  $\{\Delta_{t-\tau_{t,i}}^i, i \in \mathcal{S}_t\}$ , where  $\mathcal{S}_t$  is a “buffer” of responsive clients. Note that for each client  $i \in \mathcal{S}_t$ , it may participate at a historical timestamp  $t - \tau_{t,i}$ , which is up to a random delay  $\tau_{t,i}$ . In server-centric FL algorithm,  $\tau_{t,i} = 0$ . The global update only takes place once the buffer collects  $m$  client updates, i.e., as long as  $|\mathcal{S}_t|$  reaches  $m$ , regardless of the status of the rest of the clients. Such parallel runs of clients and server avoid unnecessary waiting time and wasteful resources. The global updating rule is adapted from a popular non-federated adaptive optimizer AMSGrad [59], which uses a max stabilization step

$\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$  that generates a non-decreasing  $\hat{v}_t$  sequence to solve the non-convergence issue in Adam.

### 3.2.2 Key Algorithmic Designs.

Several unique features of Algorithm 2 distinguish itself from ordinary server-centric FL, which include (a) **Normalized Model Update** (Line 9 in Alg 3), (b) **Size  $m$  Buffer  $\mathcal{S}_t$**  (Line 8 in Alg 2), (c) **Server-side Adaptive Optimization** (Line 11-15 in Alg 2). These features are key to alleviate system heterogeneity and client drift. We summarize the details and reasoning in Appendix

## 4 CONVERGENCE ANALYSIS

In this section, we provide the convergence analysis for our proposed Client-Centric Federated Adaptive Optimization and its practical implications.

To our best knowledge, this is the first definitive convergence result on federated adaptive optimization with system heterogeneity. Please note that enabling adaptive scaling in asynchronous FL is challenging in theoretical proof and is not an extension of results in [58] or [92], since the chaotic iterates caused by asynchrony and a nonzero momentum factor  $\beta$  are ignored in [58, 92].

We study general non-convex objective functions under statistical heterogeneity, i.e., each local loss  $f_i(x)$  (and thus the global loss  $f(x)$ ) may be non-convex<sup>2</sup>, and  $\mathcal{D}_i \neq \mathcal{D}_j$  when  $i \neq j$ .

**Assumption 1** (Smoothness). Each local loss  $f_i(x)$  has  $L$ -Lipschitz continuous gradients, i.e.,  $\forall x, x' \in \mathbb{R}^d$ , we have  $\|\nabla f_i(x) - \nabla f_i(x')\| \leq L \|x - x'\|$ , where  $L$  is the Lipschitz constant. And  $f$  has finite optimal value, i.e.,  $f^* \triangleq \min_x f(x)$  exists, and  $f^* > -\infty$ .

**Assumption 2** (Unbiased Bounded Gradient). We could access an unbiased estimator  $g_{t,k}^i = \nabla f_i(x_{t,k}^i; \xi_{t,k}^i)$  of true gradient  $\nabla f_i(x_{t,k}^i)$  for all  $t, k, i$ , where  $g_{t,k}^i$  is the stochastic gradient with minibatch  $\xi_{t,k}^i$ . And the stochastic gradient is bounded, i.e.,  $\|g_{t,k}^i\| \leq G$ .

**Assumption 3** (Bounded Local Variance). Each stochastic gradient on the  $i$ -th client has a bounded local variance, i.e., we have  $\mathbb{E} [\|g_{t,k}^i - \nabla f_i(x_{t,k}^i)\|^2] \leq \sigma_l^2$ .

**Assumption 4** (Bounded Global Variance). Local loss  $\{f_i\}_{i=1}^n$  have bounded global variance, i.e.,  $\forall x, \frac{1}{n} \sum_{i=1}^n \|\nabla f_i(x) - \nabla f(x)\|^2 \leq \sigma_g^2$ .

Assumptions 1 - 4 are all standard and mild assumptions in federated learning research, and have been universally adopted in most existing works [4, 34, 35, 42, 56, 58, 59, 80]. Assumption 2 is a standard assumption when studying adaptive optimization approaches and has been widely adopted in [13, 35, 58, 59, 80].

Note that Assumption 4 marks that we are studying general *non i.i.d.* settings, where we allow each client has heterogeneous data distributions, as  $\sigma_g^2 = 0$  corresponds to the *i.i.d.* setting. We also do not require stronger and unrealistic assumptions such as convex objective or Lipschitz Hessian that are used in existing works such as [22, 88].

<sup>2</sup>Objective functions in modern large-scale neural networks e.g., VGG [62], ResNet [25], and DenseNet [29] are non-convex [37].

Note that the convergence analysis is not a simple combination of existing results, the analysis needs to overcome novel challenges due to the chaotic behavior caused by asynchrony and heterogeneous local computing. As a matter of fact, there is no existing theoretical result on federated adaptive optimization with system heterogeneity to our best knowledge.

We state the main convergence theorem of Client-Centric Federated Adaptive Optimization in Theorem 4.1<sup>3</sup>, and then analyze the practical implications in Corollaries and Remarks.

**Theorem 4.1** (Convergence of CC-Federated Adaptive Optimization). *Suppose  $\{f_i\}_{i=1}^n$  fulfills Assumptions 1-4. Suppose the maximum delay is bounded, i.e.,  $\tau_{t,i} \leq \tau < \infty$  for any  $i \in S_t$  and  $t \in \{1, \dots, T\}$ .*

Under the condition

$$\eta_l \leq \min \left\{ \frac{1}{8K_{t,\max}L}, \sqrt{\frac{1}{120L^2\epsilon\tau K_{t,\max}^2}}, \frac{\epsilon}{\sqrt{T}\bar{G}} \right\}$$

, where  $K_{t,\max} = \max_{i \in S_t} K_{t,i}$ , and suppose  $H_1\eta_l^2 + H_2\eta_l \leq \epsilon^2$ , where  $H_1 \triangleq 2\eta^2 L^2 \tau^2$ ,  $H_2 \triangleq 4\eta LC_\beta^2 + 6\eta Le + 2Ge$ ,  $C_\beta = \frac{\beta}{1-\beta}$ .<sup>4</sup> And further assume each client is included in  $S_t$  with probability  $\frac{m}{n}$  uniformly and independently. We would have:

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla f(x_t)\|^2] \leq \frac{8\epsilon}{\eta\eta_l T} (f(z_0) - f^*) + \frac{\Phi}{T} + \Phi_g \sigma_g^2 + \Phi_l \sigma_l^2$$

where<sup>5</sup>

$$\begin{aligned} \Phi &\triangleq 4dG^2C_\beta + L\eta\eta_l G^2 C_\beta^2 \frac{4d}{\epsilon}, \quad \Phi_g \triangleq 240\eta_l^2 L^2 \phi_2, \\ \Phi_l &\triangleq 40\eta_l^2 L^2 \phi_1 + \frac{4\eta\eta_l LC_\beta^2 + 6L\eta\eta_l + 2\eta_l G}{m\epsilon} \phi_3 + \frac{2\eta_l^2 \eta_l^2 L^2 \tau^2}{\epsilon^2 m} \phi_3 \end{aligned}$$

PROOF. We defer the proof to Appendix B due to space limit. □

**Corollary 4.1.1** (Convergence Rate of Client-Centric Federated Adaptive Optimization). *Suppose an identical  $K$  for all  $t$  and  $i$ . By setting  $\eta_l = \Theta(\frac{1}{\sqrt{T}})$ ,  $\eta = \Theta(\sqrt{mK})$ , we have the convergence rate as*

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla f(x_t)\|^2] = O\left(\sqrt{\frac{1}{mKT}}\right) + O\left(\frac{K^2}{T}\right) + O\left(\frac{\tau^2}{T}\right) \quad (2)$$

If we further have a sufficiently large  $T$  (i.e. run the algorithm long enough) and an only moderately large  $\tau$  (i.e. a manageable level of random delay), specifically, if we have  $T \geq mK^5$  and  $\tau \leq \left(\frac{T}{mK}\right)^{\frac{1}{4}}$ , we obtain a rate,

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla f(x_t)\|^2] = O\left(\sqrt{\frac{1}{mKT}}\right) \quad (3)$$

<sup>3</sup>For expository purposes, we prove the convergence for CC-FedAdagrad. Similar convergence results could be easily generalized to other CC-Federated Adaptive Optimization approaches.

<sup>4</sup>Such constraint on  $\eta_l$  is standard and easily fulfilled by ordinary value assignment. Similar constraint has been used in [80, 91, 92].

<sup>5</sup>We denote  $\frac{1}{K_t} = \frac{1}{m} \sum_{i \in S_t} \frac{1}{K_{t,i}}$ ,  $\bar{K}_t \triangleq \frac{1}{m} \sum_{i \in S_t} K_{t,i}$ ,  $\hat{K}_t^2 \triangleq \frac{1}{m} \sum_{i \in S_t} K_{t,i}^2$ ,  $\phi_1 \triangleq \frac{1}{T} \sum_{t=0}^{T-1} \bar{K}_t$ ,  $\phi_2 \triangleq \frac{1}{T} \sum_{t=0}^{T-1} \hat{K}_t^2$ , and  $\phi_3 \triangleq \frac{1}{T} \sum_{t=0}^{T-1} \frac{1}{K_t}$ , for ease of notation.

**Remark 4.1.1** (How Random Delay Impacts Convergence?). Intuitively, due to the chaotic behavior brought by asynchronous clients, the convergence of CC-federated adaptive optimization is expected to be negatively impacted by the random delay  $\tau$ . Corollary 4.1.1 indicates the slowdown effect from the random delay  $\tau$  through the  $O\left(\frac{\tau^2}{T}\right)$  term. But fortunately, Corollary 4.1.1 also implies, if  $\tau$  is only moderately large, then client-centric federated adaptive optimization can obtain an  $O\left(\sqrt{\frac{1}{mKT}}\right)$  rate, which does not depend on  $\tau$ , i.e. the negative impact of using outdated information vanishes asymptotically. Such  $O\left(\sqrt{\frac{1}{mKT}}\right)$  rate indicates our client-centric federated adaptive optimization can converge with chaotic system heterogeneity, and matches the best known convergence rate in asynchronous computing [3, 44, 92].

**Remark 4.1.2** (Convergence Rate of Arbitrary Participation). A natural question is that what happens if the clients participate entirely arbitrarily, since we assume client is included in participation uniformly at random in Theorem 4.1. We could show the resulting convergence rate is  $O\left(\frac{1}{\sqrt{mKT}}\right) + O\left(\frac{\tau^2}{T}\right) + O\left(\frac{K^2}{T}\right) + \Omega\left(\sigma_g^2\right)$ . And we could further show  $\Omega(\sigma_g^2)$  is indeed the lower-bound rate (i.e. unavoidable) by constructing worst-case scenario (imagine only two clients  $f(x) = \frac{1}{2}(f_1 + f_2)$ , where the loss for 1st client is  $f_1(x) = (x+G)^2$  and the 2nd client is  $f_2(x) = (x-G)^2$ . In this case  $\sigma_g^2 = 4G^2$ . Suppose client 1 never participates, then the optimum  $x^*$  any algorithm could find is  $G$ , where  $\mathbb{E}[\|\nabla f(x^*)\|^2] = \Omega(\sigma_g^2)$ ). Thus, if there is no condition for the participation, any algorithm is subject to non-convergence. Theorem 4.1 is shown with one particular participation pattern, which can be relaxed or altered.

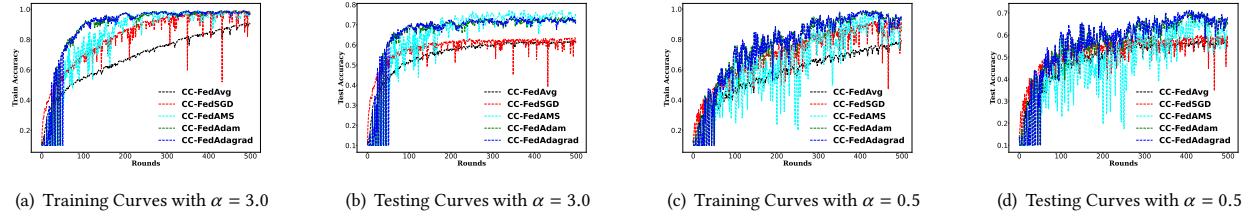
**Remark 4.1.3** (Linear Speedup w.r.t  $m, K$ ). The  $O\left(\sqrt{\frac{1}{mKT}}\right)$  rate also reveals an appealing linear speedup effect of number of participating clients  $m$  and number of local epochs  $K$ . Linear speedup in terms of  $m$  indicates the convergence is faster with more participating clients. Such speedup is not revealed by existing bound in [54]. The speedup in terms of  $K$  reflects an important trade-off between local computation and server-client communication, i.e., more local computation (larger  $K$ ) can shorten the convergence, thus requiring fewer rounds of communication (smaller  $T$ ).

## 5 EXPERIMENTS

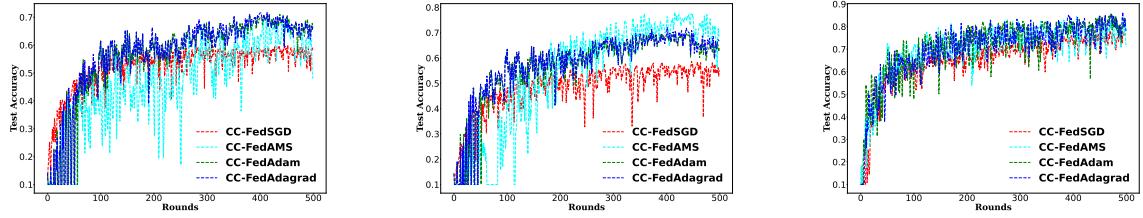
We present extensive experimental results on vision and language tasks in 5.2 that validate the effectiveness of our proposed approaches. We also analyze the hyperparameter sensitivity in 5.3. We defer extra experimental results to Appendix due to space limit. Our codes are available at <https://anonymous.4open.science/r/CC-Federated-Adaptive-Optimization>

### 5.1 Experimental Setting

We experiment with benchmark datasets such as Fashion-MNIST [87], CIFAR10, CIFAR100 [36], and StackOverflow. For baseline and



**Figure 1: Training and testing curves for various CC-Federated Adaptive Optimizers (ResNet on CIFAR-10) under different Concentration Parameters  $\alpha$ .**



(a) Testing curve for ResNet on CIFAR-10 with  $\tau = 10$ . (b) Testing curve for ResNet on CIFAR-10 with  $R = 3$ . (c) Testing curve for a shallow CNN on FMNIST.

