

---

# FAVAS: Federated AVeraging with ASynchronous clients

---

Anonymous Author(s)

Affiliation

Address

email

## Abstract

In this paper, we propose a novel centralized Asynchronous Federated Learning (FL) framework, FAVAS for training Deep Neural Networks (DNNs) in resource-constrained environments. Despite its popularity, “classical” federated learning faces the increasingly difficult task of scaling synchronous communication over large wireless networks. Moreover, clients typically have different computing resources and therefore computing speed, which can lead to a significant bias (in favor of “fast” clients) when the updates are asynchronous. Therefore, practical deployment of FL requires to handle users with strongly varying computing speed in communication/resource constrained setting. We provide convergence guarantees for FAVAS in a smooth, non-convex environment and carefully compare the obtained convergence guarantees with existing bounds, when they are available. Experimental results show that the FAVAS algorithm outperforms current methods on standard benchmarks.

## 1 Introduction

Federated learning, a promising approach for training models from networked agents, involves the collaborative aggregation of locally computed updates, such as parameters, under centralized orchestration (Konečný et al., 2015; McMahan et al., 2017; Kairouz et al., 2021). The primary motivation behind this approach is to maintain privacy, as local data is never shared between agents and the central server (Zhao et al., 2018; Horváth et al., 2022). However, communication of training information between edge devices and the server is still necessary. The central server aggregates the local models to update the global model, which is then sent back to the devices. Federated learning helps alleviate privacy concerns, and it distributes the computational load among networked agents. However, each agent must have more computational power than is required for inference, leading to a computational power bottleneck. This bottleneck is especially important when federated learning is used in heterogeneous, cross-device applications.

Most approaches to centralized federated learning (FL) rely on synchronous operations, as assumed in many studies (McMahan et al., 2017; Wang et al., 2021). At each global iteration, a copy of the current model is sent from the central server to a selected subset of agents. The agents then update their model parameters using their private data and send the model updates back to the server. The server aggregates these updates to create a new shared model, and this process is repeated until the shared model meets a desired criterion. However, device heterogeneity and communication bottlenecks (such as latency and bandwidth) can cause delays, message loss, and stragglers, and the agents selected in each round must wait for the slowest one before starting the next round of computation. This waiting time can be significant, especially since nodes may have different computation speeds.

To address this challenge, researchers have proposed several approaches that enable asynchronous communication, resulting in improved scalability of distributed/federated learning (Xie et al., 2019;

Chen et al., 2020, 2021; Xu et al., 2021). In this case, the central server and local agents typically operate with inconsistent versions of the shared model, and synchronization in lockstep is not required, even between participants in the same round. As a result, the server can start aggregating client updates as soon as they are available, reducing training time and improving scalability in practice and theory.

**Contributions.** Our work takes a step toward answering this question by introducing FAVAS, a centralized federated learning algorithm designed to accommodate clients with varying computing resources and support asynchronous communication.

- In this paper, we introduce a new algorithm called FAVAS that uses an unbiased aggregation scheme for centralized federated learning with asynchronous communication. Our algorithm does not assume that clients computed the same number of epochs while being contacted, and we give non-asymptotic complexity bounds for FAVAS in the smooth nonconvex setting. We emphasize that the dependence of the bounds on the total number of agents  $n$  is improved compared to Zakerinia et al. (2022) and does not depend on a maximum delay.
- Experimental results show that our approach consistently outperforms other asynchronous baselines on the challenging TinyImageNet dataset (Le and Yang, 2015).

Our proposed algorithm FAVAS is designed to allow clients to perform their local steps independently of the server’s round structure, using a fully local, possibly outdated version of the model. Upon entering the computation, all clients are given a copy of the global model and perform at most  $K \geq 1$  optimization steps based on their local data. The server randomly selects a group of  $s$  clients in each server round, which, upon receiving the server’s request, submit an *unbiased* version of their progress. Although they may still be in the middle of the local optimization process, they send reweighted contributions so that fast and slow clients contribute equally. The central server then aggregates the models and sends selected clients a copy of the current model. The clients take this received server model as a new starting point for their next local iteration.

## 2 Related Works

Federated Averaging (FedAvg), also known as local SGD, is a widely used approach in federated learning. In this method, each client updates its local model using multiple steps of stochastic gradient descent (SGD) to optimize a local objective function. The local devices then submit their model updates to the central server for aggregation, and the server updates its own model parameters by averaging the client models before sending the updated server parameters to all clients. FedAvg has been shown to achieve high communication efficiency with infrequent synchronization, outperforming distributed large mini-batches SGD (Lin et al., 2019).

However, the use of multiple local epochs in FedAvg can cause each device to converge to the optima of its local objective rather than the global objective, a phenomenon known as client drift. This problem has been discussed in previous work; see (Karimireddy et al., 2020). Most of these studies have focused on synchronous federated learning methods, which have a similar update structure to FedAvg (Wang et al., 2020; Karimireddy et al., 2020; Qu et al., 2021; Makarenko et al., 2022; Mao et al., 2022; Tyurin and Richtárik, 2022). However, synchronous methods can be disadvantageous because they require all clients to wait when one or more clients suffer from high network delays or have more data, and require a longer training time. This results in idle time and wasted computing resources.

Moreover, as the number of nodes in a system increases, it becomes infeasible for the central server to perform synchronous rounds among all participants, and synchrony can degrade the performance of distributed learning. A simple approach to mitigate this problem is node sampling, e.g. Smith et al. (2017); Bonawitz et al. (2019), where the server only communicates with a subset of the nodes in a round. But if the number of stragglers is large, the overall training process still suffers from delays.

Synchronous FL methods are prone to stragglers. One important research direction is based on FedAsync (Xie et al., 2019) and subsequent works. The core idea is to update the global model immediately when the central server receives a local model. However, when staleness is important, performance is similar to FedAvg, so it is suboptimal in practice. ASO-Fed (Chen et al., 2020) proposes to overcome this problem and handles asynchronous FL with local streaming data by

introducing memory-terms on the local client side. AsyncFedED (Wang et al., 2022) also relies on the FedAsync instantaneous update strategy and also proposes to dynamically adjust the learning rate and the number of local epochs to staleness. Only one local updated model is involved in FedAsync-like global model aggregations. As a result, a larger number of training epochs are required and the frequency of communication between the server and the workers increases greatly, resulting in massive bandwidth consumption. From a different perspective, QuAFL (Zakerinia et al., 2022) develops a concurrent algorithm that is closer to the FedAvg strategy. QuAFL incorporates both asynchronous and compressed communication with convergence guarantees. Each client must compute  $K$  local steps and can be interrupted by the central server at any time. The client updates its model with the (compressed) central version and its current private model. The central server randomly selects  $s$  clients and updates the model with the (compressed) received local progress (since last contact) and the previous central model. QuAFL works with old variants of the model at each step, which slows convergence. However, when time, rather than the number of server rounds, is taken into account, QuAFL can provide a speedup because the asynchronous framework does not suffer from delays caused by stragglers. A concurrent and asynchronous approach aggregates local updates before updating the global model: FedBuff (Nguyen et al., 2022) addresses asynchrony using a buffer on the server side. Clients perform local iterations, and the base station updates the global model only after  $Z$  different clients have completed and sent their local updates. The gradients computed on the client side may be stale. The main assumption is that the client computations completed at each step come from a uniform distribution across all clients. Fedbuff is asynchronous, but is also sensitive to stragglers (must wait until  $Z$  different clients have done all local updates). Similarly, Koloskova et al. (2022) focus on Asynchronous SGD, and provide guarantees depending on some  $\tau_{max}$ . Similar to Nguyen et al. (2022) the algorithm is also impacted by stragglers, during the transitional regime at least. A recent work by Fraboni et al. (2023) extend the idea of Koloskova et al. (2022) by allowing multiple clients to contribute in one round. But this scheme also favors fast clients. Liu et al. (2021) does not run on buffers, but develops an Adaptive Asynchronous Federated Learning (AAFL) mechanism to deal with speed differences between local devices. Similar to FedBuff, in Liu et al. (2021)’s method, only a certain fraction of the locally updated models contribute to the global model update. Most convergence guarantees for asynchronous distributed methods depend on staleness or gradient delays (Nguyen et al., 2022; Toghiani and Uribe, 2022; Koloskova et al., 2022). Only Mishchenko et al. (2022) analyzes the asynchronous stochastic gradient descent (SGD) independently of the delays in the gradients. However, in the heterogeneous (non-IID) setting, convergence is proved up to an additive term that depends on the dissimilarity limit between the gradients of the local and global objective functions.

### 3 Algorithm

We consider optimization problems in which the components of the objective function (i.e., the data for machine learning problems) are distributed over  $n$  clients, i.e.,

$$\min_{w \in \mathbb{R}^d} R(w); \quad R(w) = \frac{1}{n} \sum_{i=1}^n \mathbb{E}_{(x,y) \sim p_{\text{data}}^i} [\ell(\text{NN}(x, w), y)], \quad (1)$$

where  $d$  is the number of parameters (network weights and biases),  $n$  is the total number of clients,  $\ell$  is the training loss (e.g., cross-entropy or quadratic loss),  $\text{NN}(x, w)$  is the DNN prediction function,  $p_{\text{data}}^i$  is the training distribution on client  $i$ . In FL, the distributions  $p_{\text{data}}^i$  are allowed to differ between clients (statistical heterogeneity).

Each client maintains three key values in its local memory: the local model  $w^i$ , a counter  $q^i$ , and the value of the initial model with which it started the iterations  $w_{\text{init}}^i$ . The counter  $q^i$  is incremented for each SGD step the client performs locally until it reaches  $K$ , at which point the client stops updating its local model and waits for the server request. Upon the request to the client  $i$ , the local model and counter  $q^i$  are reset. If a server request occurs before the  $K$  local steps are completed, the client simply pauses its current training process, reweights its gradient based on the number of local epochs (defined by  $E_{t+1}^i$ ), and sends its current *reweighted* model to the server.

In Zakerinia et al. (2022), we identified the client update  $w^i = \frac{1}{s+1}w_{t-1} + \frac{s}{s+1}w^i$  as a major shortcoming. When the number of sampled clients  $s$  is large enough,  $\frac{s}{s+1}w^i$  dominates the update and basically no server term are taken into consideration. This leads to a significant client drift. As a

---

**Algorithm 1:** FAVAS over  $T$  iterations. In **red** are highlighted the differences with QuAFL.

---

<p><b>Input</b> : Number of steps <math>T</math>, LR <math>\eta</math>, Selection Size <math>s</math>, Maximum local steps <math>K</math> ;</p> <p>/* At the Central Server</p> <p>1 <b>Initialize</b></p> <p>2   Initialize parameters <math>w_0</math>;</p> <p>3   Server sends <math>w_0</math> to all clients;</p> <p>4 <b>end</b></p> <p>5 <b>for</b> <math>t = 1, \dots, T</math> <b>do</b></p> <p>6   Generate set <math>\mathcal{S}_t</math> of <math>s</math> clients uniformly at random;</p> <p>7   <b>for all</b> clients <math>i \in \mathcal{S}_t</math> <b>do</b></p> <p>8     Server receives <math>w_{unbiased}^i</math> from client <math>i</math>;</p> <p>9   <b>end</b></p> <p>10   Update central server model  <math>w_t \leftarrow \frac{1}{s+1} w_{t-1} + (\frac{1}{s+1} \sum_{i \in \mathcal{S}_t} w_{unbiased}^i)</math>;</p> <p>11   <b>for all</b> clients <math>i \in \mathcal{S}_t</math> <b>do</b></p> <p>12     Server sends <math>w_t</math> to client <math>i</math>;</p> <p>13   <b>end</b></p> <p>14 <b>end</b></p>	<p>/* At Client <math>i</math> */</p> <p>15 <b>Initialize</b></p> <p>16   Client receives <math>w_0</math> and <math>K</math> from the Server;</p> <p>17   <b>Local variables</b> <math>w^i = w_0, q^i = 0</math>;</p> <p>18 <b>end</b></p> <p>19 <b>Loop</b></p> <p>20   Run ClientLocalTraining() concurrently;</p> <p>21 <b>When Contacted by the Server do</b></p> <p>22   Interrupt ClientLocalTraining();</p> <p>23   Define <math>\alpha^i</math> following (3) ;</p> <p>24   Send <math>w_{unbiased}^i := w_{init}^i + \frac{1}{\alpha^i} (w^i - w_{init}^i)</math> to the server;</p> <p>25   Receive <math>w_t</math> from the server;</p> <p>26   Update <math>w_{init}^i \leftarrow w_t, w^i \leftarrow w_t, q^i \leftarrow 0</math>;</p> <p>27   Restart ClientLocalTraining() from zero with updated variables;</p> <p>28 <b>end</b></p> <p>29 <b>end</b></p> <p>30 <b>function</b> ClientLocalTraining():</p> <p>31   <b>while</b> <math>q^i &lt; K</math> <b>do</b></p> <p>32     Compute local stochastic gradient <math>\tilde{g}^i</math> at <math>w^i</math>;</p> <p>33     Update local model <math>w^i \leftarrow w^i - \eta \tilde{g}^i</math>;</p> <p>34     Update local counter <math>q^i \leftarrow q^i + 1</math>;</p> <p>35   <b>end</b></p> <p>36   Wait();</p> <p>37 <b>end function</b></p>
---	---

---

consequence, QuAFL does not perform well in the heterogeneous case (see Section 5). Second, one can note that the updates in QuAFL are biased in favor of fast clients. Indeed each client computes gradients at its own pace and can reach different numbers of epochs while being contacted by the central server. It is assumed that clients compute the *same* number of local epochs in the analysis from Zakerinia et al. (2022), but it is not the case in practice. As a consequence, we propose FAVAS to deal with asynchronous updates without favoring fast clients. A first improvement is to update local weight directly with the received central model. Details can be found in Algorithm 1. Another idea to tackle gradient unbiasedness is to reweight the contributions from each of the  $s$  selected clients: these can be done either by dividing by the (proper) number of locally computed epochs, or by the expected value of locally computed epochs. In practice, we define the reweight  $\alpha^i = \mathbb{E}[E_{t+1}^i \wedge K]$ , or  $\alpha^i = \mathbf{P}(E_{t+1}^i > 0)(E_{t+1}^i \wedge K)$ , where  $\wedge$  stands for min. We assume that the server performs a number of training epochs  $T \geq 1$ . At each time step  $t \in \{1, \dots, T\}$ , the server has a model  $w_t$ . At initialization, the central server transmits identical parameters  $w_0$  to all devices. At each time step  $t$ , the central server selects a subset  $\mathcal{S}_t$  of  $s$  clients uniformly at random and requests their local models. Then, the requested clients submit their *reweighted* local models back to the server. When all requested models arrive at the server, the server model is updated based on a simple average (see Line 10). Finally, the server multicasts the updated server model to all clients in  $\mathcal{S}_t$ . In particular, all clients  $i \notin \mathcal{S}_t$  continue to run their individual processes without interruption.

**Remark 1.** In FAVAS's setting, we assume that each client  $i \in \{1, \dots, n\}$  keeps a full-precision local model  $w^i$ . In order to reduce the computational cost induced by the training process, FAVAS can also be implemented with a quantization function  $Q$ . First, each client computes backpropagation with respect to its quantized weights  $Q(w^i)$ . That is, the stochastic gradients are unbiased estimates of  $\nabla f_i(Q(w^i))$ . Moreover, the activations computed at forward propagation are quantized. Finally, the stochastic gradient obtained at backpropagation is quantized before the SGD update. In our supplementary experiments, we use the logarithmic unbiased quantization method of Chmiel et al. (2021).

Table 1: How long one has to wait to reach an  $\epsilon$  accuracy for non-convex functions. For simplicity, we ignore all constant terms. Each constant  $C_-$  depends on client speeds and represents the unit of time one has to wait in between two consecutive server steps.  $L$  is the Lipschitz constant, and  $F := (f(w_0) - f_*)$  is the initial conditions term.  $a_i, b$  are constants depending on client speeds statistics, and defined in Theorem 3.