**Figure 2: Testing curve for various CC-Federated Adaptive Optimizers on different settings of  $\tau$ ,  $R$ , datasets, and architectures.**

our proposed approaches, we sweep a wide range of hyperparameters and display the curves under best hyperparameter settings [9, 50, 98, 100].

### How Statistical Heterogeneity is Implemented?

We enforce the label imbalance across all clients to simulate the statistical heterogeneity (i.e. *non i.i.d.* settings). Specifically, each client is allocated a proportion of the samples of each label according to a Dirichlet distribution<sup>6</sup> which is parameterized by a *concentration* parameter  $\alpha > 0$ .  $\alpha$  controls the degree of non i.i.d. across clients, with  $\alpha \rightarrow \infty$  indicates all clients have identical distributions (i.e., no statistical heterogeneity) and  $\alpha \rightarrow 0$  indicates each client only has samples from one random class. Same procedure has been used in [27, 38, 77, 93].

### How System Heterogeneity is Implemented?

System heterogeneity has two facets in this paper, i.e., asynchronous aggregation and heterogeneous local computing. To simulate the asynchrony, we allow each client to select one global model randomly from the last recent  $\tau + 1$  global models, where  $\tau$  is the maximum random delay and controls the degree of asynchrony. Note that ordinary synchronous setting is a special example when  $\tau = 0$ . To simulate the heterogeneous local computation, we allow each worker to randomly select local epoch number from  $\{1, 2, \dots, K * R\}$  at each round so that each worker has a time-varying, device-dependent local epoch, where  $K$  is a default number of local epoch, and  $R$  is a degree of randomness. For example, if  $K = 3$ , ordinary homogeneous setting enforces all clients

carry out 3 local epochs, but we allow each client randomly select a number of local epoch from  $\{1, 2, \dots, 6\}$  if  $R = 2$ . Larger  $R$  obviously indicates larger randomness.

### Default Experimental Setting

Unless specified otherwise, we take the following default experimental settings as summarized in Table 1. We have 100 clients in all experiments, and the buffer size  $m$  is 5 (i.e. server updates globally once it collects 5 local updates), concentration parameter  $\alpha = 0.5$ , maximum random delay  $\tau = 5$ , default local epoch  $K$  is 3, and degree of randomness  $R$  is 2. We fix  $\beta = 0.9$ ,  $\gamma = 0.99$  following the settings in [58], except in Section 5.3 where we test the sensitivity of these hyperparameters.

**Table 1: Default Experimental Settings**

Number of Clients: 100	Buffer Size $m$ : 5
Concentration Parameter: $\alpha = 0.5$	Default local epoch $K$ : 3
Local Learning Rate: $\eta_L = 0.01$	Total Number of Rounds: 500
$\beta$ : 0.9	$\gamma$ : 0.99
$\tau$ : 5	Local Momentum: Disabled

## 5.2 Performances on Benchmark Datasets

We sweep  $\eta$  in a wide grid  $\eta \in \{10^{-3}, 10^{-2.5}, \dots, 10^1\}$ , and show the curves with best hyperparameter settings below [43, 49, 74, 97].

Figure 1(a) and 1(b) show the training/testing performances of training ResNet [26] on CIFAR-10 for 500 rounds. The concentration parameter  $\alpha = 3.0$  and maximum delay  $\tau = 5$ . Note that CC-FedAvg in the figures denotes the ordinary FedAvg [51] (i.e.

<sup>6</sup>The training examples on each client are sampled independently with class labels following a categorical distribution parameterized by a vector  $q$  over  $C$  classes (10 in CIFAR-10). Apparently,  $q$  fulfills the following properties,  $q_i \geq 0$  for  $i \in \{1, \dots, C\}$ , and  $\sum_{i=1}^C q_i = 1$ . We draw  $q \sim \text{Dir}(\alpha p)$  from a Dirichlet distribution, and  $\alpha > 0$  is a concentration parameter controlling the identicalness among clients. We refer full details of the simulation procedure to [27, 38]

**Table 2: Results on CIFAR-100 & StackOverflow**

Approach \ Dataset	CIFAR-100 (Accuracy)	StackOverflow (Recall@5)
CC-FedSGD	44.0	29.8
CC-FedAdagrad (Ours)	47.3	<b>65.1</b>
CC-FedAdam (Ours)	<b>51.5</b>	64.8

$\eta = 1$ ) in CC framework, while CC-FedSGD denotes a generalized FedAvg (i.e. search the best  $\eta$  from the same grid as our proposed approaches). We could observe that this extra degree of freedom brings a significant acceleration in training (CC-FedSGD converges much faster than CC-FedAvg in Figure 1(a)). Though it only exhibits a marginal improvement in testing, CC-FedSGD is apparently a stronger baseline than CC-FedAvg. As of our proposed approaches, any of CC-FedAMS/CC-FedAdam/CC-FedAdagrad outperforms CC-FedSGD by more than 10% in testing performances after only 300 rounds. CC-FedSGD could catch up the training curve of CC-FedAMS in Figure 1(a), while the large gap between testing curves persists in Figure 1(b).

#### Improvement is Consistent under Various Settings

We test the performances across different settings of  $\tau$ ,  $R$ , datasets, and architectures.

We run experiments on more benchmarks, one image task (CIFAR-100) and one language task (tag prediction on StackOverflow). The results are presented in Table 2. Our proposed approaches outperform the baseline significantly across benchmarks, especially in NLP task (approximately 30%), where gradient tends to be more sparse which adaptive optimizers can capitalize on [58].

In Figure 1(c) and 1(d), we show the training/testing performances with a higher level of statistical heterogeneity, i.e.  $\alpha = 0.5$  while all other settings are identical to Figure 1(a) and 1(b). The similarly superior performances of CC-FedAMS/CC-FedAdam/CC-FedAdagrad persist (by a 10% margin over CC-FedSGD).

In Figure 2, we show results across different  $\tau$ ,  $R$ , and benchmarks. Note that we only show the testing curves here as it is a more important metric than training curves. We defer training curves to Appendix due to space limit.

Specifically, in Figure 2(a), we test a higher level of system heterogeneity  $\tau = 10$ , i.e., each client is allowed to randomly sample from the most recent 10 global models, while all other settings are identical to Figure 1(d) where  $\tau = 5$ . Still, under a more asynchronous environment, it is obvious our proposed approaches outperform CC-FedSGD consistently. In Figure 2(b), we test  $R = 3$ , i.e., each client is allowed to randomly select a local epoch in  $\{1, 2, \dots, 9\}$  instead of  $R = 2$  in Figure 1(d). We could observe, the performance gap between CC-FedAMS and CC-FedSGD actually increases to a 15% margin with more heterogeneous local computations.

Figure 2(c) shows the results of running a shallow CNN architecture from [51] on Fashion-MNIST dataset. The CC-Federated Adaptive family of approaches still perform better than CC-FedSGD. But the gap is not as large as ResNet. The reason is likely due to adaptivity is most advantageous in settings where the feature space is

sparse or anisotropic, which is common in training overparameterized deep models such as ResNet.

More experimental results are deferred to Appendix.

### 5.3 Ease of Hyperparameter Tuning

We sweep a wide range of key hyperparameters  $\{\beta, \gamma, \epsilon\}$  in CC-Federated Adaptive Optimizers under default experimental settings. Specifically, we sweep a  $\beta$  grid:  $\{0.0, 0.5, 0.7, 0.8, 0.9, \dots, 0.995\}$ ,  $\epsilon$  grid:  $\{10^{-4}, 10^{-3.5}, \dots, 10^{-1}\}$ , and  $\gamma$  grid:  $\{0.0, 0.5, 0.7, 0.9, 0.95, 0.99\}$ , and report results in Figure 3. We plot the best performance of CC-FedSGD as a reference. We could see it is quite easy to tune the hyperparameters to obtain a much better performance compared to CC-FedSGD, e.g.,  $\beta \geq 0.7$ ,  $\epsilon \in [10^{-3.5}, 10^{-1.5}]$  and  $\gamma$  is flexible.

## 6 RELATED WORK

In this section, we discuss related work and how this paper develops on top of prior arts. We mainly review works on (1) Adaptive Optimization Approaches, (2) Server-Centric Federated Optimization, and (3) System Heterogeneity Aware FL.

### 6.1 Adaptive Optimization Approaches

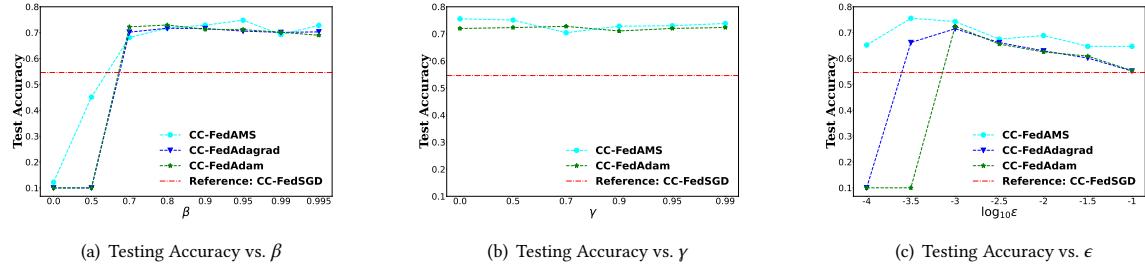
Machine learning models have been widely applied in many different domains, e.g. [14, 16, 23, 25, 53, 70, 72, 73, 81–83, 89], and the de facto optimizer for these ML models especially in the era of deep models is a large class of adaptive optimizers represented by Adam.

There is a wealth of work on adaptive optimization approaches in non-federated settings. As this paper is mainly focused on FL, we only provide a brief review of this line of research. SGD type optimizers rely heavily and sensitively on proper hyperparameters (learning rate, batch size) setup [21, 24, 30, 47, 63, 65, 66, 68, 69] and also motivated by the poor performances of SGD type optimizers in presence of sparse features and heavy-tail stochastic gradient noise distributions, adaptive optimizers, is shown to converge much faster than SGD in many applications [84]. Most adaptive optimizers share similar design, i.e. scale coordinates of the gradient by square roots of some form of averaging of the squared coordinates on a per-feature basis [59]. Most representative adaptive optimizers include AdaGrad [18], Adam [35], and their variants, e.g. RMSProp [71], AdaDelta [95], AdaBound [48], AMSGrad [59], AdaBelief [103], RAdam [45].

Note that most of the adaptive optimizers listed above can serve as a plug-and-play server-side module in our proposed client-centric federated adaptive optimization framework, similarly as how we construct CC-FedAdam. This paper shows the empirical and theoretical properties of three of them, i.e. Adam/AMSGrad/Adagrad.

### 6.2 Server-Centric Federated Optimization

As we explained in Section 1, most of the existing federated optimization researches belong to the server-centric FL regime, which naturally distinguish from this paper. We mainly review the works through the lens of how they mitigate *statistical heterogeneity*, which this paper also considers as a source of the *client drift*. We refer readers to [76] for a complete survey of federated optimization.



**Figure 3: Hyperparameter sensitivity of CC-Federated Adaptive Optimizers.** Note we do not plot CC-FedAdagrad in Figure 3(b), as there is no  $\gamma$  in CC-FedAdagrad.

FedAvg can diverge under high degrees of heterogeneity in worst case scenarios. To stabilize training even in presence of heterogeneity, the following approaches have been proposed, (1) penalize local models that are far away from the global model by regularizing the local objectives, e.g., FedProx [40], FedPD [96], and FedDyn [1]; (2) de-bias client's local model updates using techniques like control variates in SCAFFOLD [33] or local posterior sampling in FedPA [2]; and (3) treat local updates as pseudo-gradient and incorporate momentum/adaptive optimizers in either server or client optimization, e.g. FedAvgM [27], SlowMo [78], FedAdam [58], FedAMS [80], which are most related to this paper. These server-centric federated momentum/adaptive works largely ignore system heterogeneity, which this paper focuses on.

### 6.3 System Heterogeneity Aware FL

Existing works that study system heterogeneity can be categorized into the following classes,

(1) *Heterogeneous local computing but synchronous aggregation.* [77] first shows heterogeneous number of local updates results in a mismatched global convergence and proposes an approach called FedNova to effectively alleviate such mismatch. Other representative approaches in this class include [3, 6].

(2) *Flexible participation scheme but synchronous aggregation.* There are two themes of research in this class. The first line concentrates on biased client selection [11, 31, 55], most of which study the biased sampling strategy that the probability a client is sampled is related to its local loss to accelerate training [20, 60]. This line of works is less related to this paper. The second line studies different patterns of client participation [22, 61, 79, 90]. For example, FedLaAvg [90] and MIFA [22] both allow clients to participate arbitrarily as long as all devices participate once in the first round. But the clients are still strictly synchronized and the theoretical analysis of MIFA requires a Lipschitz Hessian assumption, which is too stringent and our analysis gets rid of such assumption.

(3) *Asynchronous aggregation.* This class is mostly related to this paper, but there are very limited works on asynchronous FL. Most related to ours are [54, 88, 92]. Specifically, FedAsync [88] allows the server immediately updates the global model whenever it receives a single local model to enable asynchrony. However, since the update from each individual client is no longer anonymized in an aggregate, FedAsync has privacy concerns. Moreover, their theoretical analysis only applies to convex objectives. FedBuff [54] and AFL [92] both propose to trigger a global update whenever the server receives  $m$  local updates to ensure anonymity. However,

FedBuff only considers homogeneous local computing. And both FedBuff and AFL ignore the lack of adaptivity issue.

In summary, the above existing works either relax only one of the unrealistic assumptions server-centric FL makes but not all of them, or require stringent assumptions in theoretical analysis.

## 7 CONCLUSION

In this paper, we proposed the Client-Centric Federated Adaptive Optimization, which is a class of novel federated optimization approaches. In contrast to most existing literature, we enable arbitrary client participation, asynchronous server aggregation, and heterogeneous local computing to fully characterize the ubiquitous system heterogeneity in real-world applications. We show both theoretically and empirically the convincing performances of our proposed algorithms. To our best knowledge, this is the first work that addresses statistical/system heterogeneity and lack of adaptivity issues simultaneously. The proposed client-centric FL regime is of independent interest for future FL algorithmic development as a more realistic testbed.