Method	Units of time
FedAvg	$\left( \frac{FL\sigma^2 + (1 - \frac{a}{n})KG^2}{sK} \epsilon^{-2} + FL^{\frac{1}{2}}G\epsilon^{-\frac{3}{2}} + LFB^2\epsilon^{-1} \right) C_{FedAvg}$
FedBuff	$\left( FL(\sigma^2 + G^2)\epsilon^{-2} + FL((\frac{\tau_{max}}{s^2} + 1)(\sigma^2 + nG^2))^{\frac{1}{2}}\epsilon^{-\frac{3}{2}} + FL\epsilon^{-1} \right) C_{FedBuff}$
AsyncSGD	$\left( FL(3\sigma^2 + 4G^2)\epsilon^{-2} + FLG(s\tau_{avg})^{\frac{1}{2}}\epsilon^{-\frac{3}{2}} + (s\tau_{max}F)^{\frac{1}{2}}\epsilon^{-1} \right) C_{AsyncSGD}$
QuAFL	$\frac{1}{E^2}FLK(\sigma^2 + 2KG^2)\epsilon^{-2} + \frac{n\sqrt{n}}{E\sqrt{E}s}FKL(\sigma^2 + 2KG^2)^{\frac{1}{2}}\epsilon^{-\frac{3}{2}} + \frac{1}{E\sqrt{s}}n\sqrt{n}FBK^2L\epsilon^{-1}$
FAVAS	$FL(\sigma^2 \sum_i \frac{a_i}{n} + 8G^2b)\epsilon^{-2} + \frac{n}{s}FL^2(K^2\sigma^2 + L^2K^2G^2 + s^2\sigma^2 \sum_i \frac{a_i}{n} + s^2G^2b)^{\frac{1}{2}}\epsilon^{-\frac{3}{2}} + nFB^2KLb\epsilon^{-1}$

## 4 Analysis

In this section we provide complexity bounds for FAVAS in a smooth nonconvex environment. We introduce an abstraction to model the stochastic optimization process and prove convergence guarantees for FAVAS.

**Preliminaries.** We abstract the optimization process to simplify the analysis. In the proposed algorithm, each client asynchronously computes its own local updates without taking into account the server time step  $t$ . Here in the analysis, we introduce a different, but statistically equivalent setting. At the beginning of each server timestep  $t$ , each client maintains a local model  $w_{t-1}^i$ . We then assume that all  $n$  clients *instantaneously* compute local steps from SGD. The update in local step  $q$  for a client  $i$  is given by:

$$\tilde{h}_{t,q}^i = \tilde{g}^i \left( w_{t-1}^i - \sum_{s=1}^{q-1} \eta \tilde{h}_{t,s}^i \right), \quad (2)$$

where  $\tilde{g}^i$  represents the stochastic gradient that client  $i$  computes for the function  $f_i$ . We also define  $n$  independent random variables  $E_t^1, \dots, E_t^n$  in  $\mathbb{N}$ . Each random variable  $E_t^i$  models the number of local steps the client  $i$  could take before receiving the server request. We then introduce the following random variable:  $\tilde{h}_t^i = \sum_{q=1}^{E_t^i} \tilde{h}_{t,q}^i$ . Compared to Zakerinia et al. (2022), we do not assume that clients performed the same number of local epochs. Instead, we reweight the sum of the gradients by weights  $\alpha^i$ , which can be either *stochastic* or *deterministic*:

$$\alpha^i = \begin{cases} \mathbf{P}(E_{t+1}^i > 0)(E_{t+1}^i \wedge K) & \text{stochastic version,} \\ \mathbb{E}[E_{t+1}^i \wedge K] & \text{deterministic version.} \end{cases} \quad (3)$$

And we can define the *unbiased* gradient estimator:  $\check{h}_t^i = \frac{1}{\alpha^i} \sum_{q=1}^{E_t^i \wedge K} \tilde{h}_{t,q}^i$ .

Finally, a subset  $\mathcal{S}_t$  of  $s$  clients is chosen uniformly at random. This subset corresponds to the clients that send their models to the server at time step  $t$ . In the current notation, each client  $i \in \mathcal{S}_t$  sends the value  $w_{t-1}^i - \eta \check{h}_t^i$  to the server. We emphasise that in our abstraction, all clients compute  $E_t^i$  local updates. However, only the clients in  $\mathcal{S}_t$  send their updates to the server, and each client  $i \in \mathcal{S}_t$  sends only the  $K$  first updates. As a result, we introduce the following update equations:

$$\begin{cases} w_t = \frac{1}{s+1}w_{t-1} + \frac{1}{s+1} \sum_{i \in \mathcal{S}_t} (w_{t-1}^i - \eta \frac{1}{\alpha^i} \sum_{s=1}^{E_t^i \wedge K} \tilde{h}_{t,s}^i), \\ w_t^i = w_t, & \text{for } i \in \mathcal{S}_t, \\ w_t^i = w_{t-1}^i, & \text{for } i \notin \mathcal{S}_t. \end{cases}$$

### Assumptions and notations.

**A1. Uniform Lower Bound:** There exists  $f_* \in \mathbb{R}$  such that  $f(x) \geq f_*$  for all  $x \in \mathbb{R}^d$ .

**A2. Smooth Gradients:** For any client  $i$ , the gradient  $\nabla f_i(x)$  is  $L$ -Lipschitz continuous for some  $L > 0$ , i.e. for all  $x, y \in \mathbb{R}^d$ :  $\|\nabla f_i(x) - \nabla f_i(y)\| \leq L\|x - y\|$ .

192 **A3. Bounded Variance:** For any client  $i$ , the variance of the stochastic gradients is bounded by some  
 193  $\sigma^2 > 0$ , i.e. for all  $x \in \mathbb{R}^d$ :  $\mathbb{E}[\|\tilde{g}^i(x) - \nabla f_i(x)\|^2] \leq \sigma^2$ .

194 **A4. Bounded Gradient Dissimilarity:** There exist constants  $G^2 \geq 0$  and  $B^2 \geq 1$ , such that for all  
 195  $x \in \mathbb{R}^d$ :  $\sum_{i=1}^n \frac{\|\nabla f_i(x)\|^2}{n} \leq G^2 + B^2 \|\nabla f(x)\|^2$ .

196 We define the notations required for the analysis. Consider a time step  $t$ , a client  $i$ , and a local step  $q$ .  
 197 We define

$$\mu_t = \left( w_t + \sum_{i=1}^n w_t^i \right) / (n+1) \quad (4)$$

198 the average over all node models in the system at a given time  $t$ . The first step of the proof is to  
 199 compute a preliminary upper bound on the divergence between the local models and their average.  
 200 For this purpose, we introduce the Lyapunov function:  $\Phi_t = \|w_t - \mu_t\|^2 + \sum_{i=1}^n \|w_t^i - \mu_t\|^2$ .

201 **Upper bounding the expected change in potential.** A key result from our analysis is to upper  
 202 bound the change (in expectation) of the aforementioned potential function  $\Phi_t$ :

203 **Lemma 2.** For any time step  $t > 0$  we have:

$$\mathbb{E}[\Phi_{t+1}] \leq (1 - \kappa) \mathbb{E}[\Phi_t] + 3 \frac{s^2}{n} \eta^2 \sum_{i=1}^n \mathbb{E}[\|\check{h}_{t+1}^i\|^2], \quad \text{with } \kappa = \frac{1}{n} \left( \frac{s(n-s)}{2(n+1)(s+1)} \right).$$

204 The intuition behind Lemma 2 is that the potential function  $\Phi_t$  remains concentrated around its mean,  
 205 apart from deviations induced by the local gradient steps. The full analysis involves many steps and  
 206 we refer the reader to Appendix B for complete proofs. In particular, Lemmas 16 and 18 allow us  
 207 to examine the scalar product between the expected node progress  $\sum_{i=1}^n \check{h}_t^i$  and the true gradient  
 208 evaluated on the mean model  $\nabla f(\mu_t)$ . The next theorem allows us to compute an upper-bound  
 209 on the averaged norm-squared of the gradient, a standard quantity studied in nonconvex stochastic  
 210 optimization.

211 **Convergence results.** The following statement shows that FAVAS algorithm converges towards a  
 212 first-order stationary point, as  $T$  the number of global epochs grows.

213 **Theorem 3.** Assume A1 to A4 and assume that the learning rate  $\eta$  satisfies  $\eta \leq \frac{1}{20B^2bKLs}$ . Then  
 214 FAVAS converges at rate:

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 \leq \frac{2(n+1)F}{Ts\eta} + \frac{Ls}{n+1} \left( \frac{\sigma^2}{n} \sum_{i=1}^n a^i + 8G^2b \right) \eta + L^2 s^2 \left( \frac{720\sigma^2}{n} \sum_{i=1}^n a^i + 5600bG^2 \right) \eta^2,$$

215 with  $F := (f(\mu_0) - f_*)$ , and

$$\begin{cases} a^i, b = \frac{1}{\mathbb{P}(E_{t+1}^i > 0)^2} \left( \frac{\mathbb{P}(E_{t+1}^i > 0)}{K^2} + \mathbb{E} \left[ \frac{\mathbb{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \right), \max_i \left( \frac{1}{\mathbb{P}(E_{t+1}^i > 0)} \right) \text{ for } \alpha^i = \mathbb{P}(E_{t+1}^i > 0)(E_{t+1}^i \wedge K), \\ a^i, b = \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]}, \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \text{ for } \alpha^i = \mathbb{E}[E_{t+1}^i \wedge K]. \end{cases}$$

216 Note that the previous convergence result refers to the average model  $\mu_t$ . In practice, this does not  
 217 pose much of a problem. After training is complete, the server can ask each client to submit its final  
 218 model. It should be noted that each client communicates  $\frac{sT}{n}$  times with the server during training.  
 219 Therefore, an additional round of data exchange represents only a small increase in the total amount  
 220 of data transmitted.

221 The bound in Theorem 3 contains 3 terms. The first term is standard for a general non-convex target  
 222 and expresses how initialization affects convergence. The second and third terms depend on the  
 223 statistical heterogeneity of the client distributions and the fluctuation of the minibatch gradients.  
 224 Table 1 compares complexity bounds along with synchronous and asynchronous methods. One can  
 225 note the importance of the ratio  $\frac{s}{n}$ . Compared to Nguyen et al. (2022) or Koloskova et al. (2022),  
 226 FAVAS can potentially suffer from delayed updates when  $\frac{s}{n} \ll 1$ , but FAVAS does *not* favor fast  
 227 clients at all. In practice, it is not a major shortcoming, and FAVAS is more robust to fast/slow clients  
 228 distribution than FedBuff/AsyncSGD (see Figure 2). We emphasize both FedBuff and AsyncSGD rely  
 229 on strong assumptions: neither the queuing process, nor the transitional regime are taken into account

in their analysis. In practice, during the first iterations, only fast clients contribute. It induces a serious bias. Our experiments indicate that a huge amount of server iterations has to be accomplished to reach the stationary regime. Still, under this regime, slow clients are contributing with delayed information. Nguyen et al. (2022); Koloskova et al. (2022) propose to uniformly bound this delay by some quantity  $\tau_{max}$ . We keep this notation while reporting complexity bounds in Table 1, but argue nothing guarantee  $\tau_{max}$  is properly defined (i.e. finite). All analyses except that of Mishchenko et al. (2022) show that the number of updates required to achieve accuracy grows linearly with  $\tau_{max}$ , which can be very adverse. Specifically, suppose we have two parallel workers - a fast machine that takes only 1 unit of time to compute a stochastic gradient, and a slow machine that takes 1000 units of time. If we use these two machines to implement FedBuff/AsyncSGD, the gradient delay of the slow machine will be one thousand, because in the 1 unit of time we wait for the slow machine, the fast machine will produce one thousand updates. As a result, the analysis based on  $\tau_{max}$  deteriorates by a factor of 1000.

In the literature, guarantees are most often expressed as a function of server steps. In the asynchronous case, this is *inappropriate* because a single step can take very different amounts of time depending on the method. For example, with FedAvg or Scaffold (Karimireddy et al., 2020), one must wait for the slowest client for each individual server step. Therefore, we introduce in Table 1 constants  $C$  that depend on the client speed and represent the unit of time to wait between two consecutive server steps. Finally, optimizing the value of the learning rate  $\eta$  with Lemma 12 yields the following:

**Corollary 4.** Assume A1 to A4. We can optimize the learning rate by Lemma 12 and FAVAS reaches an  $\epsilon$  precision for a number of server steps  $T$  greater than (up to numerical constants):

$$\frac{FL(\frac{\sigma^2}{n} \sum_i^n a^i + 8G^2b)}{\epsilon^2} + (n+1) \left( \frac{FL^2(K^2\sigma^2 + L^2K^2G^2 + \frac{s^2\sigma^2}{n} \sum_i^n a^i + s^2G^2b)^{\frac{1}{2}}}{s\epsilon^{\frac{3}{2}}} + \frac{FB^2K Lb}{\epsilon} \right),$$

where  $F = (f(\mu_0) - f_*)$ , and  $(a^i, b)$  are defined in Theorem 3.

The second term in Corollary 4 is better than the one from the QuAFL analysis ( $n^3$  of Zakerinia et al., 2022). Although this  $(n+1)$  term can be suboptimal, note that it is only present at second order from  $\epsilon$  and therefore becomes negligible when  $\epsilon$  goes to 0 (Lu and De Sa, 2020; Zakerinia et al., 2022).

**Remark 5.** Our analysis can be extended to the case of quantized neural networks. The derived complexity bounds also hold for the case when the quantization function  $Q$  is biased. We make only a weak assumption about  $Q$  (we assume that there is a constant  $r_d$  such that for any  $x \in \mathbb{R}^d$   $\|Q(x) - x\|^2 \leq r_d$ ), which holds for standard quantization methods such as stochastic rounding and deterministic rounding. The only effect of quantization would be increased variance in the stochastic gradients. We need to add to the upper bound given in Theorem 3 an "error floor" of  $12L^2r_d$ , which remains independent of the number of server epochs. For stochastic or deterministic rounding,  $r_d = \Theta(d\frac{1}{2^b})$ , where  $b$  is the number of bits used. The error bound is the cost of using quantization as part of the optimization algorithm. Previous works with quantized models also include error bounds (Li et al., 2017; Li and Sa, 2019).

## 5 Numerical Results

We test FAVAS on three image classification tasks: MNIST (Deng, 2012), CIFAR-10 (Krizhevsky et al., 2009), and TinyImageNet (Le and Yang, 2015). For the MNIST and CIFAR-10 datasets, two training sets are considered: an IID and a non-IID split. In the first case, the training images are randomly distributed among the  $n$  clients. In the second case, each client takes two classes (out of the ten possible) without replacement. This process leads to heterogeneity among the clients.

The standard evaluation measure for FL is the number of server rounds of communication to achieve target accuracy. However, the time spent between two consecutive server steps can be very different for asynchronous and synchronous methods. Therefore, we compare different synchronous and asynchronous methods w.r.t. *total simulation time* (see below). We also measured the loss and accuracy of the model in terms of server steps and total local client steps (see Appendix C.3). In all experiments, we track the performance of each algorithm by evaluating the server model against an unseen validation dataset. We present the test accuracy and variance, defined as  $\sum_{i=1}^n \|w_t^i - w_t\|^2$ .



We decide to focus on non-uniform timing experiments as in Nguyen et al. (2022), and we base our simulation environment on QuAFL’s code<sup>1</sup>. After simulating  $n$  clients, we randomly group them into fast or slow nodes. We assume that at each time step  $t$  (for the central server), a set of  $s$  clients is randomly selected without replacement. We assume that the clients have different computational speeds, and refer to Appendix C.2 for more details. We assume that only one-third of the clients are slow, unless otherwise noted. We compare FAVAS with the classic synchronous approach FedAvg (McMahan et al., 2017) and two newer asynchronous methods QuAFL (Zakerinia et al., 2022) and FedBuff (Nguyen et al., 2022). Details on implementing other methods can be found in Appendix C.1.

We use the standard data augmentations and normalizations for all methods. FAVAS is implemented in Pytorch, and experiments are performed on an NVIDIA Tesla-P100 GPU. Standard multiclass cross entropy loss is used for all experiments. All models are fine-tuned with  $n = 100$  clients,  $K = 20$  local epochs, and a batch of size 128. Following the guidelines of Nguyen et al. (2022), the buffer size in FedBuff is set to  $Z = 10$ . In FedAvg, the total simulated time depends on the maximum number of local steps  $K$  and the slowest client runtime, so it is proportional to the number of local steps and the number of global steps. In QuAFL and FAVAS on the other hand, each global step has a predefined duration that depends on the central server clock. Therefore, the global steps have similar durations and the total simulated time is the sum of the durations of the global steps. In FedBuff, a global step requires filling a buffer of size  $Z$ . Consequently, both the duration of a global step and the total simulated time depend on  $Z$  and on the proportion of slow clients (see Appendix C.2 for a detailed discussion).