## REFERENCES

- [1] Durmus Alp Emre Acar, Yue Zhao, Ramon Matas, Matthew Mattina, Paul Whatmough, and Venkatesh Saligrama. 2021. Federated Learning Based on Dynamic Regularization. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=B7v4QMR6Z9w>
- [2] Maruan Al-Shedivat, Jennifer Gillenwater, Eric Xing, and Afshin Rostamizadeh. 2021. Federated Learning via Posterior Averaging: A New Perspective and Practical Algorithms. In *International Conference on Learning Representations (ICLR)*.
- [3] Dmitrii Avdiukhin and Shiva Kasiviswanathan. 2021. Federated Learning under Arbitrary Communication Patterns. In *Proceedings of the 38th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 139)*, Marina Meila and Tong Zhang (Eds.). PMLR, 425–435. <https://proceedings.mlr.press/v139/avdiukhin21a.html>
- [4] Runxue Bao, Bin Gu, and Heng Huang. 2020. Fast oscar and owl regression via safe screening rules. In *International conference on machine learning*. PMLR, 653–663.
- [5] Runxue Bao, Xidong Wu, Wenhan Xian, and Heng Huang. 2022. Doubly Sparse Asynchronous Learning. In *The 31st International Joint Conference on Artificial Intelligence (IJCAI 2022)*.
- [6] Debraj Basu, Deepesh Data, Can Karakus, and Suhas Diggavi. 2019. *Qsparse-Local-SGD: Distributed SGD with Quantization, Sparsification, and Local Computations*. Curran Associates Inc., Red Hook, NY, USA.
- [7] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.
- [8] PRESTON BUKATY. 2019. *The California Consumer Privacy Act (CCPA): An implementation guide*. IT Governance Publishing. <http://www.jstor.org/stable/j.ctvjghvn>
- [9] Liwei Che, Zewei Long, Jiaqi Wang, Yaqing Wang, Houping Xiao, and Fenglong Ma. 2021. FedTriNet: A pseudo labeling method with three players for federated semi-supervised learning. In *2021 IEEE International Conference on Big Data (Big Data)*. IEEE, 715–724.
- [10] Liwei Che, Jiaqi Wang, Yao Zhou, and Fenglong Ma. 2023. Multimodal federated learning: A survey. *Sensors* 23, 15 (2023), 6986.
- [11] Wenlin Chen, Samuel Horváth, and Peter Richtárik. 2020. Optimal Client Sampling for Federated Learning. *ArXiv abs/2010.13723* (2020).
- [12] Xiangyi Chen, Xiaoyun Li, and Ping Li. 2020. Toward Communication Efficient Adaptive Gradient Method. In *Proceedings of the 2020 ACM-IMS on Foundations of Data Science Conference (Virtual Event, USA) (FODS '20)*. Association for Computing Machinery, New York, NY, USA, 119–128. <https://doi.org/10.1145/3412815.3416891>
- [13] Xiangyi Chen, Sijia Liu, Ruoyu Sun, and Mingyi Hong. 2019. On the Convergence of A Class of Adam-Type Algorithms for Non-Convex Optimization. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=H1x-x309tm>
- [14] Heng-Tze Cheng, Levent Koc, Jeremiah Harmsen, Tal Shaked, Tushar Chandra, Hrishikesh B. Aradhya, Glen Anderson, Gregory S. Corrado, Wei Choi, Mustafa Ispir, Rohan Anil, Zakaria Haque, Lichan Hong, Vihan Jain, Xiaobing Liu, and Hemal Shah. 2016. Wide & Deep Learning for Recommender Systems. *Proceedings of the 1st Workshop on Deep Learning for Recommender Systems* (2016). <https://api.semanticscholar.org/CorpusID:3352400>
- [15] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
- [16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *North American Chapter of the Association for Computational Linguistics*. <https://api.semanticscholar.org/CorpusID:52967399>
- [17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=YicbFdNTTy>
- [18] John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive Subgradient Methods for Online Learning and Stochastic Optimization. *Journal of Machine Learning Research* 12, 61 (2011), 2121–2159. <http://jmlr.org/papers/v12/duchi11a.html>
- [19] European Commission. 2016. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance). <https://eur-lex.europa.eu/eli/reg/2016/679/oj>
- [20] Jack Goetz, Kshitiz Malik, Duc Viet Bui, Seungwhan Moon, Honglei Liu, and Anuj Kumar. 2019. Active Federated Learning. *ArXiv abs/1909.12641* (2019).
- [21] Priya Goyal, Piotr Dollár, Ross B. Girshick, Pieter Noordhuis, Lukasz Wesolowski, Apo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. 2017. Accurate, Large Minibatch SGD: Training ImageNet in 1 Hour. *CoRR abs/1706.02677* (2017). arXiv:1706.02677 <http://arxiv.org/abs/1706.02677>
- [22] Xinran Gu, Kaixuan Huang, Jingzhao Zhang, and Longbo Huang. 2021. Fast Federated Learning in the Presence of Arbitrary Device Unavailability. In *Advances in Neural Information Processing Systems*, A. Beygelzimer, Y. Dauphin, P. Liang, and J. Wortman Vaughan (Eds.). [https://openreview.net/forum?id=1\\_gaHbaRYt](https://openreview.net/forum?id=1_gaHbaRYt)
- [23] William L. Hamilton, Zhitao Ying, and Jure Leskovec. 2017. Inductive Representation Learning on Large Graphs. In *Neural Information Processing Systems*. <https://api.semanticscholar.org/CorpusID:4755450>
- [24] Fengxiang He, Tongliang Liu, and Dacheng Tao. 2019. Control Batch Size and Learning Rate to Generalize Well: Theoretical and Empirical Evidence. In *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 1143–1152.
- [25] Kaiming He, X. Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep Residual Learning for Image Recognition. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2016), 770–778.
- [26] K. He, X. Zhang, S. Ren, and J. Sun. 2016. Deep Residual Learning for Image Recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 770–778.
- [27] Tzu-Ming Harry Hsu, Qi, and Matthew Brown. 2019. Measuring the Effects of Non-Identical Data Distribution for Federated Visual Classification. *ArXiv abs/1909.06335* (2019).
- [28] Chun-Yin Huang, Kartik Srinivas, Xin Zhang, and Xiaoxiao Li. 2024. Overcoming Data and Model Heterogeneities in Decentralized Federated Learning via Synthetic Anchors. *arXiv preprint arXiv:2405.11525* (2024).
- [29] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger. 2017. Densely Connected Convolutional Networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2261–2269.
- [30] Stanisław Jastrzębski, Zaq Kenton, Devansh Arpit, Nicolas Ballas, Asja Fischer, Amos Storkey, and Yoshua Bengio. 2018. Three factors influencing minima in SGD. <https://openreview.net/forum?id=rJma2bZCW>
- [31] Yae Jee Cho, Jianyu Wang, and Gauri Joshi. 2022. Towards Understanding Biased Client Selection in Federated Learning. In *Proceedings of The 25th International Conference on Artificial Intelligence and Statistics (Proceedings of Machine Learning Research, Vol. 151)*, Gustau Camps-Valls, Francisco J. R. Ruiz, and Isabel Valera (Eds.). PMLR, 10351–10375. <https://proceedings.mlr.press/v151/jee-cho22a.html>
- [32] Peter Kairouz, H. Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista A. Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, Rafael G. L. D’Oliveira, Hubert Eichner, Salim El Rouayheb, David Evans, Josh Gardner, Zachary Garrett, Adrià Gascón, Badih Ghazi, Phillip B. Gibbons, Marco Gruteser, Zaid Harchaoui, Chaoyang He, Lie He, Zhouyuan Huo, Ben Hutchinson, Justin Hsu, Martin Jaggi, Tara Javid, Gauri Joshi, Mikhail Khodak, Jakub Konecný, Aleksandra Korolova, Farinaz Koushanfar, Sami Koyejo, Tancrède Lepoint, Yang Liu, Prateek Mittal, Mehryar Mohri, Richard Nock, Ayfer Özgür, Rasmus Pagh, Hang Qi, Daniel Ramage, Ramesh Raskar, Mariana Raykova, Dawn Song, Weikang Song, Sebastian U. Stich, Ziteng Sun, Ananda Theertha Suresh, Florian Tramèr, Praneeth Vepakomma, Jianyu Wang, Li Xiong, Zheng Xu, Qiang Yang, Feilix X. Yu, Han Yu, and Sen Zhao. 2021. Advances and Open Problems in Federated Learning. *Found. Trends Mach. Learn.* 14, 1–2 (2021), 1–210. <https://doi.org/10.1561/2200000083>
- [33] Sai Praneeth Karimireddy, Satyen Kale, Mehryar Mohri, Sashank Reddi, Sebastian Stich, and Ananda Theertha Suresh. 2020. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*. PMLR, 5132–5143.
- [34] Prashant Khanduri, Pranay Sharma, Haibo Yang, Mingyi Hong, Jia Liu, Ketan Rajawat, and Pramod Varshney. 2021. Stem: A stochastic two-sided momentum algorithm achieving near-optimal sample and communication complexities for federated learning. *Advances in Neural Information Processing Systems* 34 (2021).
- [35] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. *CoRR abs/1412.6980* (2015).
- [36] Alex Krizhevsky. 2009. Learning Multiple Layers of Features from Tiny Images.
- [37] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. 2018. Visualizing the Loss Landscape of Neural Nets. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems (Montréal, Canada) (NIPS'18)*. Curran Associates Inc., Red Hook, NY, USA, 6391–6401.
- [38] Qinbin Li, Yiqun Dia, Quan Chen, and Bingsheng He. 2022. Federated Learning on Non-IID Data Silos: An Experimental Study. In *IEEE International Conference on Data Engineering*.
- [39] Qinbin Li, Zeyi Wen, Zhaomin Wu, Sixu Hu, Naibo Wang, Yuan Li, Xu Liu, and Bingsheng He. 2021. A survey on federated learning systems: vision, hype and reality for data privacy and protection. *IEEE Transactions on Knowledge and Data Engineering* (2021).