We first report the accuracy of a shallow neural network trained on MNIST. The learning rate is set to 0.5 and the total simulated time is set to 5000. We also compare the accuracy of a Resnet20 (He et al., 2016) with the CIFAR-10 dataset (Krizhevsky et al., 2009), which consists of 50000 training images and 10000 test images (in 10 classes). For CIFAR-10, the learning rate is set to 0.005 and the total simulation time is set to 10000. In Figure 1, we show the test accuracy of FAVAS and competing

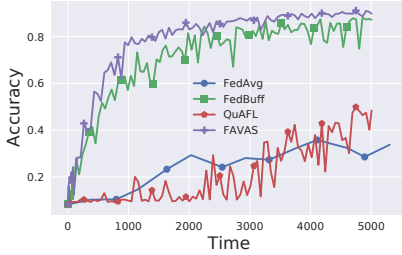


Figure 1: Test accuracy on the MNIST dataset with a non-IID split in between  $n = 100$  total nodes,  $s = 20$ .

Table 2: Final accuracy on the test set (average and standard deviation over 10 random experiments) for the MNIST classification task. The last two columns correspond to Figures 1 and 2.

Methods	IID split	non-IID split ( $\frac{2}{3}$ fast clients)	non-IID split ( $\frac{1}{9}$ fast clients)
FedAvg	$93.4 \pm 0.3$	$38.7 \pm 7.7$	$44.8 \pm 6.9$
QuAFL	$92.3 \pm 0.9$	$40.7 \pm 6.7$	$45.5 \pm 4.0$
FedBuff	<b><math>96.0 \pm 0.1</math></b>	$85.1 \pm 3.2$	$67.3 \pm 5.5$
FAVAS	$95.1 \pm 0.1$	<b><math>88.9 \pm 0.9</math></b>	<b><math>87.3 \pm 2.3</math></b>

methods on the MNIST dataset. We find that FAVAS and other asynchronous methods can offer a significant advantage over FedAvg when time is taken into account. However, QuAFL does not appear to be adapted to the non-IID environment. We identified client-side updating as a major shortcoming. While this is not severe when each client optimizes (almost) the same function, the QuAFL mechanism suffers from significant client drift when there is greater heterogeneity between clients. FedBuff is efficient when the number of stragglers is negligible compared to  $n$ . However, FedBuff is sensitive to the fraction of slow clients and may get stuck if the majority of clients are classified as slow and a few are classified as fast. In fact, fast clients will mainly feed the buffer, so the central updates will be heavily biased towards fast clients, and little information from slow clients will be considered. Figure 2 illustrates this phenomenon, where one-ninth of the clients are classified as fast. To provide a fair comparison, Table 2 gives the average performance of 10 random experiments with the different methods on the test set.

In Figure 3a, we report accuracy on a non-IID split of the CIFAR-10 dataset. FedBuff and FAVAS both perform better than other approaches, but FedBuff suffers from greater variance. We explain this limitation by the bias FedBuff provides in favor of fast clients. We also tested FAVAS on the TinyImageNet dataset (Le and Yang, 2015) with a ResNet18. TinyImageNet has 200 classes and each

<sup>1</sup><https://github.com/ShayanTaleai/QuAFL>



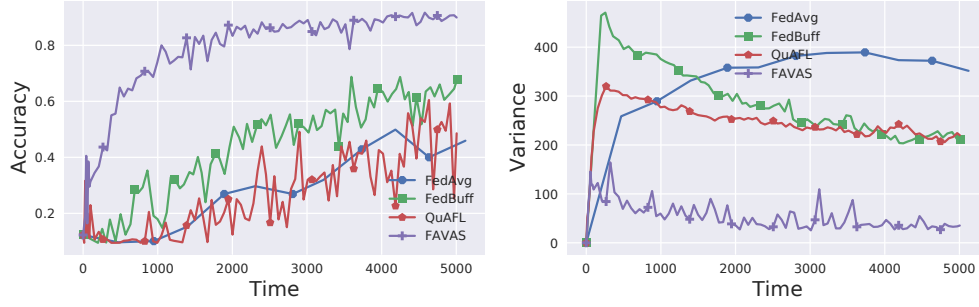


Figure 2: Test accuracy and variance on the MNIST dataset with a non-IID split between  $n = 100$  total nodes. In this particular experiment, one-ninth of the clients are defined as fast.

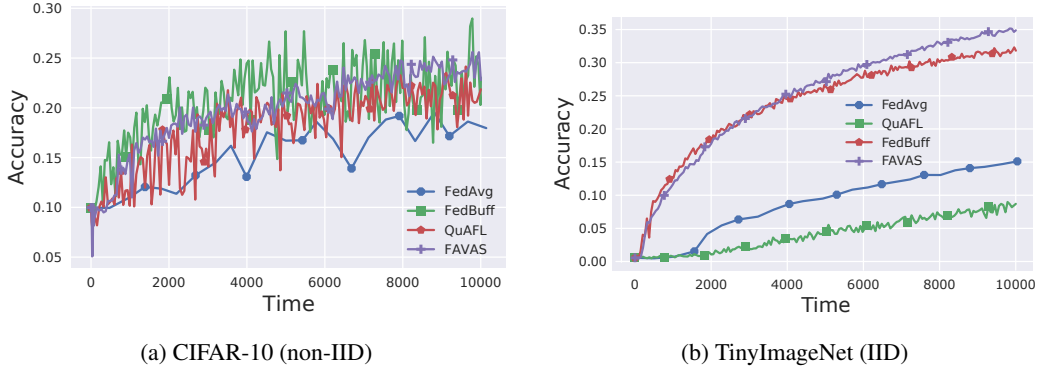


Figure 3: Test accuracy on CIFAR-10 and TinyImageNet datasets with  $n = 100$  total nodes. Central server selects  $s = 20$  clients at each round.

320 class has 500 (RGB) training images, 50 validation images and 50 test images. To train ResNet18, we  
 321 follow the usual practices for training NNs: we resize the input images to  $64 \times 64$  and then randomly  
 322 flip them horizontally during training. During testing, we center-crop them to the appropriate size.  
 323 The learning rate is set to 0.1 and the total simulated time is set to 10000. Figure 3b illustrates  
 324 the performance of FAVAS in this experimental setup. While the partitioning of the training dataset  
 325 follows an IID strategy, TinyImageNet provides enough diversity to challenge federated learning  
 326 algorithms. Figure 3b shows that FAVAS scales much better on large image classification tasks than  
 327 any of the methods we considered.

328 **Remark 6.** We also evaluated the performance of FAVAS with and without quantization. We ran the  
 329 code<sup>2</sup> from LUQ (Chmiel et al., 2021) and adapted it to our datasets and the FL framework. Even  
 330 when the weights and activation functions are highly quantized, the results are close to their full  
 331 precision counterpart (see Figure 7 in Appendix C).

## 332 6 Conclusion

333 We have presented FAVAS the first (centralised) Federated Learning method of federated averaging  
 334 that accounts for asynchrony in resource-constrained environments. We established complexity  
 335 bounds under verifiable assumptions with explicit dependence on all relevant constants. Empirical  
 336 evaluation shows that FAVAS is more efficient than synchronous and asynchronous state-of-the-art  
 337 mechanisms in standard CNN training benchmarks for image classification.

<sup>2</sup><https://openreview.net/forum?id=clwYez4n8e8>

## References

- Bonawitz, K., Eichner, H., Grieskamp, W., Huba, D., Ingerman, A., Ivanov, V., Kiddon, C., Konečný, J., Mazzocchi, S., McMahan, B., et al. (2019). Towards federated learning at scale: System design. *Proceedings of Machine Learning and Systems*, 1:374–388.
- Chen, Y., Ning, Y., Slawski, M., and Rangwala, H. (2020). Asynchronous online federated learning for edge devices with non-iid data. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 15–24. IEEE.
- Chen, Z., Liao, W., Hua, K., Lu, C., and Yu, W. (2021). Towards asynchronous federated learning for heterogeneous edge-powered internet of things. *Digital Communications and Networks*, 7(3):317–326.
- Chmiel, B., Banner, R., Hoffer, E., Yaacov, H. B., and Soudry, D. (2021). Logarithmic unbiased quantization: Simple 4-bit training in deep learning. *arXiv preprint arXiv:2112.10769*.
- Deng, L. (2012). The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE signal processing magazine*, 29(6):141–142.
- Fraboni, Y., Vidal, R., Kameni, L., and Lorenzi, M. (2023). A general theory for federated optimization with asynchronous and heterogeneous clients updates. *Journal of Machine Learning Research*, 24(110):1–43.
- He, K., Zhang, X., Ren, S., and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778.
- Horváth, S., Sanjabi, M., Xiao, L., Richtárik, P., and Rabbat, M. (2022). Fedshuffle: Recipes for better use of local work in federated learning. *arXiv preprint arXiv:2204.13169*.
- Kairouz, P., McMahan, H. B., Avent, B., Bellet, A., Bennis, M., Bhagoji, A. N., Bonawitz, K., Charles, Z., Cormode, G., Cummings, R., et al. (2021). Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210.
- Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., and Suresh, A. T. (2020). Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pages 5132–5143. PMLR.
- Koloskova, A., Stich, S. U., and Jaggi, M. (2022). Sharper convergence guarantees for asynchronous sgd for distributed and federated learning. *arXiv preprint arXiv:2206.08307*.
- Konečný, J., McMahan, B., and Ramage, D. (2015). Federated optimization: Distributed optimization beyond the datacenter. *arXiv preprint arXiv:1511.03575*.
- Krizhevsky, A., Hinton, G., et al. (2009). Learning multiple layers of features from tiny images.
- Lannelongue, L., Grealey, J., and Inouye, M. (2021). Green algorithms: Quantifying the carbon footprint of computation. *Advanced Science*, 8(12):2100707.
- Le, Y. and Yang, X. (2015). Tiny imagenet visual recognition challenge. *CS 231N*, 7(7):3.
- Li, H., De, S., Xu, Z., Studer, C., Samet, H., and Goldstein, T. (2017). Training quantized nets: A deeper understanding. *Advances in Neural Information Processing Systems*, 30.
- Li, Z. and Sa, C. D. (2019). Dimension-free bounds for low-precision training.
- Lin, T., Stich, S. U., Patel, K. K., and Jaggi, M. (2019). Don’t use large mini-batches, use local sgd. In *International Conference on Learning Representations*.
- Liu, J., Xu, H., Wang, L., Xu, Y., Qian, C., Huang, J., and Huang, H. (2021). Adaptive asynchronous federated learning in resource-constrained edge computing. *IEEE Transactions on Mobile Computing*.
- Lu, Y. and De Sa, C. (2020). Moniqua: Modulo quantized communication in decentralized sgd. In *International Conference on Machine Learning*, pages 6415–6425. PMLR.

383 Makarenko, M., Gasanov, E., Islamov, R., Sadiev, A., and Richtarik, P. (2022). Adaptive compression  
384 for communication-efficient distributed training. *arXiv preprint arXiv:2211.00188*.

385 Mao, Y., Zhao, Z., Yan, G., Liu, Y., Lan, T., Song, L., and Ding, W. (2022). Communication-efficient  
386 federated learning with adaptive quantization. *ACM Transactions on Intelligent Systems and  
387 Technology (TIST)*, 13(4):1–26.

388 McMahan, B., Moore, E., Ramage, D., Hampson, S., and y Arcas, B. A. (2017). Communication-  
389 efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*,  
390 pages 1273–1282. PMLR.

391 Mishchenko, K., Bach, F., Even, M., and Woodworth, B. (2022). Asynchronous sgd beats minibatch  
392 sgd under arbitrary delays. *arXiv preprint arXiv:2206.07638*.

393 Nguyen, J., Malik, K., Zhan, H., Yousefpour, A., Rabbat, M., Malek, M., and Huba, D. (2022).  
394 Federated learning with buffered asynchronous aggregation. In *International Conference on  
395 Artificial Intelligence and Statistics*, pages 3581–3607. PMLR.

396 Qu, L., Song, S., and Tsui, C.-Y. (2021). Feddq: Communication-efficient federated learning with  
397 descending quantization. *arXiv preprint arXiv:2110.02291*.

398 Smith, V., Chiang, C.-K., Sanjabi, M., and Talwalkar, A. S. (2017). Federated multi-task learning.  
399 *Advances in neural information processing systems*, 30.

400 Toghiani, M. T. and Uribe, C. A. (2022). Unbounded gradients in federated leaning with buffered  
401 asynchronous aggregation. *arXiv preprint arXiv:2210.01161*.

402 Tyurin, A. and Richtárik, P. (2022). Dasha: Distributed nonconvex optimization with communi-  
403 cation compression, optimal oracle complexity, and no client synchronization. *arXiv preprint  
404 arXiv:2202.01268*.

405 Wang, J., Charles, Z., Xu, Z., Joshi, G., McMahan, H. B., Al-Shedivat, M., Andrew, G., Avestimehr,  
406 S., Daly, K., Data, D., et al. (2021). A field guide to federated optimization. *arXiv preprint  
407 arXiv:2107.06917*.

408 Wang, J., Liu, Q., Liang, H., Joshi, G., and Poor, H. V. (2020). Tackling the objective inconsistency  
409 problem in heterogeneous federated optimization. *Advances in neural information processing  
410 systems*, 33:7611–7623.

411 Wang, Q., Yang, Q., He, S., Shui, Z., and Chen, J. (2022). Asyncfeded: Asynchronous federated learn-  
412 ing with euclidean distance based adaptive weight aggregation. *arXiv preprint arXiv:2205.13797*.

413 Xie, C., Koyejo, S., and Gupta, I. (2019). Asynchronous federated optimization. *arXiv preprint  
414 arXiv:1903.03934*.

415 Xu, C., Qu, Y., Xiang, Y., and Gao, L. (2021). Asynchronous federated learning on heterogeneous  
416 devices: A survey. *arXiv preprint arXiv:2109.04269*.

417 Zakerinia, H., Talaei, S., Nadiradze, G., and Alistarh, D. (2022). Quaf: Federated averaging can be  
418 both asynchronous and communication-efficient. *arXiv preprint arXiv:2206.10032*.

419 Zhao, Y., Li, M., Lai, L., Suda, N., Civin, D., and Chandra, V. (2018). Federated learning with non-iid  
420 data. *arXiv preprint arXiv:1806.00582*.

## A Environmental footprint

In the current context, we estimated the carbon footprint of our experiments to be about 10 kg CO<sub>2</sub>e (calculated using green-algorithms.org v2.1 Lannelongue et al. (2021)). This shed light on the crucial need to develop energy friendly NNs.

## B Proofs of Section 4

The complete analysis of Theorem 3 is fully provided in the following pages, and is heavily inspired by Zakerinia et al. (2022)'s analysis. We ask the reader to refer to Section 4 for a detailed definition of the random variables used in the analysis.

### B.1 Preliminaries

Let  $(\Omega, \mathcal{F}, \mathbb{P})$  be a probability space. We assume that all the random variables defined in this proof are defined on this space. Consider  $I = \mathbb{N}^* \times \mathbb{N}$ , together with the lexicographical ordering in  $I$ . We recall that  $(a, b) \leq (c, d)$  in lexicographical ordering if and only if  $a < c$  or  $a = c$  and  $b \leq d$ . We define two families of  $\sigma$ -algebras  $(\mathcal{F}_{(t,q)})_{(t,q) \in I}$  and  $(\mathcal{F}_t)_{t \in \mathbb{N}}$ . These are defined by the relations  $\mathcal{F}_0 = \{\emptyset, \Omega\}$  and, for  $t \geq 1$  and  $q \geq 0$ ,

$$\begin{aligned} \mathcal{F}_{(t,q)} &= \sigma \left( \mathcal{F}_{t-1} \cup \sigma \left( \tilde{h}_{t,q'}^i : q' \leq q, i \in [1, n] \right) \right) \\ \mathcal{F}_t &= \sigma \left( \sigma \left( \mathcal{S}_t, E_t^i : i \in [1, n] \right) \cup \bigcup_{q \geq 0} \mathcal{F}_{(t,q)} \right). \end{aligned} \quad (5)$$

Therefore,  $\mathcal{F}_t$  contains all information up to the end of time step  $t$ . Additionally,  $\mathcal{F}_{(t,q)}$  contains all information up to the local step  $q$  of time step  $t$ . Notice that at this point we do not have information about  $\mathcal{S}_t$  and  $E_t^i$ . We define  $\mathbb{E}_t$  to be the conditional expectation with respect to  $\mathcal{F}_t$ .