- [40] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated Optimization in Heterogeneous Networks. In *Proceedings of Machine Learning and Systems*, I. Dhillon, D. Papailiopoulos, and V. Sze (Eds.), Vol. 2. 429–450.
- [41] Tian Li, Anit Kumar Sahu, Manzil Zaheer, Maziar Sanjabi, Ameet Talwalkar, and Virginia Smith. 2020. Federated optimization in heterogeneous networks. *Proceedings of Machine Learning and Systems* 2 (2020), 429–450.
- [42] Xiang Li, Kaixuan Huang, Wenhao Yang, Shusen Wang, and Zhihua Zhang. 2020. On the Convergence of FedAvg on Non-IID Data. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=HjxNAnVtDS>
- [43] Xinjin Li, Yu Ma, Yangchen Huang, Xingqi Wang, Yuzhen Lin, and Chenxi Zhang. 2024. Integrated Optimization of Large Language Models: Synergizing Data Utilization and Compression Techniques. (2024).
- [44] Xiangru Lian, Yijun Huang, Yunchen Li, and Ji Liu. 2015. Asynchronous Parallel Stochastic Gradient for Nonconvex Optimization. In *Proceedings of the 28th International Conference on Neural Information Processing Systems - Volume 2* (Montreal, Canada) (*NIPS'15*). MIT Press, Cambridge, MA, USA, 2737–2745.
- [45] Liyuan Liu, Haoming Jiang, Pengcheng He, Weizhu Chen, Xiaodong Liu, Jianfeng Gao, and Jiawei Han. 2020. On the Variance of the Adaptive Learning Rate and Beyond. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=rkgz2aEKDr>
- [46] Weidong Liu, Xiaojun Mao, Xiaofei Zhang, and Xin Zhang. 2024. Robust Personalized Federated Learning with Sparse Penalization. *J. Amer. Statist. Assoc.* (2024), 1–12.
- [47] Ben London. 2017. A PAC-Bayesian Analysis of Randomized Learning with Application to Stochastic Gradient Descent. In *NIPS*.
- [48] Liangchen Luo, Yuanhao Xiong, and Yan Liu. 2019. Adaptive Gradient Methods with Dynamic Bound of Learning Rate. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=Bkg3g2R9FX>
- [49] Yu Mao, Yufei Cui, Tei-Wei Kuo, and Chun Jason Xue. 2022. Accelerating general-purpose lossless compression via simple and scalable parameterization. In *Proceedings of the 30th ACM International Conference on Multimedia*. 3205–3213.
- [50] Yu Mao, Yufei Cui, Tei-Wei Kuo, and Chun Jason Xue. 2022. Trace: A fast transformer-based general-purpose lossless compressor. In *Proceedings of the ACM Web Conference 2022*. 1829–1838.
- [51] H. B. McMahan, Eider Moore, Daniel Ramage, Seth Hampson, and Blaise Agüera y Arcas. 2017. Communication-Efficient Learning of Deep Networks from Decentralized Data. In *International Conference on Artificial Intelligence and Statistics*.
- [52] Yongsheng Mei, Liangqi Yuan, Dong-Jun Han, Kevin S Chan, Christopher G Brinton, and Tian Lan. 2024. Using Diffusion Models as Generative Replay in Continual Federated Learning—What will Happen? *arXiv preprint arXiv:2411.06618* (2024).
- [53] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, Ioannis Antonoglou, Daan Wierstra, and Martin A. Riedmiller. 2013. Playing Atari with Deep Reinforcement Learning. *ArXiv abs/1312.5602* (2013).
- [54] John Nguyen, Kshitiz Malik, Hongyuan Zhan, Ashkan Yousefpour, Michael G. Rabbat, Mani Malek, and Dzmitry Huba. 2021. Federated Learning with Buffered Asynchronous Aggregation. In *International Conference on Artificial Intelligence and Statistics*.
- [55] Takayuki Nishio and Ryo Yonetani. 2018. Client Selection for Federated Learning with Heterogeneous Resources in Mobile Edge. *ICC 2019 - 2019 IEEE International Conference on Communications (ICC)* (2018), 1–7.
- [56] Qi Qi, Youzhi Luo, Zhao Xu, Shuiwang Ji, and Tianbao Yang. 2021. Stochastic optimization of areas under precision-recall curves with provable convergence. *Advances in neural information processing systems* 34 (2021), 1752–1765.
- [57] Qi Qi, Jiameng Lyu, Er Wei Bai, Tianbao Yang, et al. 2022. Stochastic constrained dro with a complexity independent of sample size. *arXiv preprint arXiv:2210.05740* (2022).
- [58] Sashank Reddi, Zachary Charles, Manzil Zaheer, Zachary Garrett, Keith Rush, Jakub Konečný, Sanjiv Kumar, and H Brendan McMahan. 2020. Adaptive federated optimization. *arXiv preprint arXiv:2003.00295* (2020).
- [59] Sashank J. Reddi, Satyen Kale, and Sanjiv Kumar. 2018. On the Convergence of Adam and Beyond. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=ryQu7f-RZ>
- [60] Mónica Ribero and Haris Vikalo. 2020. Communication-Efficient Federated Learning via Optimal Client Sampling. *ArXiv abs/2007.15197* (2020).
- [61] Yichen Ruan, Xiaoxi Zhang, Shu-Che Liang, and Carlee Joe-Wong. 2021. Towards Flexible Device Participation in Federated Learning. In *International Conference on Artificial Intelligence and Statistics*.
- [62] Karen Simonyan and Andrew Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. <http://arxiv.org/abs/1409.1556>
- [63] Samuel L. Smith, Pieter-Jan Kindermans, and Quoc V. Le. 2018. Don't Decay the Learning Rate, Increase the Batch Size. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=B1Yy1BxCZ>
- [64] Sebastian U. Stich. 2019. Local SGD Converges Fast and Communicates Little. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=S1g2JnRcFX>
- [65] Jianhui Sun, Mengdi Huai, Kishlay Jha, and Aidong Zhang. 2022. Demystify Hyperparameters for Stochastic Optimization with Transferable Representations. In *Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Washington DC, USA) (*KDD '22*). Association for Computing Machinery, New York, NY, USA, 1706–1716. <https://doi.org/10.1145/3534678.3539298>
- [66] Jianhui Sun, Sanchit Sinha, and Aidong Zhang. 2023. Enhance Diffusion to Improve Robust Generalization. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining* (Long Beach, CA, USA) (*KDD '23*). Association for Computing Machinery, New York, NY, USA, 2083–2095. <https://doi.org/10.1145/3580305.3599333>
- [67] Jianhui Sun, Xidong Wu, Heng Huang, and Aidong Zhang. 2024. On the role of server momentum in federated learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 38. 15164–15172.
- [68] Jianhui Sun, Ying Yang, Guangxu Xun, and Aidong Zhang. 2021. A Stagewise Hyperparameter Scheduler to Improve Generalization. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining* (Virtual Event, Singapore) (*KDD '21*). Association for Computing Machinery, New York, NY, USA, 1530–1540. <https://doi.org/10.1145/3447548.3467287>
- [69] Jianhui Sun, Ying Yang, Guangxu Xun, and Aidong Zhang. 2023. Scheduling Hyperparameters to Improve Generalization: From Centralized SGD to Asynchronous SGD. *ACM Trans. Knowl. Discov. Data* 17, 2, Article 29 (mar 2023), 37 pages. <https://doi.org/10.1145/3544782>
- [70] Qiaoling Suo, Liuyi Yao, Guangxu Xun, Jianhui Sun, and Aidong Zhang. 2019. Recurrent Imputation for Multivariate Time Series with Missing Values. In *2019 IEEE International Conference on Healthcare Informatics, ICHI 2019, Xi'an, China, June 10-13, 2019*. IEEE, 1–3. <https://doi.org/10.1109/ICHI.2019.8904638>
- [71] Tijmen Tieleman, Geoffrey Hinton, et al. 2012. Lecture 6.5-rmsprop: Divide the gradient by a running average of its recent magnitude. *COURSERA: Neural networks for machine learning* 4, 2 (2012), 26–31.
- [72] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems* 30 (2017).
- [73] Haohui Wang, Yuzhen Mao, Yujun Yan, Yaoqing Yang, Jianhui Sun, Kevin Choi, Balaji Veeramani, Alison Hu, Edward Bowen, Tyler Cody, and Dawei Zhou. 2025. EvoluNet: advancing dynamic non-IID transfer learning on graphs. In *Proceedings of the 41st International Conference on Machine Learning* (Vienna, Austria) (*ICML '24*). JMLR.org, Article 2095, 19 pages.
- [74] Han Wang, Jun Tang, Xiaodong Liu, Shanyan Guan, Rong Xie, and Li Song. 2022. Ptformer: Progressive temporal-spatial enhanced transformer towards video object detection. In *European Conference on Computer Vision*. Springer, 732–747.
- [75] Haoyu Wang, Handong Zhao, Yaqing Wang, Tong Yu, Juxiang Gu, and Jing Gao. 2022. FedKC: Federated knowledge composition for multilingual natural language understanding. In *Proceedings of the ACM Web Conference 2022*. 1839–1850.
- [76] Jianyu Wang, Zachary B. Charles, Zheng Xu, Gauri Joshi, H. B. McMahan, Blaise Agüera y Arcas, Maruan Al-Shedivat, Galen Andrew, Salman Avestimehr, Katharine Daly, Deepesh Data, Suhas N. Diggavi, Hubert Eichner, Advait Gadhiraju, Zachary Garrett, Antonios M. Gligorić, Filip Hanzeley, Andrew Hard, Chaoyang He, Samuel Horváth, Zhouyuan Huo, Alex Ingberman, Martin Jaggi, Tara Javidi, Peter Kairouz, Sathyen Kale, Sai Praneeth Karimireddy, Jakub Konečný, Sammi Koyejo, Tian Li, Luyang Liu, Mehryar Mohri, Hang Qi, Sashank J. Reddi, Peter Richtárik, K. Singhal, Virginia Smith, Mahdi Soltanolkotabi, Weikang Song, Ananda Theertha Suresh, Sebastian U. Stich, Ameet S. Talwalkar, Hongyi Wang, Blake E. Woodworth, Shanshan Wu, Felix X. Yu, Honglin Yuan, Manzil Zaheer, Mi Zhang, Tong Zhang, Chunxiang Zheng, Chen Zhu, and Wennan Zhu. 2021. A Field Guide to Federated Optimization. *ArXiv abs/2107.06917* (2021).
- [77] Jianyu Wang, Qinghua Liu, Hao Liang, Gauri Joshi, and H. Vincent Poor. 2020. Tackling the Objective Inconsistency Problem in Heterogeneous Federated Optimization. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (Vancouver, BC, Canada) (*NIPS'20*). Curran Associates Inc., Red Hook, NY, USA, Article 638, 13 pages.
- [78] Jianyu Wang, Vinayak Tantia, Nicolas Ballas, and Michael Rabbat. 2020. SlowMo: Improving Communication-Efficient Distributed SGD with Slow Momentum. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=Slkj8REYPh>
- [79] Shiqiang Wang and Mingyue Ji. 2022. A Unified Analysis of Federated Learning with Arbitrary Client Participation. In *Advances in Neural Information Processing Systems*, Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho (Eds.). <https://openreview.net/forum?id=qSs7C7c4G8D>
- [80] Yujia Wang, Lu Lin, and Jinghui Chen. 2022. Communication-Efficient Adaptive Federated Learning. In *Proceedings of the 39th International Conference on*

- Machine Learning (Proceedings of Machine Learning Research, Vol. 162)*. PMLR, 22802–22838. <https://proceedings.mlr.press/v162/wang22o.html>
- [81] Tianxin Wei, Bowen Jin, Ruirui Li, Hansi Zeng, Zhengyang Wang, Jianhui Sun, Qingyu Yin, Hanqing Lu, Suhang Wang, Jingrui He, et al. 2023. Towards universal multi-modal personalization: A language model empowered generative paradigm. In *The Twelfth International Conference on Learning Representations*.
  - [82] Jiawei Wen, Songshan Yang, Chris Wang, Yishen Jiang, and Runze Li. 2023. Feature-splitting algorithms for ultrahigh dimensional quantile regression. *Journal of Econometrics* (2023). <https://api.semanticscholar.org/CorpusID:257747996>
  - [83] Jiawei Wen, Songshan Yang, and Delin Zhao. 2024. Nonconvex Dantzig selector and its parallel computing algorithm. *Statistics and Computing* 34, 6 (Sept. 2024), 21 pages. <https://doi.org/10.1007/s11222-024-10492-8>
  - [84] Ashia C Wilson, Rebecca Roelofs, Mitchell Stern, Nati Srebro, and Benjamin Recht. 2017. The Marginal Value of Adaptive Gradient Methods in Machine Learning. In *Advances in Neural Information Processing Systems 30*. Curran Associates, Inc., 4148–4158.
  - [85] Xidong Wu, Jianhui Sun, Zhengmian Hu, Junyi Li, Aidong Zhang, and Heng Huang. 2023. Federated Conditional Stochastic Optimization. In *Advances in Neural Information Processing Systems*, A. Oh, T. Naumann, A. Globerson, K. Saenko, M. Hardt, and S. Levine (Eds.), Vol. 36. Curran Associates, Inc., 5752–5764.
  - [86] Xidong Wu, Jianhui Sun, Zhengmian Hu, Aidong Zhang, and Heng Huang. 2024. Solving a class of non-convex minimax optimization in federated learning. *Advances in Neural Information Processing Systems* 36 (2024).
  - [87] Han Xiao, Kashif Rasul, and Roland Vollgraf. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. *ArXiv* abs/1708.07747 (2017).
  - [88] Cong Xie, Oluwasanmi Koyejo, and Indranil Gupta. 2019. Asynchronous Federated Optimization. *ArXiv* abs/1903.03934 (2019).
  - [89] Guangxi Xun, Kishlay Jha, Jianhui Sun, and Aidong Zhang. 2020. Correlation Networks for Extreme Multi-label Text Classification. *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (2020).
  - [90] Yikai Yan, Chaoyue Niu, Yucheng Ding, Zhenzhe Zheng, Fan Wu, Guihai Chen, Shaojie Tang, and Zhihua Wu. 2020. Distributed Non-Convex Optimization with Sublinear Speedup under Intermittent Client Availability. *ArXiv* abs/2002.07399 (2020).
  - [91] Haibo Yang, Minghong Fang, and Jia Liu. 2021. Achieving Linear Speedup with Partial Worker Participation in Non-IID Federated Learning. In *International Conference on Learning Representations*. <https://openreview.net/forum?id=jDdzhs5ul-d>
  - [92] Haibo Yang, Xin Zhang, Prashant Khanduri, and Jia Liu. 2021. Anarchic Federated Learning. In *International Conference on Machine Learning*.
  - [93] Mikhail Yurochkin, Mayank Agarwal, Soumya Shubhra Ghosh, Kristjan H. Greenewald, Trong Nghia Hoang, and Yasaman Khazaeni. 2019. Bayesian Non-parametric Federated Learning of Neural Networks. In *International Conference on Machine Learning*.
  - [94] Manzil Zaheer, Sashank J. Reddi, Devendra Sachan, Satyen Kale, and Sanjiv Kumar. 2018. Adaptive Methods for Nonconvex Optimization. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems* (Montréal, Canada) (NIPS’18). Curran Associates Inc., Red Hook, NY, USA, 9815–9825.
  - [95] Matthew D. Zeiler. 2012. ADADELTA: An Adaptive Learning Rate Method. *ArXiv* abs/1212.5701 (2012).
  - [96] Xinwei Zhang, Mingyi Hong, Sairaj Dhople, Wotao Yin, and Yang Liu. 2020. Fedpd: A federated learning framework with optimal rates and adaptivity to non-iid data. *arXiv preprint arXiv:2005.11418* (2020).
  - [97] Zeyu Zhang, Thuy Vu, and Alessandro Moschitti. 2021. Joint models for answer verification in question answering systems. *arXiv preprint arXiv:2107.04217* (2021).
  - [98] Ce Zheng, Xianpeng Liu, Guo-Jun Qi, and Chen Chen. 2023. Potter: Pooling attention transformer for efficient human mesh recovery. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 1611–1620.
  - [99] Jiayun Zheng and Maggie Makar. 2022. Causally motivated multi-shortcut identification and removal. *Advances in Neural Information Processing Systems* 35 (2022), 12800–12812.
  - [100] Hanhan Zhou, Tian Lan, and Vaneet Aggarwal. 2022. PAC: Assisted Value Factorization with Counterfactual Predictions in Multi-Agent Reinforcement Learning. *Advances in Neural Information Processing Systems (NeurIPS)* 36 (2022), 15757–15769.
  - [101] Hanhan Zhou, Tian Lan, Guru Prasad Venkataramani, and Wenbo Ding. 2022. Federated learning with online adaptive heterogeneous local models. In *Workshop on Federated Learning: Recent Advances and New Challenges (in Conjunction with NeurIPS 2022)*.
  - [102] Hanhan Zhou, Tian Lan, Guru Prasad Venkataramani, and Wenbo Ding. 2023. Every parameter matters: Ensuring the convergence of federated learning with dynamic heterogeneous models reduction. *Advances in Neural Information Processing Systems* 37 (2023).

## Appendix

In Section A, we discuss key model designs omitted from Section 3.2.2. In Section B, we provide complete proof of Theorem 4.1 and Corollary 4.1.1. In Section C, we provide more experimental results which are omitted from main text.

### A KEY ALGORITHMIC DESIGNS

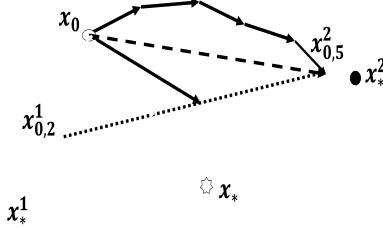
We would like to summarize the following unique features of Algorithm 2 that distinguish itself from server-centric FL,

- Global update by server happens concurrently with local update by client, which avoids low-capacity clients from straggling training or any system locking.
- Client participates whenever it intends to, and each client can self-determine a time-varying and device-dependent number of local updates  $K_{t,i}$ .
- Each participating client can work with an asynchronous view of the global model from an outdated timestamp  $t - \tau_{t,i}$ .

We highlight several algorithmic designs that are key to Algorithm 2 in this section.

#### (a) Normalized Model Update

In CC-FedAMS, we allow a time-varying and device-dependent  $K_{t,i}$ , which would result in the global update biased towards the client with larger  $K_{t,i}$ . Figure 4 demonstrates this phenomenon with a two-client toy example. Specifically, suppose client 1 and client 2 carry out two and five local updates, and reach  $x_{0,2}^1$  and  $x_{0,5}^2$  (following the same  $x_{t,k}^i$  notation in Algorithm 2), respectively.  $x_*$ ,  $x_*^1$ , and  $x_*^2$  denote the global optimum, local optimum for client 1 and 2, respectively. As is evident in this example,  $x_{0,5}^2$  would likely drag the global update (red solid arrow) more towards  $x_*^2$  since client 2 takes more local updates and  $x_0 - x_{0,5}^2$  thus has a larger scale. This will make the convergence towards global optimum  $x_*$  slowing down or even failing in worst case [77].