For all time steps  $t$ , local steps  $q$ , and client  $i$ , we define:

$$h_{t,q}^i = \nabla f_i \left( w_t^i - \eta \sum_{s=1}^{q-1} \tilde{h}_{t,s}^i \right) = \mathbb{E} \left[ \tilde{h}_{t,q}^i | \mathcal{F}_{(t,q-1)} \right]$$

where  $\tilde{h}_{t,s}^i$  is defined in (2).

Contrary to Zakerinia et al. (2022), we do *not* assume that all clients have computed the same number of epochs upon being contacted by the server.

We start by establishing a basic, yet useful, algebraic equality in the following Lemma.

**Lemma 7.** *Let  $s \in \mathbb{N}^*$ ,  $a_i, b \in \mathbb{R}^d$  for  $i \in [1, s]$  be vectors. It holds that:*

$$\left\| \frac{\sum_{i=1}^s a_i + b}{s+1} \right\|^2 - \frac{1}{s+1} \sum_{i=1}^s \|a_i\|^2 - \frac{1}{s+1} \|b\|^2 = \frac{-1}{(s+1)^2} \sum_{i=1}^s \|a_i - b\|^2 - \frac{1}{(s+1)^2} \sum_{i,j=1}^s \|a_i - a_j\|^2$$

*Proof.*

$$\begin{aligned} I &= \left\| \frac{\sum_{i=1}^s a_i + b}{s+1} \right\|^2 - \frac{1}{s+1} \sum_{i=1}^s \|a_i\|^2 - \frac{1}{s+1} \|b\|^2 \\ &= (s+1)^{-2} \left( \left\langle \sum_{i=1}^s a_i + b, \sum_{i=1}^s a_i + b \right\rangle - (s+1) \sum_{i=1}^s \|a_i\|^2 - (s+1) \|b\|^2 \right). \end{aligned}$$

444 We expand the inner product to obtain:

$$\begin{aligned}
I &= (s+1)^{-2} \left( -s \|b\|^2 + 2 \sum_{i=1}^s \langle a_i, b \rangle + \sum_{i,j=1}^s \langle a_i, a_j \rangle - (s+1) \sum_{i=1}^s \|a_i\|^2 \right) \\
&= \frac{-1}{(s+1)^2} \left( \left( \sum_{i=1}^s \|a_i\|^2 + s \|b\|^2 - 2 \sum_{i=1}^s \langle a_i, b \rangle \right) + \left( s \sum_{i=1}^s \|a_i\|^2 - \sum_{i,j=1}^s \langle a_i, a_j \rangle \right) \right) \\
&= \frac{-1}{(s+1)^2} \left( \left( \sum_{i=1}^s \|a_i - b\|^2 \right) + \left( \sum_{i,j=1}^s \|a_i - a_j\|^2 \right) \right).
\end{aligned}$$

445

□

446 **Lemma 8.** Let  $n, d$  be positive integers,  $a_1, a_2, \dots, a_n, b \in \mathbb{R}^d$  be vectors, and  $g = \frac{a_1 + \dots + a_n}{n}$  be the  
447 center of mass of  $a_1, \dots, a_n$ . Then the following identity holds:

$$\sum_{i=1}^n \|b - a_i\|^2 = n \|b - g\|^2 + \sum_{i=1}^n \|g - a_i\|^2.$$

448 *Proof.* We may compute:

$$\sum_{i=1}^n \|b - a_i\|^2 = \sum_{i=1}^n \|b - g + g - a_i\|^2 = \sum_{i=1}^n (\|b - g\|^2 + \|g - a_i\|^2 + 2 \langle b - g, g - a_i \rangle)$$

449 We can easily see that  $\sum_{i=1}^n \langle b - g, g - a_i \rangle = 0$ . The identity then follows. □

450 **Lemma 9.** Let  $X_1, \dots, X_n$  be random variables. Moreover, let  $\mathcal{S}$  be a subset of  $\{1, 2, \dots, n\}$   
451 containing  $s$  elements chosen uniformly at random. Assume that  $\mathcal{S}$  is independent from  $X_i$  for  
452  $i = 1, \dots, n$ . Then, we have:

$$\mathbb{E} \left[ \sum_{i \in \mathcal{S}} X_i \right] = \sum_{i=1}^n \frac{s}{n} \mathbb{E}[X_i].$$

453 *Proof.* We introduce indicator functions in the sum above and apply linearity of expectation:

$$\mathbb{E} \left[ \sum_{i \in \mathcal{S}} X_i \right] = \mathbb{E} \left[ \sum_{i=1}^n X_i \mathbb{1}_{\mathcal{S}}(i) \right] = \sum_{i=1}^n \mathbb{E}[X_i \mathbb{1}_{\mathcal{S}}(i)].$$

454 Using that  $\mathcal{S}$  is independent of each  $X_i$  we get:

$$\mathbb{E} \left[ \sum_{i \in \mathcal{S}} X_i \right] = \sum_{i=1}^n \mathbb{E}[X_i] \mathbb{E}[\mathbb{1}_{\mathcal{S}}(i)] = \sum_{i=1}^n \frac{s}{n} \mathbb{E}[X_i].$$

455

□

456 The following preliminary lemmas allow one to recover unbiased gradient estimates in FAVAS, and  
457 bound their variance.

458 **Lemma 10.** Let  $\{Y^q\}_{q>0}$  a collection of independent random variables such that  $\mathbb{E}[Y_q] = \mu$ . Let  $S$   
459 be a positive random variable independent from the collection  $\{Y^q\}_{q>0}$ , and with expected value  
460  $\mathbb{E}[S] = m$ . Consider  $M_1 = \mathbb{E}[\frac{\mathbb{1}_{[S>0]}}{S \mathbb{P}(S>0)} \sum_{q=1}^S Y^q]$ , and  $M_2 = \mathbb{E}[\frac{\mathbb{1}_{[S>0]}}{\mathbb{E}[S]} \sum_{q=1}^S Y^q]$ .  $M_1, M_2$  are  
461 unbiased estimate of  $\mu$ , i.e.  $M = \mu$ .

462 *Proof.*  $M_1$  : One can note this setting corresponds to  $\alpha^i = \mathbf{P}(E_t^i > 0)(E_t^i \wedge K)$ . We have

$$\begin{aligned} M_1 &= \frac{1}{\mathbf{P}(S > 0)} \mathbb{E}[\mathbb{E}[\frac{\mathbf{1}[S > 0]}{S} \sum_{q=1}^S Y^q | S]] \\ &= \frac{1}{\mathbf{P}(S > 0)} \mathbb{E}[S \frac{\mathbf{1}[S > 0]}{S} \mu] \\ &= \frac{1}{\mathbf{P}(S > 0)} \mathbb{E}[\mathbf{1}[S > 0] \mu] \\ &= \mu. \end{aligned}$$

463 Thus when reweighting with the (random) number of additions, one need to also take into account  
464 the  $\mathbf{P}(s > 0)$  term to obtain an unbiased estimate.

465  $M_2$  : This setting corresponds to  $\alpha^i = \mathbb{E}[E_t^i \wedge K]$ . We have

$$\begin{aligned} M_2 &= \mathbb{E}[\mathbb{E}[\frac{\mathbf{1}[S > 0]}{\mathbb{E}[S]} \sum_{q=1}^S Y^q | S]] \\ &= \mathbb{E}[S \frac{\mathbf{1}[S > 0]}{\mathbb{E}[S]} \mu] \\ &= \frac{\mathbb{E}[S \mathbf{1}[S > 0]]}{\mathbb{E}[S]} \mu \\ &= \mu. \end{aligned}$$

466 Thus reweighting the sum with  $\mathbb{E}[s]$  allows us to obtain an unbiased estimate  $M_2 = \mu$ .  $\square$

467 **Lemma 11.** Let  $\{Y^q\}_{q>0}$  a collection of independent random variables such that  $\mathbb{E}[Y_q] = \mu$ ,  
468  $\text{Var}(Y^q) = \text{Var}(Y) < \infty$ . Let  $S$  be a positive random variable independent from the collection  
469  $\{Y^q\}_{q>0}$ , and with expected value  $\mathbb{E}[S] = m$ . Consider  $M_1 = \mathbb{E}[\frac{\mathbf{1}[S>0]}{S\mathbf{P}(S>0)} \sum_{q=1}^S Y^q]$ , and  $M_2 =$   
470  $\mathbb{E}[\frac{\mathbf{1}[S>0]}{\mathbb{E}[S]} \sum_{q=1}^S Y^q]$ . We compute the variance :

$$\begin{cases} \text{Var}(M_1) = \frac{1}{\mathbf{P}(s>0)^2} (\mu^2 \mathbf{P}(S > 0)(1 - \mathbf{P}(S > 0)) + \text{Var}(Y) \mathbb{E}[\frac{\mathbf{1}[S>0]}{S}]) \\ \text{Var}(M_2) = \frac{\mu^2 \text{Var}(S)}{\mathbb{E}[S]^2} + \frac{\text{Var}(Y)}{\mathbb{E}[S]}. \end{cases}$$

471

472 *Proof.*  $M_2$ : This setting corresponds to  $\alpha^i = \mathbb{E}[E_t^i \wedge K]$ . We have

$$\begin{aligned} \text{Var}(M_2) &= \frac{1}{\mathbb{E}[S]^2} \mathbb{E}[(\sum_{q=1}^S Y^q - \mu m)^2] \\ &= \frac{1}{\mathbb{E}[S]^2} \mathbb{E}[(\sum_{q=1}^S (Y^q - \mu) + \mu(S - m))^2] \end{aligned}$$

473 The cross products reduce to 0 in expectation, hence

$$\begin{aligned} \text{Var}(M_2) &= \frac{1}{\mathbb{E}[S]^2} \mathbb{E}[\mathbb{E}[(S(Y^q - \mu) + \mu^2(S - m)^2)]] \\ &= \frac{1}{\mathbb{E}[S]^2} (\mathbb{E}[S] \text{Var}(Y) + \mu^2 \text{Var}(S)) \end{aligned}$$

474  $M_1$ : This setting corresponds to  $\alpha^i = \mathbf{P}(E_t^i > 0)(E_t^i \wedge K)$  or QuAFL (Zakerinia et al., 2022) when  
 475 the same number of local epochs is done by clients. First note that  $\mathbb{E}[M_1|S] = \mu \frac{\mathbf{1}(S>0)}{\mathbf{P}(S>0)}$ . We have

$$\begin{aligned} \text{Var}(M_1) &= \mathbb{E}[\mathbb{E}[(M_1 - \mathbb{E}[M_1|S])^2|S]] + \mathbb{E}[(\mathbb{E}[M_1|s] - \mathbb{E}[M_1])^2] \\ &= \frac{1}{\mathbf{P}(S > 0)^2} \mathbb{E}[(\frac{1}{S} \sum_q (Y^q - \mu) \mathbf{1}[S > 0])^2] + \mathbb{E}[(\mu \mathbf{1}(S > 0) - \mu \mathbf{P}(S > 0))^2] \\ &= \frac{1}{\mathbf{P}(S > 0)^2} \mathbb{E}[\frac{1}{S^2} (S \text{Var}(Y) \mathbf{1}[S > 0])^2] + \frac{1}{\mathbf{P}(S > 0)^2} \mu^2 \mathbb{E}[(\mathbf{1}(S > 0) - \mathbf{P}(S > 0))^2] \\ &= \frac{1}{\mathbf{P}(S > 0)^2} \text{Var}(Y) \mathbb{E}[\frac{\mathbf{1}[S > 0]}{S}] + \frac{1}{\mathbf{P}(S > 0)^2} \mu^2 \mathbb{E}[(\mathbf{1}(S > 0) - \mathbf{P}(S > 0))^2] \\ &= \frac{1}{\mathbf{P}(S > 0)^2} \text{Var}(Y) \mathbb{E}[\frac{\mathbf{1}[S > 0]}{S}] + \frac{1}{\mathbf{P}(S > 0)^2} \mu^2 \mathbf{P}(S > 0)(1 - \mathbf{P}(S > 0)). \end{aligned}$$

476 □

477 Last, but not least, we will make use of a result from Koloskova et al. (2022) to optimize learning  
 478 rates and obtain sharp complexity bounds:

479 **Lemma 12.** Assume A1 to A4 and consider problem (1). If the output of an optimization algorithm  
 480 with step size  $\eta$  has an expected error upper bounded by  $\frac{r_0}{\eta(T+1)} + b\eta + e\eta^2$ , and if  $\eta$  satisfies the  
 481 constraints  $\eta \leq \min((\frac{r_0}{b(T+1)})^{\frac{1}{2}}, (\frac{r_0}{e(T+1)})^{\frac{1}{3}}, \frac{1}{d})$  for some non-negative values  $r_0, b, d, e$ , then the  
 482 number of communication rounds required to reach  $\epsilon$  accuracy is lower bounded by

$$\frac{36br_0}{\epsilon^2} + \frac{15r_0\sqrt{e}}{\epsilon^{\frac{3}{2}}} + \frac{3dr_0}{\epsilon}.$$

483 *Proof.* Consider  $\psi_T$  a positive variable such that

$$\psi_T \leq \frac{r_0}{\eta(T+1)} + b\eta + e\eta^2$$

484 for any positive step size verifying  $\eta \leq \min((\frac{r_0}{b(T+1)})^{\frac{1}{2}}, (\frac{r_0}{e(T+1)})^{\frac{1}{3}}, \frac{1}{d})$ . Then Koloskova et al. (2022)  
 485 shows the following inequality:

$$\psi_T \leq 2(\frac{br_0}{T+1})^{\frac{1}{2}} + e^{\frac{1}{3}}(\frac{r_0}{T+1})^{\frac{2}{3}} + \frac{dr_0}{T+1}.$$

486 In order to reach an  $\epsilon$  precision, we bound each term by  $\frac{\epsilon}{3}$ , and deduce the following lower bound

$$T \geq \frac{36br_0}{\epsilon^2} + \frac{15r_0\sqrt{e}}{\epsilon^{\frac{3}{2}}} + \frac{3dr_0}{\epsilon}.$$

487 □

## 488 B.2 Useful Lemmas

489 A key result of our analysis is the upper bound on the change (in expected value) of the potential  
 490 function  $\Phi_t$ . Recall that  $\Phi_t$  is defined by equation:

$$\Phi_t = \|w_t - \mu_t\|^2 + \sum_{i=1}^n \|w_t^i - \mu_t\|^2.$$

491 In the next lemma, we show that  $\Phi_t$  exhibits a contractive property, which allows us to bound its  
 492 value through the execution of the optimization algorithm.