**Figure 4: A toy example of a two-client FL setting. There is a mismatch between the ideal direction towards global optimum (green dashed arrow) and the actual search direction (red solid line) if the two clients take a different number of local updates.**

Therefore, before sending the model difference to the server, we request each client normalizes the model update by  $K_{t,i}$ , i.e.,  $\Delta_\mu^i = \frac{x_{\mu,0}^i - x_{\mu,K_{t,i}}^i}{K_{t,i}}$ .

#### (b) Size $m$ Buffer $S_t$

In CC-FedAMS, we concurrently run server and client computations. The server maintains a “buffer” of responsive clients  $S_t$  and only triggers global update once the buffer reaches size  $m$ . There is a fundamental trade-off in the selection of  $m$ . Specifically, if  $m = 1$  as in [88], the global update takes place once one client responds, which has the fastest updating frequency and avoids the idle time to collect a size  $m$  buffer. However,  $m = 1$  has the following significant drawbacks. First, it reveals the model update from one single client, which has profound privacy concerns. One of the most important motivations for FL is to anonymize any single client by server-side aggregation, which  $m = 1$  clearly violates. Second, it is more vulnerable to client drift.  $m = 1$  indicates the global update relies entirely on one single client at each round, and the global convergence would be slowdown if that client is subject to dramatic distributional shift. Third, too fast global update results in more chaotic client behaviors. Too frequent global update would make clients up to a larger delay, and potentially negatively impact the convergence.<sup>7</sup> Empirically, we find an  $m > 1$  setting converges much faster than  $m = 1$  in our experiments. ( $m = 5$  when there are 100 clients in our experiments is a good setting across benchmarks and does not need tuning).

#### (c) Server-side Adaptive Optimization

In CC-FedAMS, a critical ingredient is the server side AMSGrad-type global update. We would like to highlight why adaptive optimization is necessary here. (1) Adaptivity is crucial when training large-scale overparameterized models. As was explained in Section 1, adaptively

<sup>7</sup>In Section 4 Corollary 4.1.1, we can see the convergence rate has a term  $O\left(\frac{\tau^2}{T}\right)$ , which theoretically validates larger randomly delay  $\tau$  slows down global convergence and thus too small  $m$  may be harmful.

scaling learning rate on a per-feature basis could accelerate convergence with sparse feature space. (2) Adaptive optimization helps mitigate client drift. Note that the design of adaptive optimizer leverages the average of historical gradient information, e.g., it maintains a momentum by recursively aggregating historical model updates  $m_t = (1 - \beta)\Delta_t + \beta m_{t-1}$  and it scales coordinates of the gradient by square roots of the recursive averaging of the squared coordinates  $x_{t+1} = x_t - \eta \frac{m_t}{\sqrt{\hat{v}_t + \epsilon}}$ . If without adaptive optimization, the global update would rely entirely on the averaged current model update, which is subject to unexpected drift. When clients are dynamically heterogeneous, historical gradient information is virtually a regularizer to restrain the search direction from going wild.

## B PROOF OF THEOREM 4.1 AND COROLLARY 4.1.1

### B.1 Proof of Theorem 4.1

PROOF OF THEOREM 4.1. We introduce a Lyapunov sequence  $\{z_t\}_{t=0}^{T-1}$  which is devised as follows:

$$z_t = x_t + \frac{\beta}{1 - \beta} (x_t - x_{t-1}) \quad (4)$$

We could verify<sup>8</sup>,

$$\begin{aligned} z_{t+1} - z_t &= \frac{1}{1 - \beta} (x_{t+1} - x_t) - \frac{\beta}{1 - \beta} (x_t - x_{t-1}) \\ &= \frac{1}{1 - \beta} \left( \eta \frac{d_{t+1}}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) - \frac{\beta}{1 - \beta} \left( \eta \frac{d_t}{\sqrt{\hat{v}_t} + \epsilon} \right) \\ &= \frac{\eta}{1 - \beta} \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} ((1 - \beta) \Delta_t + \beta d_t) - \frac{\beta}{1 - \beta} \eta \frac{1}{\sqrt{\hat{v}_t} + \epsilon} d_t \\ &= \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} - \eta \frac{\beta}{1 - \beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \end{aligned}$$

Since  $f$  is  $L$ -smooth, taking conditional expectation with respect to all randomness prior to step  $t$ , we have

$$\begin{aligned} \mathbb{E}[f(z_{t+1})] &\leq f(z_t) + \mathbb{E}[\langle \nabla f(z_t), z_{t+1} - z_t \rangle] + \frac{L}{2} \mathbb{E}[\|z_{t+1} - z_t\|^2] \\ &\leq f(z_t) + \mathbb{E}\left[\left\langle \nabla f(z_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} - \eta \frac{\beta}{1 - \beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\rangle\right] \\ &\quad + \frac{L\eta^2}{2} \mathbb{E}\left[\left\| \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} - \frac{\beta}{1 - \beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\|^2\right] \\ &\leq f(z_t) + \underbrace{\mathbb{E}\left[\left\langle \nabla f(z_t) - \nabla f(x_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\rangle\right]}_{A_1} + \underbrace{\mathbb{E}\left[\left\langle \nabla f(x_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\rangle\right]}_{A_2} \\ &\quad + \underbrace{\mathbb{E}\left[\left\langle \nabla f(z_t), -\eta \frac{\beta}{1 - \beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\rangle\right]}_{A_3} \\ &\quad + \underbrace{\frac{L\eta^2}{2} \mathbb{E}\left[\left\| \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} - \frac{\beta}{1 - \beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\|^2\right]}_{A_4} \end{aligned}$$

---

<sup>8</sup>Momentum buffer  $d_t$  here is  $m_t$  in main text, also in Appendix, the zero-index of time  $t \in \{0, 1, \dots, T-1\}$  is equivalent to the one-index of time  $t \in \{1, \dots, T\}$  in main text.

**Bounding  $A_1$ :**

$$\begin{aligned}
A_1 &= \mathbb{E} \left[ \left\langle \nabla f(z_t) - \nabla f(x_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\rangle \right] \\
&\stackrel{(i)}{\leq} \mathbb{E} \left[ \|\nabla f(z_t) - \nabla f(x_t)\| \cdot \left\| \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right] \\
&\stackrel{(ii)}{\leq} L \mathbb{E} \left[ \|z_t - x_t\| \cdot \left\| \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right] \stackrel{(iii)}{\leq} \frac{\eta^2 L}{2} \mathbb{E} \left[ \left\| \frac{\beta}{1-\beta} \frac{d_t}{\sqrt{\hat{v}_t} + \epsilon} \right\|^2 \right] + \frac{\eta^2 L}{2} \mathbb{E} \left[ \left\| \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\|^2 \right] \\
&\stackrel{(iv)}{\leq} \frac{\eta^2 L}{2\epsilon^2} \left( \frac{\beta}{1-\beta} \right)^2 \mathbb{E} [\|d_t\|^2] + \frac{\eta^2 L}{2\epsilon^2} \mathbb{E} [\|\Delta_t\|^2]
\end{aligned}$$

where (i) holds by applying Cauchy-Schwarz inequality, (ii) holds as  $f$  is  $L$ -smooth, (iii) follows from Young's inequality and the definition of  $z_t$ , and (iv) holds as  $\sqrt{\hat{v}_t} + \epsilon \geq \epsilon$ .

**Bounding  $A_2$ :**

$$\begin{aligned}
A_2 &= \mathbb{E} \left[ \left\langle \nabla f(x_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\rangle \right] \\
&= \underbrace{\mathbb{E} \left[ \left\langle \nabla f(x_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} - \eta \frac{\Delta_t}{\sqrt{\hat{v}_t} + \epsilon} \right\rangle \right]}_{A_5} + \underbrace{\mathbb{E} \left[ \left\langle \nabla f(x_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_t} + \epsilon} \right\rangle \right]}_{A_6}
\end{aligned}$$

We could bound  $A_5$  as follows,

$$\begin{aligned}
A_5 &= \mathbb{E} \left[ \left\langle \nabla f(x_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} - \eta \frac{\Delta_t}{\sqrt{\hat{v}_t} + \epsilon} \right\rangle \right] \\
&\stackrel{(i)}{\leq} \eta \|\nabla f(x_t)\| \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} - \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \right\| \cdot \|\Delta_t\| \right] \\
&\leq \eta \|\nabla f(x_t)\| \mathbb{E} \left[ \left\| \frac{(\sqrt{\hat{v}_{t+1}} - \sqrt{\hat{v}_t})(\sqrt{\hat{v}_{t+1}} + \sqrt{\hat{v}_t})}{(\sqrt{\hat{v}_{t+1}} + \epsilon)(\sqrt{\hat{v}_t} + \epsilon)(\sqrt{\hat{v}_{t+1}} + \sqrt{\hat{v}_t})} \right\| \cdot \|\Delta_t\| \right] \\
&\stackrel{(ii)}{\leq} \eta \|\nabla f(x_t)\| \mathbb{E} \left[ \left\| \frac{\|\Delta_t\|^2}{(2\epsilon)^2 \|\Delta_t\|} \right\| \cdot \|\Delta_t\| \right] \\
&\stackrel{(iii)}{\leq} \eta \cdot G \cdot \frac{1}{4\epsilon^2} \mathbb{E} [\|\Delta_t\|^2]
\end{aligned}$$

where (i) holds due to Cauchy-Schwarz inequality, (ii) holds from the definition of  $\hat{v}_{t+1}$  (&  $\hat{v}_t$ ) and the fact that  $\hat{v}_t$  is non-decreasing and  $\hat{v}_t \geq \dots \geq \hat{v}_0 \geq \epsilon^2$ , (iii) holds due to bounded gradient assumption.

We could bound  $A_6$  as follows,

$$\begin{aligned}
A_6 &= \mathbb{E} \left[ \left\langle \nabla f(x_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_t} + \epsilon} \right\rangle \right] \\
&= \eta \mathbb{E} \left[ \left\langle \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t} + \epsilon}, \Delta_t + \eta_l \nabla f(x_t) - \eta_l \nabla f(x_t) \right\rangle \right] = -\eta \eta_l \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t} + \epsilon} \right\|^2 \right] \\
&+ \eta \eta_l \mathbb{E} \left[ \left\langle \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t} + \epsilon}, \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \left( \nabla f(x_t) - \frac{1}{m} \sum_{i \in S_t} \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} g_{t-\tau_{t,i},k}^i \right) \right\rangle \right] \\
&\quad \underbrace{}_{A_7}
\end{aligned}$$

We could bound  $A_7$  as follows,

$$\begin{aligned}
A_7 &= \eta \eta_l \mathbb{E} \left[ \left\langle \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t} + \epsilon}, \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \left( \nabla f(x_t) - \frac{1}{m} \sum_{i \in S_t} \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} g_{t-\tau_{t,i},k}^i \right) \right\rangle \right] \\
&\stackrel{(i)}{=} \eta \eta_l \mathbb{E} \left[ \left\langle \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t} + \epsilon}, \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \left( \nabla f(x_t) - \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right) \right\rangle \right]
\end{aligned}$$

where (i) holds as we take conditional expectation with respect to all randomness prior to step  $t$ .

We further have,

$$\begin{aligned}
A_7 &= \frac{\eta\eta_l}{2} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t + \epsilon}} \right\|^2 \right] - \frac{\eta\eta_l}{2} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t + \epsilon}} \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\
&\quad + \frac{\eta\eta_l}{2} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t + \epsilon}} \left( \nabla f(x_t) - \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right) \right\|^2 \right] \\
&\stackrel{(i)}{\leq} \frac{\eta\eta_l}{2} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t + \epsilon}} \right\|^2 \right] - \frac{\eta\eta_l}{2(\sqrt{T}\eta_l G + \epsilon)} \mathbb{E} \left[ \left\| \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\
&\quad + \underbrace{\frac{\eta\eta_l}{2} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t + \epsilon}} \left( \nabla f(x_t) - \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right) \right\|^2 \right]}_{A_8}
\end{aligned}$$

where (i) holds as  $\hat{v}_t \leq T\eta_l^2 G^2$ .

We further bound  $A_8$  as follows,

$$\begin{aligned}
A_8 &= \frac{\eta\eta_l}{2} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t + \epsilon}} \left( \nabla f(x_t) - \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right) \right\|^2 \right] \leq \frac{\eta\eta_l}{2\epsilon} \\
&\mathbb{E} \left[ \left\| \left( \nabla f(x_t) - \frac{1}{n} \sum_{i=1}^n \nabla f_i(x_{t-\tau_{t,i}}) \right) + \left( \frac{1}{n} \sum_{i=1}^n \nabla f_i(x_{t-\tau_{t,i}}) - \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right) \right\|^2 \right] \\
&\stackrel{(i)}{\leq} \frac{\eta\eta_l}{\epsilon} \mathbb{E} \left[ \left\| \frac{1}{n} \sum_{i=1}^n \nabla f_i(x_t) - \frac{1}{n} \sum_{i=1}^n \nabla f_i(x_{t-\tau_{t,i}}) \right\|^2 \right] \\
&\quad + \frac{\eta\eta_l}{\epsilon} \mathbb{E} \left[ \left\| \frac{1}{n} \sum_{i=1}^n \nabla f_i(x_{t-\tau_{t,i}}) - \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\
&\stackrel{(ii)}{\leq} \frac{\eta\eta_l L^2}{\epsilon} \frac{1}{n} \sum_{i=1}^n \mathbb{E} [\|x_t - x_{t-\tau_{t,i}}\|^2] + \frac{\eta\eta_l L^2}{\epsilon} \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \mathbb{E} [\|x_{t-\tau_{t,i}} - x_{t-\tau_{t,i},k}^i\|^2]
\end{aligned}$$

where (i) holds due to  $\|\sum_{i=1}^n x_i\|^2 \leq n \sum_{i=1}^n \|x_i\|^2$ , (ii) holds due to  $L$ -smoothness of  $f_i$ .

When  $\eta_l \leq \frac{1}{8KL}$ , we have,

$$\mathbb{E} \left[ \|x_{t-\tau_{t,i}} - x_{t-\tau_{t,i},k}^i\|^2 \right] \leq 5K_{t,i}\eta_l^2 (\sigma_l^2 + 6K_{t,i}\sigma_g^2) + 30K_{t,i}^2\eta_l^2 \mathbb{E} [\|\nabla f(x_{t-\tau_{t,i}})\|^2]$$

We can further bound  $\frac{1}{n} \sum_{i=1}^n \mathbb{E} [\|x_t - x_{t-\tau_{t,i}}\|^2]$

$$\begin{aligned}
\frac{1}{n} \sum_{i=1}^n \mathbb{E} [\|x_t - x_{t-\tau_{t,i}}\|^2] &\leq \mathbb{E} [\|x_t - x_{t-\tau_{t,u}}\|^2] = \mathbb{E} \left[ \left\| \sum_{k=t-\tau_{t,u}}^{t-1} (x_{k+1} - x_k) \right\|^2 \right] \\
&\leq \mathbb{E} \left[ \left\| \sum_{k=t-\tau_{t,u}}^{t-1} \eta y_k \right\|^2 \right] \leq \tau\eta^2 \sum_{k=t-\tau_{t,u}}^{t-1} \mathbb{E} [\|y_k\|^2]
\end{aligned}$$

where  $y_k \triangleq x_{k+1} - x_k = \eta \frac{d_{k+1}}{\sqrt{\hat{v}_{k+1} + \epsilon}}$ .

$$\begin{aligned} A_8 &\leq \frac{\eta^3 \eta_l L^2 \tau}{\epsilon} \sum_{k=t-\tau_{t,u}}^{t-1} \mathbb{E} [\|y_k\|^2] + \frac{5\eta \eta_l^3 L^2 \bar{K}_t}{\epsilon} \sigma_l^2 + \frac{30\eta \eta_l^3 L^2 \hat{K}_t^2}{\epsilon} \sigma_g^2 \\ &\quad + \frac{30\eta \eta_l^3 L^2}{\epsilon} \frac{1}{n} \sum_{i=1}^n K_{t,i}^2 \mathbb{E} [\|\nabla f(x_{t-\tau_{t,i}})\|^2] \end{aligned}$$

Merging all pieces together,

$$\begin{aligned} A_2 &= \mathbb{E} \left[ \left\langle \nabla f(x_t), \eta \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\rangle \right] \\ &\leq \frac{\eta G}{4\epsilon^2} \mathbb{E} [\|\Delta_t\|^2] - \frac{\eta \eta_l}{2} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t} + \epsilon} \right\|^2 \right] - \frac{\eta \eta_l}{2(\sqrt{T} \eta_l G + \epsilon)} \mathbb{E} \left[ \left\| \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\ &\quad + \frac{\eta^3 \eta_l L^2 \tau}{\epsilon} \sum_{k=t-\tau_{t,u}}^{t-1} \mathbb{E} [\|y_k\|^2] + \frac{5\eta \eta_l^3 L^2 \bar{K}_t}{\epsilon} \sigma_l^2 + \frac{30\eta \eta_l^3 L^2 \hat{K}_t^2}{\epsilon} \sigma_g^2 + \frac{30\eta \eta_l^3 L^2}{\epsilon} \frac{1}{n} \sum_{i=1}^n K_{t,i}^2 \mathbb{E} [\|\nabla f(x_{t-\tau_{t,i}})\|^2] \end{aligned}$$

**Bounding  $A_3$ :**

$$\begin{aligned} A_3 &= \mathbb{E} \left[ \left\langle \nabla f(z_t), -\eta \frac{\beta}{1-\beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\rangle \right] \\ &\leq \mathbb{E} \left[ \left\langle \nabla f(z_t) - \nabla f(x_t) + \nabla f(x_t), -\eta \frac{\beta}{1-\beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\rangle \right] \\ &\stackrel{(i)}{\leq} \eta \mathbb{E} \left[ \|\nabla f(z_t) - \nabla f(x_t)\| \cdot \left\| \frac{\beta}{1-\beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\| \right] \\ &\quad + \eta \mathbb{E} \left[ \|\nabla f(x_t)\| \cdot \left\| \frac{\beta}{1-\beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\| \right] \end{aligned}$$

where (i) holds due to Cauchy-Schwarz inequality.

And we would further have,

$$\begin{aligned} A_3 &\stackrel{(i)}{\leq} \eta L \mathbb{E} \left[ \left\| \frac{\beta}{1-\beta} \left( \eta \frac{d_t}{\sqrt{\hat{v}_t} + \epsilon} \right) \right\| \cdot \left\| \frac{\beta}{1-\beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\| \right] \\ &\quad + \eta G \mathbb{E} \left[ \left\| \frac{\beta}{1-\beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\| \right] \\ &\stackrel{(ii)}{\leq} \eta^2 \eta_l^2 G^2 L \left( \frac{\beta}{1-\beta} \right)^2 \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \right\| \cdot \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right] \\ &\quad + \eta \eta_l G^2 \frac{\beta}{1-\beta} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right] \\ &\stackrel{(iii)}{\leq} \eta^2 \eta_l^2 G^2 L \left( \frac{\beta}{1-\beta} \right)^2 \frac{1}{2\epsilon} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right] + \eta \eta_l G^2 \frac{\beta}{1-\beta} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right] \\ &\leq \left( \frac{\eta^2 \eta_l^2 G^2 L}{2\epsilon} \left( \frac{\beta}{1-\beta} \right)^2 + \eta \eta_l G^2 \frac{\beta}{1-\beta} \right) \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right] \end{aligned}$$

where (i), (ii) and (iii) hold as we have  $\|d_t\| \leq \eta_l G$ ,  $\|\Delta_t\| \leq \eta_l G$ ,  $\left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \right\| \leq \frac{1}{2\epsilon}$ ,  $\mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right] \leq \frac{1}{4\epsilon^2} \mathbb{E} [\|\Delta_t\|]$ .

**Bounding  $A_4$ :**

$$\begin{aligned} A_4 &= \frac{L\eta^2}{2} \mathbb{E} \left[ \left\| \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} - \frac{\beta}{1-\beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\|^2 \right] \\ &\stackrel{(i)}{\leq} L\eta^2 \mathbb{E} \left[ \left\| \frac{\Delta_t}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\|^2 \right] + L\eta^2 \mathbb{E} \left[ \left\| \frac{\beta}{1-\beta} \left( \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right) d_t \right\|^2 \right] \\ &\stackrel{(ii)}{\leq} \frac{L\eta^2}{4\epsilon^2} \mathbb{E} [\|\Delta_t\|^2] + L\eta^2 \eta_l^2 G^2 \left( \frac{\beta}{1-\beta} \right)^2 \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\|^2 \right] \end{aligned}$$

where (i) holds due to Cauchy-Schwarz inequality, (ii) holds as  $\|d_t\| \leq \eta_l G$ .

**Bounding**  $\sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\|^2 \right]$ :

$$\begin{aligned} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\|^2 \right] & \stackrel{(i)}{=} \sum_{t=0}^{T-1} \sum_{j=1}^d \mathbb{E} \left[ \left( \left[ \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \right]_j - \left[ \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right]_j \right)^2 \right] \\ & \leq \sum_{t=0}^{T-1} \sum_{j=1}^d \mathbb{E} \left[ \left[ \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \right]_j^2 - \left[ \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right]_j^2 \right] \\ & \stackrel{(ii)}{\leq} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_0} + \epsilon} \right\|^2 \right] - \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_T} + \epsilon} \right\|^2 \right] \stackrel{(iii)}{\leq} \frac{d}{4\epsilon^2} \end{aligned}$$

where  $\left[ \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \right]_j$  denotes the  $j$ -th element of vector  $\frac{1}{\sqrt{\hat{v}_t} + \epsilon}$  and (i) follows from the definition of  $\|\cdot\|^2$ . (ii) holds as  $\hat{v}_t$  is non-decreasing, (iii) holds as every element of  $\frac{1}{\sqrt{\hat{v}_0} + \epsilon}$  is smaller than  $\frac{1}{2\epsilon}$ .

**Bounding**  $\sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right]$ :

$$\begin{aligned} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\| \right] & = \sum_{t=0}^{T-1} \sum_{j=1}^d \mathbb{E} \left[ \left| \left[ \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \right]_j - \left[ \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right]_j \right| \right] \\ & \leq \sum_{t=0}^{T-1} \sum_{j=1}^d \mathbb{E} \left[ \left| \left[ \frac{1}{\sqrt{\hat{v}_t} + \epsilon} \right]_j - \left[ \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right]_j \right| \right] \\ & \stackrel{(i)}{\leq} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_0} + \epsilon} \right\| \right] - \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_T} + \epsilon} \right\| \right] \leq \frac{d}{2\epsilon} \end{aligned}$$

Similarly, (i) holds due to  $\hat{v}_t$  is non-decreasing.

**Bounding**  $\mathbb{E} [\|\Delta_t\|^2]$ :

$$\begin{aligned} \mathbb{E} [\|\Delta_t\|^2] & = \mathbb{E} \left[ \left\| \frac{1}{m} \sum_{i \in \mathcal{S}_t} \Delta_{t-\tau_{t,i}}^i \right\|^2 \right] \\ & = \mathbb{E} \left[ \left\| \frac{1}{m} \sum_{i \in \mathcal{S}_t} \frac{\eta_l}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} g_{t-\tau_{t,i},k}^i \right\|^2 \right] \\ & \leq \mathbb{E} \left[ \left\| \frac{1}{m} \sum_{i \in \mathcal{S}_t} \frac{\eta_l}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \left\{ g_{t-\tau_{t,i},k}^i - \nabla f_i(x_{t-\tau_{t,i},k}^i) + \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\} \right\|^2 \right] \\ & \leq \mathbb{E} \left[ \left\| \frac{1}{m} \sum_{i \in \mathcal{S}_t} \frac{\eta_l}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \left\{ g_{t-\tau_{t,i},k}^i - \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\} \right\|^2 \right] + \mathbb{E} \left[ \left\| \frac{1}{m} \sum_{i \in \mathcal{S}_t} \frac{\eta_l}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\ & \leq \frac{1}{m^2} \sum_{i \in \mathcal{S}_t} \frac{\eta_l^2}{K_{t,i}^2} \sum_{k=0}^{K_{t,i}-1} \sigma_l^2 + \mathbb{E} \left[ \left\| \frac{1}{m} \sum_{i \in \mathcal{S}_t} \frac{\eta_l}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\ & \leq \frac{\eta_l^2}{m} \frac{1}{K_t} \sigma_l^2 + \frac{\eta_l^2}{m^2} \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1}_{\{i \in \mathcal{S}_t\}} \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \end{aligned}$$

where  $\frac{1}{K_t} = \sum_{i \in \mathcal{S}_t} \frac{1}{K_{t,i}}$ .

**Bounding**  $\sum_{t=0}^{T-1} \mathbb{E} [\|d_t\|^2]$ :

We could verify:

$$d_t = \sum_{p=0}^t a_{t,p} \Delta p, \quad \text{where } a_{t,p} = (1 - \beta_p) \prod_{q=p+1}^t \beta_q$$

We further get,

$$\begin{aligned}
& \mathbb{E} [\|d_t\|^2] = \mathbb{E} \left[ \left\| \sum_{p=0}^t a_{t,p} \Delta_p \right\|^2 \right] \\
& \leq \sum_{e=1}^d \mathbb{E} \left[ \left( \sum_{p=0}^t a_{t,p} \Delta_{p,e} \right)^2 \right] \leq \sum_{e=1}^d \mathbb{E} \left[ \left( \sum_{p=0}^t a_{t,p} \right) \left( \sum_{p=0}^t a_{t,p} \Delta_{p,e}^2 \right) \right] \leq \left( 1 - \prod_{q=0}^t \beta_q \right) \sum_{p=0}^t a_{t,p} \mathbb{E} [\|\Delta_p\|^2] \\
& \leq \left( 1 - \prod_{q=0}^t \beta_q \right) \sum_{p=0}^t a_{t,p} \left\{ \frac{\eta_l^2}{m} \frac{1}{K_t} \sigma_l^2 + \frac{\eta_l^2}{m^2} \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1} \{i \in \mathcal{S}_p\} \frac{1}{K_{p,i}} \sum_{k=0}^{K_{p,i}-1} \nabla f_i(x_{p-\tau_{p,i},k}^i) \right\|^2 \right] \right\} \\
& \leq \frac{\eta_l^2}{m} \frac{1}{K_t} \sigma_l^2 + \frac{\eta_l^2}{m^2} \sum_{p=0}^t a_{t,p} \cdot \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1} \{i \in \mathcal{S}_p\} \frac{1}{K_{p,i}} \sum_{k=0}^{K_{p,i}-1} \nabla f_i(x_{p-\tau_{p,i},k}^i) \right\|^2 \right]
\end{aligned}$$

Summing over  $t \in \{0, 1, \dots, T-1\}$ ,

$$\begin{aligned}
\sum_{t=0}^{T-1} \mathbb{E} [\|d_t\|^2] & \leq \frac{\eta_l^2}{m} \sum_{t=0}^{T-1} \frac{1}{K_t} \sigma_l^2 + \frac{\eta_l^2}{m^2} \sum_{t=0}^{T-1} \sum_{p=0}^t a_{t,p} \cdot \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1} \{i \in \mathcal{S}_p\} \frac{1}{K_{p,i}} \sum_{k=0}^{K_{p,i}-1} \nabla f_i(x_{p-\tau_{p,i},k}^i) \right\|^2 \right] \\
& \leq \frac{\eta_l^2}{m} \sum_{t=0}^{T-1} \frac{1}{K_t} \sigma_l^2 + \frac{\eta_l^2}{m^2} \sum_{p=0}^{T-1} \left( \sum_{t=p}^{T-1} a_{t,p} \right) \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1} \{i \in \mathcal{S}_p\} \frac{1}{K_{p,i}} \sum_{k=0}^{K_{p,i}-1} \nabla f_i(x_{p-\tau_{p,i},k}^i) \right\|^2 \right] \\
& \leq \frac{\eta_l^2}{m} \sum_{t=0}^{T-1} \frac{1}{K_t} \sigma_l^2 + \frac{\eta_l^2}{m^2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1} \{i \in \mathcal{S}_t\} \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right]
\end{aligned}$$

**Bounding  $\sum_{t=0}^{T-1} \mathbb{E} [\|y_t\|^2]$ :**

$$\begin{aligned}
\sum_{t=0}^{T-1} \mathbb{E} [\|y_t\|^2] & = \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{d_{t+1}}{\sqrt{\hat{v}_{k+1}} + \epsilon} \right\|^2 \right] \leq \frac{1}{4\epsilon^2} \sum_{t=0}^{T-1} \mathbb{E} [\|d_t\|^2] \\
& \leq \frac{\eta_l^2}{4\epsilon^2 m} \sum_{t=0}^{T-1} \frac{1}{K_t} \sigma_l^2 + \frac{\eta_l^2}{4\epsilon^2 m^2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1} \{i \in \mathcal{S}_t\} \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right]
\end{aligned}$$