493 **B.2.1 Proof of Lemma 2**

494 *Proof.* Consider the following quantities:

$$\begin{aligned} \mu_t &= \left( w_t + \sum_{i=1}^n w_t^i \right) / (n+1) \\ G_{t+1} &= -\frac{1}{n+1} \eta \sum_{i \in \mathcal{S}_{t+1}} \check{h}_{t+1}^i \end{aligned}$$

496 where  $\check{h}_{t+1}^i = \frac{1}{\mathbf{P}(E_{t+1}^i > 0)(E_{t+1}^i \wedge K)} \tilde{h}_{t+1}^i$  or  $\check{h}_{t+1}^i = \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i$ . And recall the updates rules:

$$\begin{cases} w_{t+1} &= \frac{1}{s+1} \left( w_t + \sum_{i \in \mathcal{S}_{t+1}} w_t^i \right) + \frac{n+1}{s+1} G_{t+1}, \\ w_t^i &= w_t; \text{ for } i \in \mathcal{S}_t \\ w_t^i &= w_{t-1}^i; \text{ for } i \notin \mathcal{S}_t. \end{cases}$$

497 With these definitions, we get

$$\mu_{t+1} = \frac{s+1}{n+1} w_{t+1} + \frac{1}{n+1} \sum_{i \notin \mathcal{S}_{t+1}} w_t^i = \mu_t + G_{t+1}.$$

498 We can now compute the difference of potential:

$$\begin{aligned} \Phi_{t+1} - \Phi_t &= \sum_{i \in \mathcal{S}_{t+1}} \left( \|w_{t+1} - \mu_{t+1}\|^2 - \|w_t^i - \mu_t\|^2 \right) + \|w_{t+1} - \mu_{t+1}\|^2 - \|w_t - \mu_t\|^2 \\ &\quad + \sum_{i \notin \mathcal{S}_{t+1}} \left( \|w_t^i - \mu_{t+1}\|^2 - \|w_t^i - \mu_t\|^2 \right). \end{aligned}$$

499 We can rewrite this equation into a more convenient form:

$$\Phi_{t+1} - \Phi_t = (s+1) \|w_{t+1} - \mu_{t+1}\|^2 - \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - \mu_t\|^2 - \|w_t - \mu_t\|^2 + \sum_{i \notin \mathcal{S}_{t+1}} \left( \|w_t^i - \mu_t - G_{t+1}\|^2 - \|w_t^i - \mu_t\|^2 \right). \quad (6)$$

500 **Step 1.** First, notice that:

$$\begin{aligned} \sum_{i \notin \mathcal{S}_{t+1}} \left( \|w_t^i - \mu_t - G_{t+1}\|^2 - \|w_t^i - \mu_t\|^2 \right) &= \sum_{i \notin \mathcal{S}_{t+1}} (\|G_{t+1}\|^2 - 2\langle w_t^i - \mu_t, G_{t+1} \rangle) \\ &= (n-s) \|G_{t+1}\|^2 - 2 \left\langle \sum_{i \notin \mathcal{S}_{t+1}} (w_t^i - \mu_t), G_{t+1} \right\rangle. \quad (7) \end{aligned}$$

501 **Step 2.** Next, we compute the first term of Equation (6):

$$\begin{aligned} (s+1) \|w_{t+1} - \mu_{t+1}\|^2 &= (s+1) \left\| \frac{(w_t - \mu_t) + \sum_{i \in \mathcal{S}_{t+1}} (w_t^i - \mu_t)}{s+1} + \frac{n+1}{s+1} G_{t+1} - G_{t+1} \right\|^2 \\ &= (s+1) \left\| \frac{(w_t - \mu_t) + \sum_{i \in \mathcal{S}_{t+1}} (w_t^i - \mu_t)}{s+1} \right\|^2 + 2 \left\langle (w_t - \mu_t) + \sum_{i \in \mathcal{S}_{t+1}} (w_t^i - \mu_t), \frac{n+1}{s+1} G_{t+1} - G_{t+1} \right\rangle \\ &\quad + \frac{(n-s)^2}{s+1} \|G_{t+1}\|^2. \end{aligned}$$

502 We apply Young's inequality ( $\langle x, y \rangle \leq \beta \|x\|^2 + 1/(4\beta) \|y\|^2$ ) and Jensen inequality to get:

$$\begin{aligned} &\left\langle (w_t - \mu_t) + \sum_{i \in \mathcal{S}_{t+1}} (w_t^i - \mu_t), \frac{n+1}{s+1} G_{t+1} \right\rangle \\ &\leq \frac{\alpha(n+1)}{s+1} \|w_t - \mu_t\|^2 + \frac{\alpha(n+1)}{s+1} \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - \mu_t\|^2 + \frac{n+1}{4\alpha} \|G_{t+1}\|^2. \end{aligned}$$

503 Applying Lemma 7, we get:

$$\begin{aligned} & (s+1) \left\| \frac{(w_t - \mu_t) + \sum_{i \in \mathcal{S}_{t+1}} (w_t^i - \mu_t)}{s+1} \right\|^2 \\ &= \left( \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - \mu_t\|^2 \right) + \|w_t - \mu_t\|^2 - \frac{1}{s+1} \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - w_t\|^2 - \frac{1}{s+1} \sum_{i,j \in \mathcal{S}_{t+1}} \|w_t^i - w_t^j\|^2 \end{aligned}$$

504 Combining the results above, we get:

$$\begin{aligned} (s+1) \|w_{t+1} - \mu_{t+1}\|^2 &\leq \|w_t - \mu_t\|^2 + \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - \mu_t\|^2 - \frac{1}{s+1} \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - w_t\|^2 \\ &\quad + \frac{2\alpha(n+1)}{s+1} \|w_t - \mu_t\|^2 + \frac{2\alpha(n+1)}{s+1} \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - \mu_t\|^2 + \left( \frac{(n-s)^2}{s+1} + \frac{n+1}{2\alpha} \right) \|G_{t+1}\|^2 \\ &\quad - 2 \left\langle (w_t - \mu_t) + \sum_{i \in \mathcal{S}_{t+1}} (w_t^i - \mu_t), G_{t+1} \right\rangle. \end{aligned}$$

505 Using simple algebra, this relation can be rewritten as :

$$\begin{aligned} (s+1) \|w_{t+1} - \mu_{t+1}\|^2 &\leq \left( 1 + \frac{2\alpha(n+1)}{s+1} \right) \|w_t - \mu_t\|^2 \\ &\quad + \left( 1 + \frac{2\alpha(n+1)}{s+1} \right) \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - \mu_t\|^2 + \left( \frac{n+1}{2\alpha} + \frac{(n-s)^2}{s+1} \right) \|G_{t+1}\|^2 \\ &\quad - 2 \left\langle (w_t - \mu_t) + \sum_{i \in \mathcal{S}_{t+1}} (w_t^i - \mu_t), G_{t+1} \right\rangle - \frac{1}{s+1} \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - w_t\|^2. \end{aligned} \tag{8}$$

506 **Step 3.** We combine the results above to get bound  $\Phi_{t+1} - \Phi_t$ . Plugging (7) into (6), (8), and using  
507 (see (4))

$$(w_t - \mu_t) + \sum_{i \in \mathcal{S}_{t+1}} (w_t^i - \mu_t) + \sum_{i \notin \mathcal{S}_{t+1}} (w_t^i - \mu_t) = 0,$$

508 we get that

$$\begin{aligned} \Phi_{t+1} - \Phi_t &\leq \frac{2\alpha(n+1)}{s+1} \|w_t - \mu_t\|^2 + \frac{2\alpha(n+1)}{s+1} \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - \mu_t\|^2 + \left( (n-s) + \frac{n+1}{2\alpha} + \frac{(n-s)^2}{s+1} \right) \|G_{t+1}\|^2 \\ &\quad - \frac{1}{s+1} \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - w_t\|^2. \end{aligned}$$

509 **Step 4.** We now apply Lemma 9 to take expectations in the inequality above. We have:

$$\mathbb{E} \left[ \sum_{i \in \mathcal{S}_{t+1}} \|w_t^i - \mu_t\|^2 \right] = \sum_{i=1}^n \frac{s}{n} \mathbb{E}[\|w_t^i - \mu_t\|^2].$$

510 Moreover, we have:

$$\|G_{t+1}\|^2 \leq \frac{s}{(n+1)^2} \eta^2 \sum_{i \in \mathcal{S}_{t+1}} \|\tilde{h}_{t+1}^i\|^2.$$

511 Therefore, also by Lemma 9, we have:

$$\mathbb{E}[\|G_{t+1}\|^2] \leq \frac{s^2}{n(n+1)^2} \eta^2 \sum_{i=1}^n \mathbb{E}[\|\tilde{h}_{t+1}^i\|^2].$$

512 **Step 5.** Now we derive the final inequality. We have:

$$\begin{aligned}\mathbb{E}[\Phi_{t+1}] - \mathbb{E}[\Phi_t] &\leq \frac{2\alpha(n+1)}{s+1} \mathbb{E} \|w_t - \mu_t\|^2 + \frac{2\alpha(n+1)s}{(s+1)n} \sum_{i=1}^n \mathbb{E} \|w_t^i - \mu_t\|^2 + \\ &\quad + \left( (n-s) + \frac{n+1}{2\alpha} + \frac{(n-s)^2}{s+1} \right) \frac{s^2}{n(n+1)^2} \eta^2 \sum_{i=1}^n \mathbb{E} [\|\tilde{h}_{t+1}^i\|^2] \\ &\quad - \frac{s}{n(s+1)} \sum_{i=1}^n \mathbb{E} \|w_t^i - w_t\|^2.\end{aligned}$$

513 We can apply Lemma 8 to the above inequality's last line with  $a_i = w_t^i$  for  $i = 1, \dots, n$ , and  
514  $a_{n+1} = w_t$ , and  $b = w_t$ :

$$\sum_{i=1}^n \mathbb{E} \|w_t^i - w_t\|^2 = (n+2) \mathbb{E} \|w_t - \mu_t\|^2 + \sum_{i=1}^n \mathbb{E} \|w_t^i - \mu_t\|^2.$$

515 This allows us to simplify:

$$\begin{aligned}\mathbb{E}[\Phi_{t+1}] - \mathbb{E}[\Phi_t] &\leq \left( \frac{2\alpha(n+1)}{s+1} - \frac{s(n+2)}{n(s+1)} \right) \mathbb{E} \|w_t - \mu_t\|^2 \\ &\quad + \left( \frac{2\alpha(n+1)s}{(s+1)n} - \frac{s}{n(s+1)} \right) \sum_{i=1}^n \mathbb{E} \|w_t^i - \mu_t\|^2 \\ &\quad + \left( (n-s) + \frac{n+1}{2\alpha} + \frac{(n-s)^2}{s+1} \right) \frac{s^2}{n(n+1)^2} \eta^2 \sum_{i=1}^n \mathbb{E} [\|\tilde{h}_{t+1}^i\|^2].\end{aligned}$$

516 We let  $\alpha = \frac{1}{4(n+1)}$  and define  $\kappa = \frac{1}{n} \left( \frac{s(n-s)}{2(n+1)(s+1)} \right)$  to simply as following:

$$\begin{aligned}\mathbb{E}[\Phi_{t+1}] - \mathbb{E}[\Phi_t] &\leq -\kappa \mathbb{E} \|w_t - \mu_t\|^2 - \kappa \sum_{i=1}^n \mathbb{E} \|w_t^i - \mu_t\|^2 \\ &\quad + \left( (n-s) + 2(n+1)^2 + \frac{(n-s)^2}{s+1} \right) \frac{s^2}{n(n+1)^2} \eta^2 \sum_{i=1}^n \mathbb{E} [\|\tilde{h}_{t+1}^i\|^2].\end{aligned}$$

517 We now introduce  $\Phi_t$  on the right-hand side of the inequality above:

$$\mathbb{E}[\Phi_{t+1}] - \mathbb{E}[\Phi_t] \leq -\kappa \mathbb{E}[\Phi_t] + \left( (n-s) + 2(n+1)^2 + \frac{(n-s)^2}{s+1} \right) \frac{s^2}{n(n+1)^2} \eta^2 \sum_{i=1}^n \mathbb{E} [\|\tilde{h}_{t+1}^i\|^2].$$

518 We reorganize the terms to make the final statement:

$$\mathbb{E}[\Phi_{t+1}] \leq (1-\kappa) \mathbb{E}[\Phi_t] + 3 \frac{s^2}{n} \eta^2 \sum_{i=1}^n \mathbb{E} [\|\tilde{h}_{t+1}^i\|^2].$$

519 □

## 520 B.2.2 Bound expected local gradient Variance

521 In the next lemma we show that an analogous version of A3 holds in expectation.

522 **Lemma 13.** Assume A3. Let  $t \geq 1$  be a time step,  $q$  a local step, and  $i$  a client. We have:

$$\mathbb{E} [\|\tilde{h}_{t,q}^i\|^2] \leq \mathbb{E} [\|h_{t,q}^i\|^2] + \sigma^2.$$

523 *Proof.* We refer the reader to the filtrations  $(\mathcal{F}_{(t,q)})_{(t,q) \in I}$  defined in (5). By the tower property of  
524 conditional expectation, we have:

$$\mathbb{E} [\|\tilde{h}_{t,q}^i\|^2] = \mathbb{E} \left[ \mathbb{E} \left[ \|\tilde{h}_{t,q}^i\|^2 \mid \mathcal{F}_{(t,q-1)} \right] \right].$$

525 We denote by  $h_{t,q}^i$  the gradient of  $f_i$  at  $w_{t-1}^i - \sum_{s=1}^{q-1} \eta \tilde{h}_{t,s}^i$ . By construction,  $\mathbb{E} \left[ \tilde{h}_{t,q}^i \mid \mathcal{F}_{(t,q-1)} \right] =$   
 526  $h_{t,q}^i$ . By A3, we conclude that:

$$\mathbb{E} \left[ \mathbb{E} \left[ \|\tilde{h}_{t,q}^i\|^2 \mid \mathcal{F}_{(t,q-1)} \right] \right] \leq \mathbb{E}[\sigma^2 + \|h_{t,q}^i\|^2] = \mathbb{E}[\|h_{t,q}^i\|^2] + \sigma^2.$$

527

□

528 In the following we define  $w_{t,q}^i = w_{t-1}^i - \sum_{s=1}^q \eta \tilde{h}_{t,s}^i$ . This is the model of client  $i$ , at time step  $t$   
 529 and at local step  $q$ . Therefore,  $\tilde{h}_{t,q}^i$  is a stochastic gradient of  $f_i$  computed at the point  $w_{t,q-1}^i$ . The  
 530 next lemma sets an upper bound on the gradients of quantized weights for each client. We show that  
 531 such quantities can be upper bounded by an expression containing the true gradient at the "average  
 532 model"  $\mu_t$ . For any agent  $i$ , and time step  $t \geq 0$ , define the quantity:

$$B_t^i = \frac{\sigma^2}{K^2} + 16L^2 \mathbb{E} \|w_t^i - \mu_t\|^2 + 8\mathbb{E} \|\nabla f_i(\mu_t)\|^2. \quad (9)$$

533 **Lemma 14.** Assume A3, and that the learning rate  $\eta$  satisfies  $\eta < \frac{1}{4LK^2}$ . Under the assumptions of  
 534 Lemma 13, then, for any agent  $i$ , time step  $t \geq 0$  and local step  $q$ , the following inequality holds:

$$\mathbb{E}[\|h_{t+1,q}^i\|^2] \leq B_t^i. \quad (10)$$

535 *Proof.* We will prove the result by induction on  $q$ . Initially, we show inequalities that are necessary  
 536 for both the base case  $q = 1$  and for the general case.

$$\mathbb{E}[\|h_{t+1,q}^i\|^2] = \mathbb{E} \|\nabla f_i(w_{t+1,q-1}^i)\|^2$$

537 We introduce the gradient on the virtual point  $\mu_t$ :

$$\begin{aligned} \mathbb{E} \|\nabla f_i(w_{t+1,q-1}^i)\|^2 &\leq \mathbb{E} \left\| \left( \nabla f_i \left( w_t^i - \sum_{s=1}^{q-1} \eta \tilde{h}_{t+1,s}^i \right) - \nabla f_i(\mu_t) \right) + \nabla f_i(\mu_t) \right\|^2 \\ &\leq 2\mathbb{E} \left\| \nabla f_i \left( w_t^i - \sum_{s=1}^{q-1} \eta \tilde{h}_{t+1,s}^i \right) - \nabla f_i(\mu_t) \right\|^2 + 2\mathbb{E} \|\nabla f_i(\mu_t)\|^2 \\ &\leq 2L^2 \mathbb{E} \left\| w_t^i - \sum_{s=1}^{q-1} \eta \tilde{h}_{t+1,s}^i - \mu_t \right\|^2 + 2\mathbb{E} \|\nabla f_i(\mu_t)\|^2 \\ &\leq 4L^2 \mathbb{E} \|w_t^i - \mu_t\|^2 + 4\eta^2 L^2 (q-1) \sum_{s=1}^{q-1} \mathbb{E} \|\tilde{h}_{t+1,s}^i\|^2 + 2\mathbb{E} \|\nabla f_i(\mu_t)\|^2. \end{aligned}$$

538 Applying this result with  $q = 1$  shows that (10) holds. For  $q \geq 1$ , we proceed by induction. First, we  
 539 apply Lemma 13:

$$\mathbb{E} \|\nabla f_i(w_{t+1,q-1}^i)\|^2 \leq 4L^2 \mathbb{E} \|w_t^i - \mu_t\|^2 + 4\eta^2 L^2 (q-1) \sum_{s=1}^{q-1} \left( \mathbb{E} \|h_{t+1,s}^i\|^2 + \sigma^2 \right) + 2\mathbb{E} \|\nabla f_i(\mu_t)\|^2.$$

540 Using the induction hypothesis, we have:

$$\mathbb{E} \|\nabla f_i(w_{t+1,q-1}^i)\|^2 \leq 4L^2 \mathbb{E} \|w_t^i - \mu_t\|^2 + 4\eta^2 L^2 (q-1)^2 (B_t^i + \sigma^2) + 2\mathbb{E} \|\nabla f_i(\mu_t)\|^2.$$

541 Now we use that  $\eta < \frac{1}{4LK^2}$  and  $q \leq K$ .

$$\begin{aligned} \mathbb{E} \|\nabla f_i(w_{t+1,q-1}^i)\|^2 &\leq 4L^2 \mathbb{E} \|w_t^i - \mu_t\|^2 + \frac{4}{16K^2} (B_t^i + \sigma^2) + 2\mathbb{E} \|\nabla f_i(\mu_t)\|^2 \\ &\leq 4L^2 \mathbb{E} \|w_t^i - \mu_t\|^2 + \frac{\sigma^2}{4K^2} + 2\mathbb{E} \|\nabla f_i(\mu_t)\|^2 + \frac{B_t^i}{4} \end{aligned}$$

542 Finally, we get:

$$\mathbb{E} \|h_{t+1,q}^i\|^2 \leq 8L^2 \mathbb{E} \|w_t^i - \mu_t\|^2 + \frac{\sigma^2}{2K^2} + 4\mathbb{E} \|\nabla f_i(\mu_t)\|^2 + \frac{B_t^i}{2}$$

543 It then suffices to see that the last term above is upper bounded by  $B_t^i$ . □

544 In Lemma 14 we have found a way to bound  $h_{t+1,q}^i$ . The goal of the next lemma is to use this result  
 545 to find an upper bound for the stochastic gradients  $\tilde{h}_{t+1,q}^i$ .

546 **Corollary 15.** *Under the assumptions of Lemma 14, for any local step  $q$ , agent  $i$ , and step  $t \geq 0$ , the*  
 547 *following holds:*

$$\mathbb{E} \|\tilde{h}_{t+1,q}^i\|^2 \leq (\sigma^2 + B_t^i).$$

548 The next lemma gives an upper bound on the difference of the gradient at the average model  $\mu_t$  and  
 549 the expected value of the updates computed by the clients. In particular, the next lemma shows how  
 550 well the  $h_{t+1,q}^i$  approximate the true gradients  $\nabla f_i(\mu_t)$ . For any  $i \in \{1, \dots, n\}$  and  $t \geq 0$ , define

$$C_t^i = 4L^2\eta^2K^2\sigma^2 + 20L^2\mathbb{E} \|w_t^i - \mu_t\|^2 + 16L^2\eta^2K^2\mathbb{E} \|\nabla f_i(\mu_t)\|^2.$$

551 **Lemma 16.** *Assume the learning rate  $\eta$  satisfies  $\eta < \frac{1}{2LK}$ . Under the assumptions of Corollary 15,*  
 552 *for any  $i \in \{1, \dots, n\}$ ,  $t \geq 0$  and  $q \in \{1, \dots, K\}$ , it holds that :*

$$\mathbb{E} \|\nabla f_i(\mu_t) - h_{t+1,q}^i\|^2 \leq C_t^i.$$

*Proof.*

$$\begin{aligned} \mathbb{E} \|\nabla f_i(\mu_t) - h_{t+1,q}^i\|^2 &= \mathbb{E} \|\nabla f_i(\mu_t) - \nabla f_i(w_{t+1,q-1}^i)\|^2 \\ &\leq L^2\mathbb{E} \|\mu_t - w_{t+1,q-1}^i\|^2. \end{aligned}$$

553 We can now decompose the client drift as:

$$\begin{aligned} \mathbb{E} \|\mu_t - w_{t+1,q-1}^i\|^2 &= \mathbb{E} \left\| \mu_t - w_t^i + \sum_{s=1}^q \eta \tilde{h}_{t+1,s}^i \right\|^2 \\ &\leq 2\mathbb{E} \|w_t^i - \mu_t\|^2 + 2\eta^2\mathbb{E} \left\| \sum_{s=1}^q \tilde{h}_{t+1,s}^i \right\|^2 \\ &\leq 2\mathbb{E} \|w_t^i - \mu_t\|^2 + 2\eta^2q \sum_{s=1}^q \mathbb{E} \|\tilde{h}_{t+1,s}^i\|^2. \end{aligned}$$

554 By using Corollary 15, we get

$$\mathbb{E} \|\mu_t - w_{t+1,q-1}^i\|^2 \leq 2\mathbb{E} \|w_t^i - \mu_t\|^2 + 2\eta^2K^2(\sigma^2 + B_t^i),$$

555 where  $B_t^i$  is defined in (9). Combining the two bounds, we get:

$$\mathbb{E} \|\nabla f_i(\mu_t) - h_{t+1,q}^i\|^2 \leq 2L^2\mathbb{E} \|w_t^i - \mu_t\|^2 + 2L^2\eta^2K^2(\sigma^2 + B_t^i).$$

556 Expanding the above inequality:

$$\begin{aligned} \mathbb{E} \|\nabla f_i(\mu_t) - h_{t+1,q}^i\|^2 &\leq 2L^2\mathbb{E} \|w_t^i - \mu_t\|^2 \\ &\quad + 2L^2\eta^2K^2(\sigma^2 + \frac{\sigma^2}{K^2} + 16L^2\mathbb{E} \|w_t^i - \mu_t\|^2 + 8\mathbb{E} \|\nabla f_i(\mu_t)\|^2) \\ &\leq 4L^2\eta^2K^2\sigma^2 + 20L^2\mathbb{E} \|w_t^i - \mu_t\|^2 + 16L^2\eta^2K^2\mathbb{E} \|\nabla f_i(\mu_t)\|^2. \end{aligned}$$

557 As claimed. □

558 **Lemma 17.** *Under assumptions of Lemma 16, we have*

$$\mathbb{E} \langle \nabla f(\mu_t), -h_{t+1,q}^i \rangle \leq \frac{\mathbb{E} [\|\nabla f(\mu_t)\|^2]}{4} + C_t^i - \mathbb{E} [\langle \nabla f(\mu_t), \nabla f_i(\mu_t) \rangle].$$

559

560 *Proof.* We may manipulate the equation above to get:

$$\mathbb{E} \langle \nabla f(\mu_t), -h_{t+1}^i \rangle = \mathbb{E} \langle \nabla f(\mu_t), \nabla f_i(\mu_t) - h_{t+1,q}^i \rangle - \mathbb{E} \langle \nabla f(\mu_t), \nabla f_i(\mu_t) \rangle.$$

561 Using Young's inequality together with Lemma 16 we can upper bound  
562  $\mathbb{E} \langle \nabla f(\mu_t), \nabla f_i(\mu_t) - h_{t+1,q}^i \rangle$  by

$$\frac{\mathbb{E} \|\nabla f(\mu_t)\|^2}{4} + \mathbb{E} \|\nabla f_i(\mu_t) - h_{t+1,q}^i\|^2 \leq \frac{\mathbb{E} \|\nabla f(\mu_t)\|^2}{4} + C_t^i.$$

563 This concludes the proof.  $\square$

564 The next lemma incorporates the idea behind gradient descent. We find an upper bound for the  
565 expected value of the inner product between the true gradients  $\nabla f(\mu_t)$  and the client updates  $-\tilde{h}_{t+1}^i$ .  
566 In particular, we seek to show that, in expectation  $-\tilde{h}_{t+1}^i$  is a descent direction for the function  $f$ .  
567 In other words, that the updates proposed by the clients contribute to getting  $\mu_t$  closer to a local  
568 minimum.

569 **Lemma 18.** Assume A4. We denote by  $E_{t+1}^i$  the effective number of locals steps done by a client  $i$   
570 while being called by the central server. We clip to  $K$  and consider the random variable  $E_{t+1}^i \wedge K$ .  
571 Under assumptions of Lemma 16, and for any time step  $t > 0$ , we have:

$$\sum_{i=1}^n \mathbb{E} \langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \tilde{h}_{t+1}^i \rangle \leq 20L^2 \mathbb{E} [\Phi_t] + 4nL^2\eta^2 K^2(\sigma^2 + 4G^2) + n \left( 16L^2\eta^2 K^2 B^2 - \frac{3}{4} \right) \mathbb{E} \|\nabla f(\mu_t)\|^2.$$

572 for

$$\alpha^i = \begin{cases} \mathbf{P}(E_{t+1}^i > 0) E_{t+1}^i \wedge K \\ \mathbb{E}[E_{t+1}^i \wedge K]. \end{cases}$$

573 *Proof.* Initially, we introduce indicator random variables in order to work with the  $E_{t+1}^i$  terms. We  
574 also introduce  $Z^i$  the following random variable as :

$$Z^i = \begin{cases} 0 & \text{if } E_{t+1}^i < 1 \\ \langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \sum_{q=1}^{E_{t+1}^i} \tilde{h}_{t+1,q}^i \rangle & \text{if } 1 \leq E_{t+1}^i \leq K \\ \langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \sum_{q=1}^K \tilde{h}_{t+1,q}^i \rangle & \text{if } E_{t+1}^i > K. \end{cases}$$

575 We first claim that  $\mathbb{E} \langle \nabla f(\mu_t), \tilde{h}_{t+1,q}^i \rangle = \mathbb{E} \langle \nabla f(\mu_t), h_{t+1,q}^i \rangle$ . This result follows from the following  
576 algebraic manipulations. First, notice that:

$$\mathbb{E} \langle \nabla f(\mu_t), \tilde{h}_{t+1,q}^i \rangle = \mathbb{E} \langle \nabla f(\mu_t), \tilde{h}_{t+1,q}^i - h_{t+1,q}^i \rangle + \mathbb{E} \langle \nabla f(\mu_t), h_{t+1,q}^i \rangle.$$

577 Now, we recall that  $\mathbb{E}[\langle \nabla f(\mu_t), \tilde{h}_{t+1,q}^i - h_{t+1,q}^i \rangle | \mathcal{F}_{(t+1,q-1)}] = 0$ . Therefore:

$$\mathbb{E} \langle \nabla f(\mu_t), \tilde{h}_{t+1,q}^i - h_{t+1,q}^i \rangle = \mathbb{E}[\mathbb{E}[\langle \nabla f(\mu_t), \tilde{h}_{t+1,q}^i - h_{t+1,q}^i \rangle | \mathcal{F}_{(t+1,q-1)}]] = 0.$$

578 Now notice that  $E_{t+1}^i$  is independent of the random variables  $\nabla f(\mu_t)$  and  $\tilde{h}_{t+1,q}^i$ . Therefore:

$$\begin{aligned} \sum_{i=1}^n \mathbb{E} \left\langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \tilde{h}_{t+1}^i \right\rangle &= \sum_i^n \mathbb{E}[Z^i] \\ &= \sum_{i=1}^n \mathbb{E}[\mathbb{1}[E_{t+1}^i > K] \langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \sum_q^K \tilde{h}_{t+1,q}^i \rangle \\ &\quad + \mathbb{1}[1 \leq E_{t+1}^i \leq K] \langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \sum_q^K \mathbb{1}[q \leq E_{t+1}^i] \tilde{h}_{t+1,q}^i \rangle] \\ &= \sum_i^n \sum_q^K \mathbb{E} \left[ \frac{\mathbb{1}[E_{t+1}^i \geq 1] \mathbb{1}[q \leq (E_{t+1}^i \wedge K)]}{\alpha^i} \langle \nabla f(\mu_t), -\tilde{h}_{t+1,q}^i \rangle \right] \\ &= \sum_i^n \mathbb{E} \left[ \frac{\mathbb{1}[E_{t+1}^i \geq 1]}{\alpha^i} \sum_q^{E_{t+1}^i \wedge K} \langle \nabla f(\mu_t), -\tilde{h}_{t+1,q}^i \rangle \right]. \end{aligned}$$

579 We now apply Lemma 17 to obtain the following:

$$\begin{aligned}\mathbb{E} \left\langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \tilde{h}_{t+1}^i \right\rangle &\leq \mathbb{E} \left[ \frac{\mathbb{1}[E_{t+1}^i \geq 1]}{\alpha^i} \sum_q^{E_{t+1}^i \wedge K} \mathbb{E}[\langle \nabla f(\mu_t), -\tilde{h}_{t+1,q}^i | E_{t+1}^i \rangle] \right] \\ &\leq \mathbb{E} \left[ \frac{\mathbb{1}[E_{t+1}^i \geq 1]}{\alpha^i} \sum_q^{E_{t+1}^i \wedge K} \left( \frac{\mathbb{E}[\|\nabla f(\mu_t)\|^2]}{4} + C_t^i - \mathbb{E}[\langle \nabla f(\mu_t), \nabla f_i(\mu_t) \rangle] \right) \right]\end{aligned}$$

580 We can make use of Lemma 10 with  $\alpha^i = \mathbf{P}(E_{t+1}^i > 0)E_{t+1}^i \wedge K$  or  $\alpha^i = \mathbb{E}[E_{t+1}^i \wedge K]$ , and

581  $S = E_{t+1}^i \wedge K$ ,  $Y_q = \mathbb{E}[\langle \nabla f(\mu_t), -\tilde{h}_{t+1,q}^i | E_{t+1}^i \rangle]$  to achieve the following:

$$\sum_{i=1}^n \mathbb{E} \left\langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \tilde{h}_{t+1}^i \right\rangle \leq \sum_{i=1}^n \frac{\mathbb{E}[\|\nabla f(\mu_t)\|^2]}{4} + C_t^i - \mathbb{E}[\langle \nabla f(\mu_t), \nabla f_i(\mu_t) \rangle].$$

582 Now we use that  $\sum_{i=1}^n \frac{f_i(w)}{n} = f(w)$ , for any vector  $w \in \mathbb{R}^d$ .

$$\sum_{i=1}^n \mathbb{E} \left\langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \tilde{h}_{t+1}^i \right\rangle \leq \frac{n \mathbb{E} \|\nabla f(\mu_t)\|^2}{4} - n \mathbb{E} \|\nabla f(\mu_t)\|^2 + \sum_{i=1}^n C_t^i.$$

583 Finally, we compute:

$$\begin{aligned}\sum_{i=1}^n C_t^i &= \sum_{i=1}^n \left( 4L^2 \eta^2 K^2 \sigma^2 + 20L^2 \mathbb{E} \|w_t^i - \mu_t\|^2 + 16L^2 \eta^2 K^2 \mathbb{E} \|\nabla f_i(\mu_t)\|^2 \right) \\ &= 4nL^2 \eta^2 K^2 \sigma^2 + 20L^2 \sum_{i=1}^n \mathbb{E} \|w_t^i - \mu_t\|^2 + 16L^2 \eta^2 K^2 \sum_{i=1}^n \mathbb{E} \|\nabla f_i(\mu_t)\|^2.\end{aligned}$$

584 We can then use assumption A4:

$$\sum_{i=1}^n C_t^i \leq 4nL^2 \eta^2 K^2 \sigma^2 + 20L^2 \mathbb{E} [\Phi_t] + 16nL^2 \eta^2 K^2 \left( G^2 + B^2 \mathbb{E} \|\nabla f(\mu_t)\|^2 \right).$$

585 In conclusion, we get the following upper bound for  $\sum_{i=1}^n \mathbb{E} \left\langle \nabla f(\mu_t), -\frac{1}{\alpha^i} \tilde{h}_{t+1}^i \right\rangle$ :

$$20L^2 \mathbb{E} [\Phi_t] + 4nL^2 \eta^2 K^2 (\sigma^2 + 4G^2) + n \left( 16L^2 \eta^2 K^2 B^2 - \frac{3}{4} \right) \mathbb{E} \|\nabla f(\mu_t)\|^2.$$

586 □

### 587 B.3 Bound Sum of expected local gradient variance

588 The next lemma gives a bound on the expected update computed by clients at time step  $t$ . The result  
589 is useful, for example, in setting an upper bound on how much the average model  $\mu_t$  changes between  
590 time steps. The proof follows a similar reasoning as the proof of the previous lemma.