Merging  $A_1, A_2, A_3$ , and  $A_4$  together, we get,

$$\begin{aligned}
& \mathbb{E} [f(z_{t+1})] \leq f(z_t) + A_1 + A_2 + A_3 + A_4 \\
& \leq f(z_t) + \frac{\eta^2 L}{2\epsilon^2} \left( \frac{\beta}{1-\beta} \right)^2 \mathbb{E} [\|d_t\|^2] + \frac{\eta^2 L}{2\epsilon^2} \mathbb{E} [\|\Delta_t\|^2] \\
& + \frac{\eta G}{4\epsilon^2} \mathbb{E} [\|\Delta_t\|^2] - \frac{\eta \eta_l}{2} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t} + \epsilon} \right\|^2 \right] - \frac{\eta \eta_l}{2(\sqrt{T}\eta_l G + \epsilon)} \mathbb{E} \left[ \left\| \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\
& + \frac{\eta^3 \eta_l L^2 \tau}{\epsilon} \sum_{k=t-\tau_{t,u}}^{t-1} \mathbb{E} [\|y_k\|^2] + \frac{5\eta \eta_l^3 L^2 \bar{K}_t}{\epsilon} \sigma_l^2 + \frac{30\eta \eta_l^3 L^2 \hat{K}_t^2}{\epsilon} \sigma_g^2 + \frac{30\eta \eta_l^3 L^2}{\epsilon} \frac{1}{n} \sum_{i=1}^n K_{t,i}^2 \mathbb{E} [\|\nabla f(x_{t-\tau_{t,i}})\|^2] \\
& + \left( \frac{\eta^2 \eta_l^2 G^2 L}{2\epsilon} \left( \frac{\beta}{1-\beta} \right)^2 + \eta \eta_l G^2 \frac{\beta}{1-\beta} \right) \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\|^2 \right] \\
& + \frac{L \eta^2}{4\epsilon^2} \mathbb{E} [\|\Delta_t\|^2] + L \eta^2 \eta_l^2 G^2 \left( \frac{\beta}{1-\beta} \right)^2 \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t} + \epsilon} - \frac{1}{\sqrt{\hat{v}_{t+1}} + \epsilon} \right\|^2 \right]
\end{aligned}$$

Sum over  $t \in \{0, 1, \dots, T-1\}$ ,

$$\begin{aligned}
& \mathbb{E}[f(z_T)] \leq f(z_0) + \\
& \frac{\eta^2 L}{2\epsilon^2} \left( \frac{\beta}{1-\beta} \right)^2 \sum_{t=0}^{T-1} \mathbb{E}[\|d_t\|^2] + \frac{\eta^2 L}{2\epsilon^2} \sum_{t=0}^{T-1} \mathbb{E}[\|\Delta_t\|^2] + \frac{\eta G}{4\epsilon^2} \sum_{t=0}^{T-1} \mathbb{E}[\|\Delta_t\|^2] - \frac{\eta\eta_l}{2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t + \epsilon}} \right\|^2 \right] \\
& - \frac{\eta\eta_l}{2(\sqrt{T}\eta_l G + \epsilon)} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] + \frac{\eta^3 \eta_l L^2 \tau}{\epsilon} \sum_{t=0}^{T-1} \sum_{k=t-\tau_{t,u}}^{t-1} \mathbb{E}[\|y_k\|^2] \\
& + \frac{5\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \hat{K}_t}{\epsilon} \sigma_l^2 + \frac{30\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \hat{K}_t^2}{\epsilon} \sigma_g^2 + \frac{30\eta\eta_l^3 L^2}{\epsilon} \sum_{t=0}^{T-1} \frac{1}{n} \sum_{i=1}^n K_{t,i}^2 \mathbb{E}[\|\nabla f(x_{t-\tau_{t,i}})\|^2] \\
& + \left( \frac{\eta^2 \eta_l^2 G^2 L}{2\epsilon} \left( \frac{\beta}{1-\beta} \right)^2 + \eta\eta_l G^2 \frac{\beta}{1-\beta} \right) \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t + \epsilon}} - \frac{1}{\sqrt{\hat{v}_{t+1} + \epsilon}} \right\| \right] \\
& + \frac{L\eta^2}{4\epsilon^2} \sum_{t=0}^{T-1} \mathbb{E}[\|\Delta_t\|^2] + L\eta^2 \eta_l^2 G^2 \left( \frac{\beta}{1-\beta} \right)^2 \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t + \epsilon}} - \frac{1}{\sqrt{\hat{v}_{t+1} + \epsilon}} \right\|^2 \right]
\end{aligned}$$

Plug in the bounds for  $\sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t + \epsilon}} - \frac{1}{\sqrt{\hat{v}_{t+1} + \epsilon}} \right\| \right]$  and  $\sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{\sqrt{\hat{v}_t + \epsilon}} - \frac{1}{\sqrt{\hat{v}_{t+1} + \epsilon}} \right\|^2 \right]$ , we have,

$$\begin{aligned}
& \mathbb{E}[f(z_T)] \leq f(z_0) + \\
& \frac{\eta^2 L}{2\epsilon^2} \left( \frac{\beta}{1-\beta} \right)^2 \sum_{t=0}^{T-1} \mathbb{E}[\|d_t\|^2] + \left( \frac{3L\eta^2 + \eta G}{4\epsilon^2} \right) \sum_{t=0}^{T-1} \mathbb{E}[\|\Delta_t\|^2] - \frac{\eta\eta_l}{2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t + \epsilon}} \right\|^2 \right] \\
& - \frac{\eta\eta_l}{2(\sqrt{T}\eta_l G + \epsilon)} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{1}{n} \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] + \frac{\eta^3 \eta_l L^2 \tau}{\epsilon} \sum_{t=0}^{T-1} \sum_{k=t-\tau_{t,u}}^{t-1} \mathbb{E}[\|y_k\|^2] \\
& + \frac{5\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \hat{K}_t}{\epsilon} \sigma_l^2 + \frac{30\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \hat{K}_t^2}{\epsilon} \sigma_g^2 + \frac{30\eta\eta_l^3 L^2}{\epsilon} \sum_{t=0}^{T-1} \frac{1}{n} \sum_{i=1}^n K_{t,i}^2 \mathbb{E}[\|\nabla f(x_{t-\tau_{t,i}})\|^2] \\
& + \left( \frac{\eta^2 \eta_l^2 G^2 L}{2\epsilon} \left( \frac{\beta}{1-\beta} \right)^2 + \eta\eta_l G^2 \frac{\beta}{1-\beta} \right) \frac{d}{2\epsilon} + L\eta^2 \eta_l^2 G^2 \left( \frac{\beta}{1-\beta} \right)^2 \frac{d}{4\epsilon^2}
\end{aligned}$$

We could further get,

$$\begin{aligned}
\mathbb{E}[f(z_T)] & \leq f(z_0) - \frac{\eta\eta_l}{2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t + \epsilon}} \right\|^2 \right] + \frac{30\eta\eta_l^3 L^2}{\epsilon} \sum_{t=0}^{T-1} \frac{1}{n} \sum_{i=1}^n K_{t,i}^2 \mathbb{E}[\|\nabla f(x_{t-\tau_{t,i}})\|^2] \\
& + \left( \left( \frac{\eta^2 L}{2\epsilon^2} \left( \frac{\beta}{1-\beta} \right)^2 + \frac{3L\eta^2 + \eta G}{4\epsilon^2} \right) \frac{\eta_l^2}{m^2} + \frac{\eta^3 \eta_l L^2 \tau^2}{\epsilon} \frac{\eta_l^2}{4\epsilon^2 m^2} \right). \\
& \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1}_{\{i \in \mathcal{S}_t\}} \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\
& - \frac{\eta\eta_l}{2(\sqrt{T}\eta_l G + \epsilon)n^2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] + \frac{30\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \hat{K}_t^2}{\epsilon} \sigma_g^2 \\
& + \left( \frac{\eta^2 \eta_l^2 G^2 L}{2\epsilon} \left( \frac{\beta}{1-\beta} \right)^2 + \eta\eta_l G^2 \frac{\beta}{1-\beta} \right) \frac{d}{2\epsilon} + L\eta^2 \eta_l^2 G^2 \left( \frac{\beta}{1-\beta} \right)^2 \frac{d}{4\epsilon^2} \\
& + \left( \frac{5\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \hat{K}_t}{\epsilon} + \frac{2\eta^2 \eta_l^2 L C_\beta^2 + 3L\eta^2 \eta_l^2 + \eta\eta_l^2 G}{4m\epsilon^2} \sum_{t=0}^{T-1} \frac{1}{K_t} + \frac{\eta^3 \eta_l^3 L^2 \tau^2}{4\epsilon^3 m} \sum_{t=0}^{T-1} \frac{1}{K_t} \right) \sigma_l^2
\end{aligned}$$

We have the following inequality,

$$\begin{aligned} \mathbb{E} \left[ \left\| \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] &= n \sum_{i=1}^n \mathbb{E} \left[ \left\| \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\ &\quad - \frac{1}{2} \sum_{i \neq j} \left\| \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) - \frac{1}{K_{t,j}} \sum_{k=0}^{K_{t,j}-1} \nabla f_j(x_{t-\tau_{t,j},k}^j) \right\|^2 \end{aligned}$$

We also know the following,

$$\begin{aligned} \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1}\{i \in \mathcal{S}_t\} \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] &= \sum_{i=1}^n \mathbb{P}\{i \in \mathcal{S}_t\} \left\| \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \\ &\quad + \sum_{i \neq j} \mathbb{P}\{i, j \in \mathcal{S}_t\} \left\langle \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i), \frac{1}{K_{t,j}} \sum_{k=0}^{K_{t,j}-1} \nabla f_j(x_{t-\tau_{t,j},k}^j) \right\rangle \\ &= \frac{m}{n} \sum_{i=1}^n \left\| \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 + \frac{m(m-1)}{n(n-1)} \sum_{i \neq j} \left\langle \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i), \frac{1}{K_{t,j}} \sum_{k=0}^{K_{t,j}-1} \nabla f_j(x_{t-\tau_{t,j},k}^j) \right\rangle \\ &= \frac{m^2}{n} \sum_{i=1}^n \left\| \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \\ &\quad - \frac{m(m-1)}{2n(n-1)} \sum_{i \neq j} \left\| \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) - \frac{1}{K_{t,j}} \sum_{k=0}^{K_{t,j}-1} \nabla f_j(x_{t-\tau_{t,j},k}^j) \right\|^2 \end{aligned}$$

Merging these two pieces together, and assume  $\eta_l \leq \frac{\epsilon}{\sqrt{T}G}$ , we have  $-\frac{\eta\eta_l}{2(\sqrt{T}\eta_l G + \epsilon)} \leq -\frac{\eta\eta_l}{4\epsilon}$ . And if  $H_1\eta_l^2 + H_2\eta_l \leq \epsilon^2$ , where  $H_1 \triangleq 2\eta^2 L^2 \tau^2$ ,  $H_2 \triangleq 4\eta L C_\beta^2 + 6\eta L\epsilon + 2G\epsilon$ ,  $C_\beta = \frac{\beta}{1-\beta}$ , we have the following inequality,

$$\left( \frac{\eta^2 L}{2\epsilon^2} \left( \frac{\beta}{1-\beta} \right)^2 + \frac{3L\eta^2 + \eta G}{4\epsilon^2} \right) \frac{\eta_l^2}{m^2} + \frac{\eta^3 \eta_l L^2 \tau^2}{\epsilon} \frac{\eta_l^2}{4\epsilon^2 m^2} \leq \frac{\eta\eta_l}{8\epsilon m^2}$$

thus, we have,

$$\begin{aligned} &\left( \left( \frac{\eta^2 L}{2\epsilon^2} \left( \frac{\beta}{1-\beta} \right)^2 + \frac{3L\eta^2 + \eta G}{4\epsilon^2} \right) \frac{\eta_l^2}{m^2} + \frac{\eta^3 \eta_l L^2 \tau^2}{\epsilon} \frac{\eta_l^2}{4\epsilon^2 m^2} \right). \\ &\sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1}\{i \in \mathcal{S}_t\} \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\ &\quad - \frac{\eta\eta_l}{2(\sqrt{T}\eta_l G + \epsilon)n^2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\ &\leq \frac{\eta\eta_l}{8\epsilon m^2} \cdot \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \sum_{i=1}^n \mathbb{1}\{i \in \mathcal{S}_t\} \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\ &\quad - \frac{\eta\eta_l}{4\epsilon n^2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \sum_{i=1}^n \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \right] \\ &= \left( -\frac{\eta\eta_l}{4\epsilon n} + \frac{\eta\eta_l}{8\epsilon n} \right) \cdot \mathcal{G}_1 + \left( \frac{\eta\eta_l}{8\epsilon n^2} - \frac{\eta\eta_l(m-1)}{16\epsilon m(n-1)n} \right) \cdot \mathcal{G}_2 \end{aligned}$$

where,

$$\begin{aligned}\mathcal{G}_1 &\triangleq \sum_{t=0}^{T-1} \sum_{i=1}^n \left\| \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) \right\|^2 \\ \mathcal{G}_2 &\triangleq \sum_{t=0}^{T-1} \sum_{i \neq j} \left\| \frac{1}{K_{t,i}} \sum_{k=0}^{K_{t,i}-1} \nabla f_i(x_{t-\tau_{t,i},k}^i) - \frac{1}{K_{t,j}} \sum_{k=0}^{K_{t,j}-1} \nabla f_j(x_{t-\tau_{t,j},k}^j) \right\|^2\end{aligned}$$

We can verify  $\mathcal{G}_2 \leq 2(n-1)\mathcal{G}_1$ , thus, we could verify the above term is non-positive.