591 **Lemma 19.** Assume A3 and A4. Under assumptions of Lemma 14, and for any step  $t$ , we have that:

$$\begin{aligned}\sum_i^n \mathbb{E} \left[ \left\| \frac{1}{\mathbf{P}(E_{t+1}^i > 0)E_{t+1}^i \wedge K} \tilde{h}_{t+1}^i \right\|^2 \right] &\leq \sigma^2 \sum_i^n \left( \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbb{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \right) + 16L^2 \mathbb{E} [\Phi_t] \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \\ &\quad + 8n \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) B^2 \mathbb{E} [\|\nabla f(\mu_t)\|^2] + 8n \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) G^2 \\ \sum_i^n \mathbb{E} \left[ \left\| \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i \right\|^2 \right] &\leq \sigma^2 \sum_i^n \left( \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]} \right) + 16L^2 \mathbb{E} [\Phi_t] \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \\ &\quad + 8n \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) B^2 \mathbb{E} [\|\nabla f(\mu_t)\|^2] + 8n \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) G^2.\end{aligned}$$



592 *Proof.* For  $\alpha^i = \mathbf{P}(E_{t+1}^i > 0)E_{t+1}^i \wedge K$  or  $\alpha^i = \mathbb{E}[E_{t+1}^i \wedge K]$ , we have:

$$\sum_{i=1}^n \mathbb{E} \left[ \left\| \frac{1}{\alpha^i} \tilde{h}_{t+1}^i \right\|^2 \right] = \sum_{i=1}^n \text{Var} \left( \frac{1}{\alpha^i} \sum_{q=1}^{E_{t+1}^i \wedge K} \tilde{h}_{t+1,q}^i \right) + \sum_i^n \|\mathbb{E}[\tilde{h}_{t+1}^i]\|^2.$$

593 Recall that for any  $q \in \{1, \dots, K\}$ , the random variables  $E_{t+1}^i$  and  $\tilde{h}_{t+1,q}^i$  are independent. For clarity  
 594 we reuse the notation where  $\tilde{h}_{t+1}^i = \frac{1}{\mathbf{P}(E_{t+1}^i > 0)(E_{t+1}^i \wedge K)} \tilde{h}_{t+1}^i$  or  $\tilde{h}_{t+1}^i = \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i$ . Therefore  
 595 we can apply Lemma 11 with  $S = E_{t+1}^i \wedge K$  and  $Y_q = \tilde{h}_{t+1,q}^i$ :

$$\begin{cases} \mathbb{E} \left[ \left\| \frac{1}{\mathbf{P}(E_{t+1}^i > 0)E_{t+1}^i \wedge K} \tilde{h}_{t+1}^i \right\|^2 \right] \leq \|\mathbb{E}[\tilde{h}_{t+1}^i]\|^2 + \frac{\|\mathbb{E}[\tilde{h}_{t+1}^i]\|^2}{\mathbf{P}(E_{t+1}^i > 0)^2} (\mathbf{P}(E_{t+1}^i > 0)(1 - \mathbf{P}(E_{t+1}^i > 0))) + \frac{\text{Var}(\tilde{h}_{t+1,q}^i)}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbf{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \\ \mathbb{E} \left[ \left\| \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i \right\|^2 \right] \leq \|\mathbb{E}[\tilde{h}_{t+1}^i]\|^2 + \frac{\text{Var}(\tilde{h}_{t+1,q}^i)}{\mathbb{E}[E_{t+1}^i \wedge K]^2} + \frac{\|\mathbb{E}[\tilde{h}_{t+1,q}^i]\|^2 \text{Var}(E_{t+1}^i \wedge K)}{\mathbb{E}[E_{t+1}^i \wedge K]^2}. \end{cases}$$

596 We now use Corollary 15 to get:

$$\begin{cases} \mathbb{E} \left[ \left\| \frac{1}{\mathbf{P}(E_{t+1}^i > 0)E_{t+1}^i \wedge K} \tilde{h}_{t+1}^i \right\|^2 \right] \leq \|\mathbb{E}[\tilde{h}_{t+1}^i]\|^2 \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} + \frac{\sigma^2 + B_t^i - \|\mathbb{E}[\tilde{h}_{t+1,q}^i]\|^2}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbf{1}[E_{t+1}^i > 0]}{E_{t+1}^i \wedge K} \right] \\ \mathbb{E} \left[ \left\| \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i \right\|^2 \right] \leq \|\mathbb{E}[\tilde{h}_{t+1}^i]\|^2 (1 + \frac{\text{Var}(E_{t+1}^i \wedge K)}{\mathbb{E}[E_{t+1}^i \wedge K]^2}) + \frac{\sigma^2 + B_t^i - \|\mathbb{E}[\tilde{h}_{t+1,q}^i]\|^2}{\mathbb{E}[E_{t+1}^i \wedge K]}. \end{cases}$$

597 Hence we can use Lemma 10 with  $S = E_{t+1}^i \wedge K$  and  $Y_q = \tilde{h}_{t+1,q}^i$  to simplify as following:

$$\begin{cases} \mathbb{E} \left[ \left\| \frac{1}{\mathbf{P}(E_{t+1}^i > 0)E_{t+1}^i \wedge K} \tilde{h}_{t+1}^i \right\|^2 \right] \leq \|\mathbb{E}[\tilde{h}_{t+1,q}^i]\|^2 \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} + \frac{\sigma^2 + B_t^i - \|\mathbb{E}[\tilde{h}_{t+1,q}^i]\|^2}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbf{1}[E_{t+1}^i > 0]}{E_{t+1}^i \wedge K} \right] \\ \mathbb{E} \left[ \left\| \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i \right\|^2 \right] \leq \|\mathbb{E}[\tilde{h}_{t+1,q}^i]\|^2 (1 + \frac{\text{Var}(E_{t+1}^i \wedge K)}{\mathbb{E}[E_{t+1}^i \wedge K]^2}) + \frac{\sigma^2 + B_t^i - \|\mathbb{E}[\tilde{h}_{t+1,q}^i]\|^2}{\mathbb{E}[E_{t+1}^i \wedge K]}. \end{cases}$$

598 We refer the reader to the filtrations  $(\mathcal{F}_{(t,q)})_{(t,q) \in I}$  defined in (5). Now we can express the expected  
 599 value of  $\tilde{h}_{t+1,q}^i$  (as  $\mu$  following the notations from Lemma 11), and upper bound its square norm by  
 600 Lemma 14 :

$$\begin{aligned} \|\mu\|^2 &= \|\mathbb{E}[\mathbb{E}[\tilde{h}_{t+1,q}^i \mid \mathcal{F}_{(t,q-1)}]]\|^2 \\ &\leq \mathbb{E}[\|\mathbb{E}[\tilde{h}_{t+1,q}^i \mid \mathcal{F}_{(t,q-1)}]\|^2] \\ &\leq \mathbb{E}[\|h_{t+1,q}^i\|^2] \\ &\leq B_t^i \end{aligned}$$

601 We can insert this bound in the above inequations:

$$\begin{cases} \mathbb{E} \left[ \left\| \frac{1}{\mathbf{P}(E_{t+1}^i > 0)E_{t+1}^i \wedge K} \tilde{h}_{t+1}^i \right\|^2 \right] \leq B_t^i \frac{\mathbf{P}(E_{t+1}^i > 0) - \mathbb{E} \left[ \frac{\mathbf{1}[E_{t+1}^i > 0]}{E_{t+1}^i \wedge K} \right]}{\mathbf{P}(E_{t+1}^i > 0)^2} + \frac{\sigma^2 + B_t^i}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbf{1}[E_{t+1}^i > 0]}{E_{t+1}^i \wedge K} \right] \\ \mathbb{E} \left[ \left\| \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i \right\|^2 \right] \leq B_t^i \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2] - \mathbb{E}[E_{t+1}^i \wedge K]}{\mathbb{E}[E_{t+1}^i \wedge K]^2} + \frac{\sigma^2 + B_t^i}{\mathbb{E}[E_{t+1}^i \wedge K]}. \end{cases}$$

602 This simplifies as:

$$\begin{cases} \mathbb{E} \left[ \left\| \frac{1}{\mathbf{P}(E_{t+1}^i > 0)E_{t+1}^i \wedge K} \tilde{h}_{t+1}^i \right\|^2 \right] \leq \frac{B_t^i}{\mathbf{P}(E_{t+1}^i > 0)} + \frac{\sigma^2}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbf{1}[E_{t+1}^i > 0]}{E_{t+1}^i \wedge K} \right] \\ \mathbb{E} \left[ \left\| \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i \right\|^2 \right] \leq B_t^i \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]^2} + \frac{\sigma^2}{\mathbb{E}[E_{t+1}^i \wedge K]}. \end{cases}$$

603 Expanding  $B_t^i$  and summing from  $i$  to  $n$ , we get:

$$\begin{aligned}
\sum_i^n \mathbb{E} \left[ \left\| \frac{1}{\mathbf{P}(E_{t+1}^i > 0) E_{t+1}^i \wedge K} \tilde{h}_{t+1}^i \right\|^2 \right] &\leq \sigma^2 \sum_i^n \left( \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbb{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \right) + 16L^2 \mathbb{E}[\Phi_t] \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \\
&\quad + 8 \sum_i^n \max_j \left( \frac{1}{\mathbf{P}(E_{t+1}^j > 0)} \right) \mathbb{E}[\|\nabla f(\mu_t)\|^2] \\
\sum_i^n \mathbb{E} \left[ \left\| \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i \right\|^2 \right] &\leq \sigma^2 \sum_i^n \left( \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]} \right) + 16L^2 \mathbb{E}[\Phi_t] \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \\
&\quad + 8 \sum_i^n \max_j \left( \frac{\mathbb{E}[(E_{t+1}^j \wedge K)^2]}{\mathbb{E}[E_{t+1}^j \wedge K]} \right) \mathbb{E}[\|\nabla f(\mu_t)\|^2].
\end{aligned}$$

604 **Remark 20.** Here we loose a lot: we have upper bounded the term  $\sum_i^n \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \|w_t^i - \mu_t\|^2 \leq$   
605  $\sum_i^n \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \|w_t^i - \mu_t\|^2$ . But still, our bounds stay better than Zakerinia et al. (2022)'s  
606 ones.

607 In order to complete the proof, we combine assumption A4 together with the above inequalities

$$\begin{aligned}
\sum_i^n \mathbb{E} \left[ \left\| \frac{1}{\mathbf{P}(E_{t+1}^i > 0) E_{t+1}^i \wedge K} \tilde{h}_{t+1}^i \right\|^2 \right] &\leq \sigma^2 \sum_i^n \left( \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbb{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \right) + 16L^2 \mathbb{E}[\Phi_t] \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \\
&\quad + \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) (8nB^2 \mathbb{E}[\|\nabla f(\mu_t)\|^2] + 8nG^2) \\
\sum_i^n \mathbb{E} \left[ \left\| \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} \tilde{h}_{t+1}^i \right\|^2 \right] &\leq \sigma^2 \sum_i^n \left( \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]} \right) + 16L^2 \mathbb{E}[\Phi_t] \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \\
&\quad + 8n \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) B^2 \mathbb{E}[\|\nabla f(\mu_t)\|^2] + 8n \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) B^2.
\end{aligned}$$

608 □

#### 609 B.4 Bound the sum (over time) of expected potential

610 **Lemma 21.** Assume that  $\eta \leq \frac{1}{20sL \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right)}, \frac{1}{20sL \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right)}$ . Under the assumptions  
611 of Lemmas 2 and 19, and for any time step  $t$  we have:

$$\begin{aligned}
\mathbb{E}[\Phi_{t+1}] &\leq \left(1 - \frac{1}{5n}\right) \mathbb{E}[\Phi_t] + 3s^2 \eta^2 \left( \frac{1}{n} \sum_i^n \left( \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbb{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \right) + 8 \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) G^2 \right) \\
&\quad + 24B^2 s^2 \eta^2 \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \mathbb{E}[\|\nabla f(\mu_t)\|^2]
\end{aligned}$$

612

$$\begin{aligned}
\mathbb{E}[\Phi_{t+1}] &\leq \left(1 - \frac{1}{5n}\right) \mathbb{E}[\Phi_t] + 3s^2 \eta^2 \left( \frac{1}{n} \sum_i^n \left( \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]} \right) + 8 \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) G^2 \right) \\
&\quad + 24B^2 s^2 \eta^2 \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \mathbb{E}[\|\nabla f(\mu_t)\|^2]
\end{aligned}$$

613 *Proof.* We first use Lemma 2:

$$\mathbb{E}[\Phi_{t+1}] \leq (1 - \kappa) \mathbb{E}[\Phi_t] + 3 \frac{s^2}{n} \eta^2 \sum_{i=1}^n \mathbb{E} \left[ \frac{1}{\alpha^{i2}} \left\| \tilde{h}_{t+1}^i \right\|^2 \right].$$

614 , with  $\alpha^i = \mathbf{P}(E_{t+1}^i > 0)(E_{t+1}^i \wedge K)$  or  $\alpha^i = \mathbb{E}[E_{t+1}^i \wedge K]$ . Now we expand the quantity above  
615 using the inequality in Lemma 19:

$$\begin{aligned} \mathbb{E}[\Phi_{t+1}] &\leq (1 - \kappa) \mathbb{E}[\Phi_t] + 3 \frac{s^2}{n} \eta^2 \left( \sigma^2 \sum_i^n a^i + b(16L^2 \mathbb{E}[\Phi_t] + 8nB^2 \mathbb{E}[\|\nabla f(\mu_t)\|^2]) + 8nG^2 \right) \\ &\leq \left( 1 - \frac{1}{n} \left( \frac{s(n-s)}{2(n+1)(s+1)} \right) + 48 \frac{s^2}{n} L^2 b \eta^2 \right) \mathbb{E}[\Phi_t] + 3 \frac{s^2}{n} \left( \sigma^2 \sum_i^n a^i + 8nbG^2 \right) \eta^2 + 24B^2 s^2 b \eta^2 \mathbb{E} \|\nabla f(\mu_t)\|^2. \end{aligned}$$

616 With

$$\begin{cases} a^i, b = \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbb{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right], \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \\ a^i, b = \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]}, \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right). \end{cases}$$

617 To complete, we use  $\eta \leq \frac{1}{20sLb}$ :

$$\mathbb{E}[\Phi_{t+1}] \leq \left( 1 - \frac{1}{5n} \right) \mathbb{E}[\Phi_t] + 3s^2 \eta^2 \left( \sigma^2 \frac{1}{n} \sum_i^n a^i + 8bG^2 \right) + 24B^2 s^2 \eta^2 b \mathbb{E} \|\nabla f(\mu_t)\|^2.$$

618 □

619 In the next lemma, we bound the cumulative sum of potential functions.

620 **Lemma 22.** *Let  $T$  be a positive integer. Under the assumptions of Lemma 21, the following inequality*  
621 *holds:*

$$\begin{aligned} \sum_{t=0}^T \mathbb{E}[\Phi_t] &\leq 120nB^2 s^2 \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \eta^2 \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 \\ &\quad + 15Ts^2 \eta^2 \left( \sigma^2 \sum_i^n \left( \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbb{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \right) + 8n \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) G^2 \right). \\ 622 \sum_{t=0}^T \mathbb{E}[\Phi_t] &\leq 120nB^2 s^2 \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \eta^2 \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 \\ &\quad + 15Ts^2 \eta^2 \left( \sigma^2 \sum_i^n \left( \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]} \right) + 8n \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) G^2 \right). \end{aligned}$$

623 *Proof.* From Lemma 21, we get that there exist  $\alpha, \beta$  not depending on  $t$  such that:

$$\mathbb{E}[\Phi_{t+1}] \leq \left( 1 - \frac{1}{5n} \right) \mathbb{E}[\Phi_t] + \alpha \mathbb{E} \|\nabla f(\mu_t)\|^2 + \beta.$$

624 Therefore:

$$\begin{aligned} \sum_{t=0}^{T-1} \mathbb{E}[\Phi_{t+1}] &\leq \sum_{t=0}^{T-1} \left( \left( 1 - \frac{1}{5n} \right) \mathbb{E}[\Phi_t] + \alpha \mathbb{E} \|\nabla f(\mu_t)\|^2 + \beta \right) \\ &\leq \left( 1 - \frac{1}{5n} \right) \sum_{t=0}^{T-1} \mathbb{E}[\Phi_t] + T\beta + \alpha \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2. \end{aligned}$$