Plugging back into the original terms,

$$\begin{aligned}\mathbb{E}[f(z_T)] &\leq f(z_0) - \frac{\eta\eta_l}{2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t + \epsilon}} \right\|^2 \right] + \frac{30\eta\eta_l^3 L^2}{\epsilon} \sum_{t=0}^{T-1} \frac{1}{n} \sum_{i=1}^n K_{t,i}^2 \mathbb{E}[\|\nabla f(x_{t-\tau_{t,i}})\|^2] \\ &\quad + \frac{30\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \hat{K}_t^2}{\epsilon} \sigma_g^2 \\ &\quad + \left( \frac{5\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \bar{K}_t}{\epsilon} + \frac{2\eta^2\eta_l^2 LC_\beta^2 + 3L\eta^2\eta_l^2 + \eta\eta_l^2 G}{4m\epsilon^2} \sum_{t=0}^{T-1} \frac{1}{K_t} + \frac{\eta^3\eta_l^3 L^2 \tau^2}{4\epsilon^3 m} \sum_{t=0}^{T-1} \frac{1}{K_t} \right) \sigma_l^2 \\ &\quad + \left( \frac{\eta^2\eta_l^2 G^2 L}{2\epsilon} \left( \frac{\beta}{1-\beta} \right)^2 + \eta\eta_l G^2 \frac{\beta}{1-\beta} \right) \frac{d}{2\epsilon} + L\eta^2\eta_l^2 G^2 \left( \frac{\beta}{1-\beta} \right)^2 \frac{d}{4\epsilon^2}\end{aligned}$$

Since  $\|\hat{v}_t\| \leq \eta_l^2 G^2 T$  and  $\eta_l \leq \frac{\epsilon}{\sqrt{T}G}$ , we have,

$$-\frac{\eta\eta_l}{2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t + \epsilon}} \right\|^2 \right] \leq -\frac{\eta\eta_l}{4\epsilon} \sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(x_t)\|^2]$$

Assume  $\eta_l \leq \frac{1}{\sqrt{120\epsilon\tau K_{\max}L}}$ , we have

$$\begin{aligned}&-\frac{\eta\eta_l}{2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t + \epsilon}} \right\|^2 \right] + \frac{30\eta\eta_l^3 L^2}{\epsilon} \sum_{t=0}^{T-1} \frac{1}{n} \sum_{i=1}^n K_{t,i}^2 \mathbb{E}[\|\nabla f(x_{t-\tau_{t,i}})\|^2] \\ &\leq -\frac{\eta\eta_l}{2} \sum_{t=0}^{T-1} \mathbb{E} \left[ \left\| \frac{\nabla f(x_t)}{\sqrt{\hat{v}_t + \epsilon}} \right\|^2 \right] + \frac{30\eta\eta_l^3 L^2}{\epsilon} K_{\max}^2 \tau \sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(x_t)\|^2] \\ &\leq -\frac{\eta\eta_l}{8\epsilon} \sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(x_t)\|^2]\end{aligned}$$

We have,

$$\begin{aligned}&\frac{\eta\eta_l}{8\epsilon} \sum_{t=0}^{T-1} \mathbb{E}[\|\nabla f(x_t)\|^2] \leq f(z_0) - \mathbb{E}[f(z_T)] \\ &\quad + \frac{30\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \hat{K}_t^2}{\epsilon} \sigma_g^2 \\ &\quad + \left( \frac{5\eta\eta_l^3 L^2 \sum_{t=0}^{T-1} \bar{K}_t}{\epsilon} + \frac{2\eta^2\eta_l^2 LC_\beta^2 + 3L\eta^2\eta_l^2 + \eta\eta_l^2 G}{4m\epsilon^2} \sum_{t=0}^{T-1} \frac{1}{K_t} + \frac{\eta^3\eta_l^3 L^2 \tau^2}{4\epsilon^3 m} \sum_{t=0}^{T-1} \frac{1}{K_t} \right) \sigma_l^2 \\ &\quad + \left( \frac{\eta^2\eta_l^2 G^2 L}{2\epsilon} \left( \frac{\beta}{1-\beta} \right)^2 + \eta\eta_l G^2 \frac{\beta}{1-\beta} \right) \frac{d}{2\epsilon} + L\eta^2\eta_l^2 G^2 \left( \frac{\beta}{1-\beta} \right)^2 \frac{d}{4\epsilon^2}\end{aligned}$$

Average with respect to  $T$ , we have,

$$\begin{aligned}
& \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla f(x_t)\|^2] \leq \frac{8\epsilon}{\eta\eta_l T} (f(z_0) - f^*) \\
& + 240\eta_l^2 L^2 \frac{1}{T} \sum_{t=0}^{T-1} \hat{K}_t^2 \sigma_g^2 + \frac{\Phi}{T} \\
& + \left( 40\eta_l^2 L^2 \frac{1}{T} \sum_{t=0}^{T-1} \bar{K}_t + \frac{4\eta\eta_l L C_\beta^2 + 6L\eta\eta_l + 2\eta_l G}{m\epsilon} \frac{1}{T} \sum_{t=0}^{T-1} \frac{1}{K_t} + \frac{2\eta^2\eta_l^2 L^2 \tau^2}{\epsilon^2 m} \frac{1}{T} \sum_{t=0}^{T-1} \frac{1}{K_t} \right) \sigma_l^2
\end{aligned}$$

We have the following,

$$\Phi \triangleq \left( \frac{\eta\eta_l G^2 L}{2\epsilon} \left( \frac{\beta}{1-\beta} \right)^2 + G^2 \frac{\beta}{1-\beta} \right) 4d + L\eta\eta_l G^2 \left( \frac{\beta}{1-\beta} \right)^2 \frac{2d}{\epsilon}$$

□

## B.2 Proof of Corollary 4.1.1

PROOF OF COROLLARY 4.1.1. By setting  $\eta_l = \Theta\left(\frac{1}{\sqrt{T}}\right)$ ,  $\eta = \Theta\left(\sqrt{mK}\right)$ , we have

$$\begin{aligned}
\Phi &= \Theta(1) + \Theta\left(\sqrt{\frac{mK}{T}}\right) \\
\Phi_l &= \Theta\left(\frac{K}{T}\right) + \Theta\left(\sqrt{\frac{1}{mKT}}\right) + \Theta\left(\frac{\tau^2}{T}\right), \quad \Phi_g = \Theta\left(\frac{K^2}{T}\right)
\end{aligned}$$

Plug in the complexity of  $\Phi$ ,  $\Phi_l$ , and  $\Phi_g$  back in, we could get,

$$\begin{aligned}
& \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla f(x_t)\|^2] \leq \Theta\left(\sqrt{\frac{1}{mKT}}\right) (f(z_0) - f^*) \\
& + \Theta\left(\frac{K^2}{T}\right) \sigma_g^2 + \Theta\left(\frac{1}{T}\right) + \Theta\left(\sqrt{\frac{mK}{T^3}}\right) + \left( \Theta\left(\frac{K}{T}\right) + \Theta\left(\sqrt{\frac{1}{mKT}}\right) + \Theta\left(\frac{\tau^2}{T}\right) \right) \sigma_l^2
\end{aligned}$$

Only keep the dominant terms, we have the convergence rate as,

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} [\|\nabla f(x_t)\|^2] = O\left(\sqrt{\frac{1}{mKT}}\right) + O\left(\frac{K^2}{T}\right) + O\left(\frac{\tau^2}{T}\right)$$

□

## C MORE EXPERIMENTAL RESULTS

We leave extra experimental results here. Basically, these are experiments with different concentration parameter  $\alpha$ , randomness  $R$ , maximum delay  $\tau$ . All experiments demonstrate the consistent superiority of our proposed approaches.

Figure 5: Training/Testing Curves in Figure 2(a), i.e. experimental results with  $\tau = 10$ .

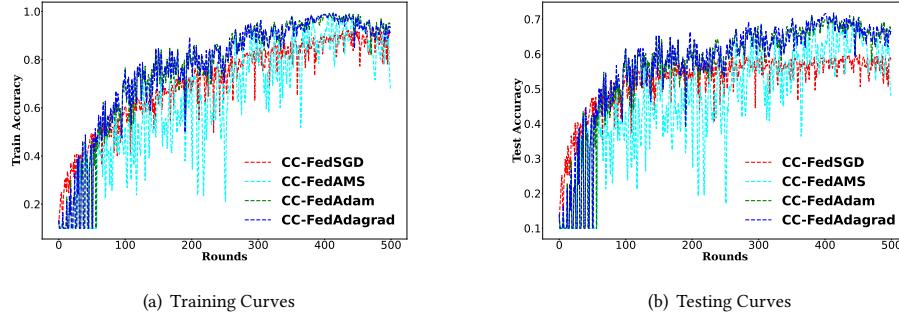


Figure 5: Training and testing curves for various CC-Federated Adaptive Optimizers (ResNet on CIFAR-10) with  $\tau = 10$ .

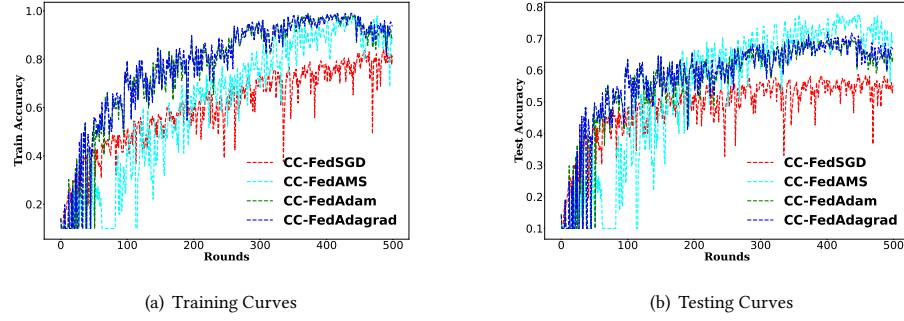
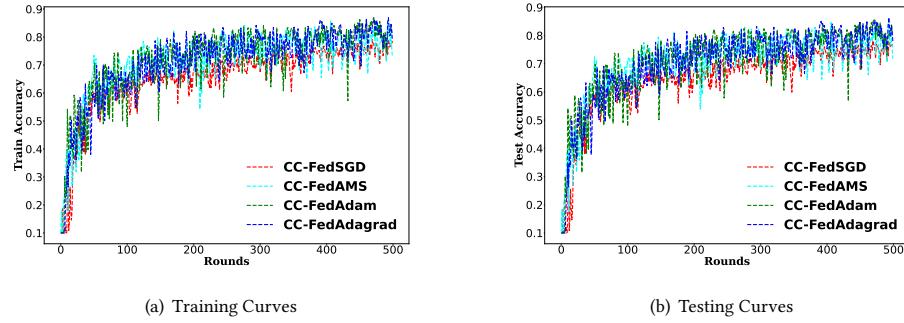
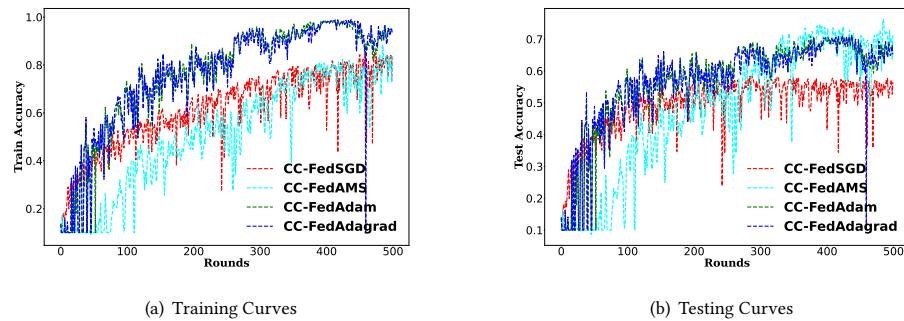
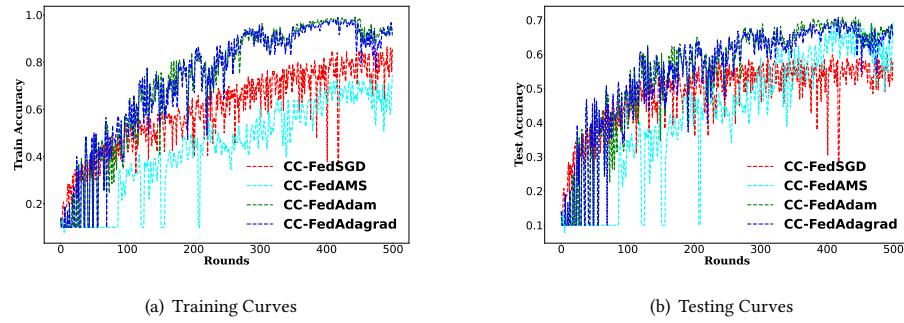
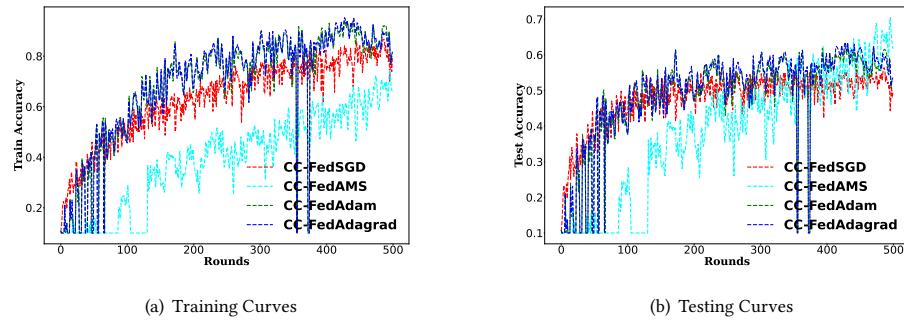
Figure 6: Training/Testing Curves in Figure 2(b), i.e. experimental results with  $R = 3$ .**Figure 6: Training and testing curves for various CC-Federated Adaptive Optimizers (ResNet on CIFAR-10) with  $R = 3$ .**

Figure 7: Training/Testing Curves in Figure 2(c), i.e. experimental results with shallow CNN on FMNIST. Note that the CNN [38] we use has the following structure, two 5x5 convolution layers followed by 2x2 max pooling (the first with 6 channels and the second with 16 channels) and two fully connected layers with ReLU activation (the first with 120 units and the second with 84 units).

**Figure 7: Training and testing curves for various CC-Federated Adaptive Optimizers (shallow CNN on FMNIST).**Figure 8, extra experimental results with  $R = 2$ .**Figure 8: Training and testing curves for various CC-Federated Adaptive Optimizers (ResNet on CIFAR-10) with  $R = 2$ .**Figure 9, extra experimental results with  $R = 1$ .Figure 10, extra experimental results with  $\alpha = 0.3$ .



**Figure 9: Training and testing curves for various CC-Federated Adaptive Optimizers (ResNet on CIFAR-10) with  $R = 1$ .**



**Figure 10: Training and testing curves for various CC-Federated Adaptive Optimizers (ResNet on CIFAR-10) with  $\alpha = 0.3$ .**

This figure "acm-jdslogo.png" is available in "png" format from:

<http://arxiv.org/ps/2501.09946v1>