625 Rearranging the terms in the sum we obtain the following:

$$\left( 1 - \frac{1}{5n} \right) \mathbb{E}[\Phi_0] + \frac{1}{5n} \sum_{t=1}^{T-1} \mathbb{E}[\Phi_t] + \mathbb{E}[\Phi_T] \leq T\beta + \alpha \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2.$$

626 From this inequality, we get:

$$\sum_{t=0}^T \mathbb{E} [\Phi_t] \leq 5n \left( T\beta + \alpha \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 \right).$$

627 Expanding on the values of  $\alpha, \beta$  obtained by Lemma 21, we get the desired result:

$$\sum_{t=0}^T \mathbb{E} [\Phi_t] \leq 5nT \frac{3s^2}{n} \eta^2 \left( \sigma^2 \sum_i^n a^i + 8nbG^2 \right) + 120nB^2s^2b\eta^2 \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2.$$

628 □

## 629 B.5 Bound the change in the average model

630 The next lemma upper bounds the expected change in the average model  $\mu_t$ .

631 **Lemma 23.** *For any time step  $t \geq 0$ ,*

$$\mathbb{E} \|\mu_{t+1} - \mu_t\|^2 \leq \frac{s^2\eta^2}{n(n+1)^2} \sum_{i=1}^n \mathbb{E} \|\check{h}_{t+1}^i\|^2.$$

632 *Proof.* Recall that

$$\mu_{t+1} - \mu_t = -\frac{\eta}{n+1} \sum_{i \in \mathcal{S}_{t+1}} \check{h}_{t+1}^i.$$

633 Therefore we may compute an upper bound:

$$\begin{aligned} \|\mu_{t+1} - \mu_t\|^2 &= \frac{\eta^2}{(n+1)^2} \left\| \sum_{i \in \mathcal{S}_{t+1}} \check{h}_{t+1}^i \right\|^2 \\ &\leq \frac{s\eta^2}{(n+1)^2} \sum_{i \in \mathcal{S}_{t+1}} \|\check{h}_{t+1}^i\|^2 \end{aligned}$$

634 We may then apply Lemma 9 to get the desired result:

$$\mathbb{E} \|\mu_{t+1} - \mu_t\|^2 \leq \frac{s^2\eta^2}{n(n+1)^2} \sum_{i=1}^n \mathbb{E} \|\check{h}_{t+1}^i\|^2.$$

635 □

636 We now give another upper bound on how the average model  $\mu_t$  changes at time step  $t$ .

637 **Lemma 24.** *Under the assumptions of Lemmas 19 and 23, and for any step  $t$ :*

$$\begin{aligned} \mathbb{E} \|\mu_{t+1} - \mu_t\|^2 &\leq \frac{s^2\eta^2\sigma^2}{n(n+1)^2} \sum_i^n \left( \frac{1}{K^2\mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbf{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \right) + \frac{16L^2s^2\eta^2}{n(n+1)^2} \mathbb{E}[\Phi_t] \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \\ &\quad + \frac{8s^2B^2\eta^2}{(n+1)^2} \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \mathbb{E}[\|\nabla f(\mu_t)\|^2] + \frac{8s^2G^2\eta^2}{(n+1)^2} \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \\ \mathbb{E} \|\mu_{t+1} - \mu_t\|^2 &\leq \frac{s^2\eta^2\sigma^2}{n(n+1)^2} \sum_i^n \left( \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2\mathbb{E}[E_{t+1}^i \wedge K]} \right) + \frac{16L^2s^2\eta^2}{n(n+1)^2} \mathbb{E}[\Phi_t] \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \\ &\quad + \frac{8s^2B^2\eta^2}{(n+1)^2} \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \mathbb{E}[\|\nabla f(\mu_t)\|^2] + \frac{8s^2G^2\eta^2}{(n+1)^2} \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right). \end{aligned}$$

638 *Proof.* For this proof, we will combine the inequality obtained from Lemma 23 to the one from  
 639 Lemma 19. This will be enough to obtain the desired result.

$$\mathbb{E}\|\mu_{t+1} - \mu_t\|^2 \leq \sum_{i=1}^n \frac{s^2 \eta^2}{n(n+1)^2} \mathbb{E}\|\tilde{h}_{t+1}^i\|^2.$$

640 Simplifying the above quantity, we get the desired inequality:

$$\begin{aligned} \mathbb{E}\|\mu_{t+1} - \mu_t\|^2 &\leq \frac{s^2 \eta^2 \sigma^2}{n(n+1)^2} \sum_i^n \left( \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E}\left[\frac{\mathbf{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K}\right] \right) + \frac{16L^2 s^2 \eta^2}{n(n+1)^2} \mathbb{E}[\Phi_t] \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \\ &\quad + \frac{8s^2 B^2 \eta^2}{(n+1)^2} \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \mathbb{E}[\|\nabla f(\mu_t)\|^2] + \frac{8s^2 G^2 \eta^2}{(n+1)^2} \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) G^2 \\ \mathbb{E}\|\mu_{t+1} - \mu_t\|^2 &\leq \frac{s^2 \eta^2 \sigma^2}{n(n+1)^2} \sum_i^n \left( \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]} \right) + \frac{16L^2 s^2 \eta^2}{n(n+1)^2} \mathbb{E}[\Phi_t] \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \\ &\quad + \frac{8s^2 B^2 \eta^2}{(n+1)^2} \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \mathbb{E}[\|\nabla f(\mu_t)\|^2] + \frac{8s^2 G^2 \eta^2}{(n+1)^2} \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) G^2. \end{aligned}$$

641 □

## 642 B.6 Convergence result

643 In this section, we use the lemmas proved so far to demonstrate Theorem 3. Following the proof, we  
 644 establish the learning rate  $\eta$  that results in the best overall rate of convergence.

645 *Proof.* Using  $L$ -smoothness, we have:

$$f(\mu_{t+1}) \leq f(\mu_t) + \langle \nabla f(\mu_t), \mu_{t+1} - \mu_t \rangle + \frac{L}{2} \|\mu_{t+1} - \mu_t\|^2. \quad (11)$$

646 First we look at the term  $\langle \nabla f(\mu_t), \mu_{t+1} - \mu_t \rangle$ . Recall that:

$$\mu_{t+1} - \mu_t = -\frac{\eta}{n+1} \sum_{i \in \mathcal{S}_{t+1}} \tilde{h}_{t+1}^i$$

647 by Lemma 9, we have:

$$\mathbb{E}_t[\mu_{t+1} - \mu_t] = -\frac{s\eta}{n(n+1)} \sum_{i=1}^n \tilde{h}_{t+1}^i$$

and subsequently

$$\mathbb{E}_t \langle \nabla f(\mu_t), \mu_{t+1} - \mu_t \rangle = \sum_{i=1}^n \frac{s\eta}{n(n+1)} \mathbb{E}_t \langle \nabla f(\mu_t), -\tilde{h}_{t+1}^i \rangle.$$

648 Hence, we can rewrite (11) as:

$$\mathbb{E}_t[f(\mu_{t+1})] \leq f(\mu_t) + \sum_{i=1}^n \frac{s\eta}{n(n+1)} \mathbb{E}_t \langle \nabla f(\mu_t), -\tilde{h}_{t+1}^i \rangle + \frac{L}{2} \mathbb{E}_t \|\mu_{t+1} - \mu_t\|^2.$$

649 Next, we remove the conditioning with the tower law of expectation:

$$\mathbb{E}[f(\mu_{t+1})] \leq \mathbb{E}[f(\mu_t)] + \sum_{i=1}^n \frac{s\eta}{n(n+1)} \mathbb{E} \langle \nabla f(\mu_t), -\tilde{h}_{t+1}^i \rangle + \frac{L}{2} \mathbb{E} \|\mu_{t+1} - \mu_t\|^2.$$

650 We now define some notation to simplify the computations. By Lemma 18, there exist  $a_1, a_2, a_3$  not  
 651 depending on  $t$  such that

$$\sum_{i=1}^n \mathbb{E} \langle \nabla f(\mu_t), -\tilde{h}_{t+1}^i \rangle \leq a_1 \mathbb{E}[\Phi_t] + a_2 \mathbb{E} \|\nabla f(\mu_t)\|^2 + a_3.$$

652 Similarly, by Lemma 24, there exist  $b_1, b_2, b_3$  not depending on  $t$  such that:

$$\mathbb{E} \|\mu_{t+1} - \mu_t\|^2 \leq b_1 \mathbb{E} [\Phi_t] + b_2 \mathbb{E} \|\nabla f(\mu_t)\|^2 + b_3.$$

653 Defining  $c_i = a_i \frac{s\eta}{n(n+1)} + b_i \frac{L}{2}$ , we have

$$\mathbb{E} [f(\mu_{t+1})] - \mathbb{E} [f(\mu_t)] \leq c_1 \mathbb{E} [\Phi_t] + c_2 \mathbb{E} \|\nabla f(\mu_t)\|^2 + c_3.$$

654 Summing the above inequality for  $t = 0, 1, \dots, T-1$  we get that:

$$\mathbb{E} [f(\mu_T)] - f(\mu_0) \leq c_1 \sum_{t=0}^{T-1} \mathbb{E} [\Phi_t] + c_2 \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 + c_3 T.$$

655 By Lemma 22 there exist  $d_1, d_2$  independent of  $T$  such that:

$$\sum_{t=0}^T \mathbb{E} [\Phi_t] \leq d_1 \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 + T d_2.$$

656 We then get:

$$\mathbb{E} [f(\mu_T)] - f(\mu_0) \leq (c_1 d_1 + c_2) \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 + T(c_1 d_2 + c_3).$$

657 We now assume that  $c_1 d_1 + c_2 < 0$ . Later in the proof, we will show that this is true for small enough  
658  $\eta$ . Using the fact that  $f(\mu_T) \geq f_*$  and rearranging the terms, we get:

$$\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 \leq \frac{f(\mu_0) - f_*}{T(-c_1 d_1 - c_2)} + \frac{c_1 d_2 + c_3}{-c_1 d_1 - c_2}.$$

659 Of course, now we expand each of these terms. Refer to Lemma 18, Lemma 24, and Lemma 22 for  
660 the specific values of the defined quantities  $a_i, b_i$ , and  $d_i$ . We have:

$$\begin{aligned} c_1 &= \frac{s\eta}{n(n+1)} a_1 + \frac{L}{2} b_1 \\ &= 20L^2 \frac{s\eta}{n(n+1)} + \frac{16s^2\eta^2 L^2 b}{n(n+1)^2} \frac{L}{2} \\ &= \frac{4L^2 s\eta}{n(n+1)^2} (5(n+1) + 2s\eta Lb). \end{aligned}$$

661 We recall here the definition from Lemma 21

$$\begin{cases} a^i, b = \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbf{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right], \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \\ a^i, b = \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]}, \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right). \end{cases}$$

662 By using  $\eta \leq \frac{n+1}{2sLb}$  we get:

$$c_1 \leq \frac{24L^2 s\eta}{n(n+1)}.$$

663 Therefore:

$$\begin{aligned} c_1 d_1 &\leq \frac{24L^2 s\eta}{n(n+1)} 120nB^2 s^2 b \eta^2 \\ &\leq \frac{2880L^2 s^3 \eta^3 B^2 b}{n+1}. \end{aligned}$$

664 Moreover:

$$\begin{aligned} c_2 &= \frac{s\eta}{n(n+1)} a_2 + \frac{L}{2} b_2 \\ &= \frac{s\eta}{n(n+1)} \left( n \left( 16L^2 \eta^2 K^2 B^2 - \frac{3}{4} \right) \right) + \frac{L}{2} \left( \frac{8s^2 \eta^2 b B^2}{(n+1)^2} \right) \\ &= \frac{s\eta}{(n+1)} \left( 16L^2 \eta^2 K^2 B^2 - \frac{3}{4} \right) + \frac{4Ls^2 \eta^2 b B^2}{(n+1)^2}. \end{aligned}$$

665 By using  $\eta \leq \frac{1}{20B^2bKLs}$  we get:

$$c_1d_1 \leq \frac{2880}{8000(n+1)LB^4K^3b^2}$$

666

$$\begin{aligned} c_2 &\leq \frac{s\eta}{n+1} \left( \frac{16}{400s^2b^2B^2} - \frac{3}{4} \right) + \frac{4}{400(n+1)^2LbB^2K^2} \\ &\leq \frac{-s\eta}{2(n+1)} - c_1d_1. \end{aligned}$$

667 Therefore we conclude that  $-c_2 - c_1d_1 \geq \frac{s\eta}{2(n+1)}$ , which is greater than 0. Now we compute:

$$\begin{aligned} c_1d_2 &\leq \frac{24L^2s\eta}{n(n+1)} \left( 15s^2\eta^2 \left( \sigma^2 \sum_i^n a^i + 8nbG^2 \right) \right) \\ &\leq \frac{360L^2s^3\eta^3}{(n+1)} \left( \sigma^2 \frac{1}{n} \sum_i^n a^i + 8bG^2 \right). \end{aligned}$$

668 And additionally:

$$\begin{aligned} c_3 &= \frac{s\eta}{n(n+1)}a_3 + \frac{L}{2}b_3 \\ &= \frac{s\eta}{n(n+1)} (4nL^2\eta^2K^2(\sigma^2 + 4G^2)) + \frac{L}{2} \left( \frac{s^2\eta^2\sigma^2}{n(n+1)^2} \sum_i^n a^i + \frac{8s^2G^2\eta^2}{(n+1)^2}b \right). \end{aligned}$$

669 Thus

$$\begin{aligned} c_3 + c_1d_2 &\leq \left( \frac{4L^2\eta^2K^2s\eta}{(n+1)} + \frac{Ls^2\eta^2}{2n(n+1)^2} \sum_i^n a^i + \frac{360L^2s^3\eta^3}{n(n+1)} \sum_i^n a^i \right) \sigma^2 \\ &\quad + \left( \frac{16L^2\eta^2K^2s\eta}{(n+1)} + \frac{4Ls^2\eta^2}{(n+1)^2}b + \frac{2880bL^2s^3\eta^3}{(n+1)} \right) G^2. \end{aligned}$$

670 And therefore:

$$\begin{aligned} \frac{c_3 + c_1d_2}{-c_1d_1 - c_2} &\leq \left( 8L^2\eta^2K^2 + \frac{Ls\eta}{n(n+1)} \sum_i^n a^i + \frac{720L^2s^2\eta^2}{n} \sum_i^n a^i \right) \sigma^2 \\ &\quad + \left( 32L^2\eta^2K^2 + \frac{8Ls\eta}{(n+1)}b + 5600bL^2s^2\eta^2 \right) G^2. \end{aligned}$$

671 Finally:

$$\begin{aligned} \frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 &\leq 2(n+1) \frac{f(\mu_0) - f_*}{Ts\eta} \\ &\quad + \left( 8L^2\eta^2K^2 + \frac{Ls\eta}{n(n+1)} \sum_i^n a^i + \frac{720L^2s^2\eta^2}{n} \sum_i^n a^i \right) \sigma^2 \\ &\quad + \left( 32L^2\eta^2K^2 + \frac{8Ls\eta}{(n+1)}b + 5600bL^2s^2\eta^2 \right) G^2. \end{aligned}$$

672

□

673 In particular for stochastic reweighting:



$$\begin{aligned}
\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 &\leq 2(n+1) \frac{f(\mu_0) - f_*}{Ts\eta} \\
&+ \left( 8L^2\eta^2 K^2 + \frac{Ls\eta}{n(n+1)} \sum_i^n \left( \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbf{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \right) \right) \sigma^2 \\
&+ \left( \frac{720L^2 s^2 \eta^2}{n} \sum_i^n \left( \frac{1}{K^2 \mathbf{P}(E_{t+1}^i > 0)} + \frac{1}{\mathbf{P}(E_{t+1}^i > 0)^2} \mathbb{E} \left[ \frac{\mathbf{1}(E_{t+1}^i > 0)}{E_{t+1}^i \wedge K} \right] \right) \right) \sigma^2 \\
&+ \left( 32L^2\eta^2 K^2 + \frac{8Ls\eta}{(n+1)} \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) + 5600L^2 s^2 \eta^2 \max_i \left( \frac{1}{\mathbf{P}(E_{t+1}^i > 0)} \right) \right) G^2.
\end{aligned}$$

674 And particular for expectation reweighting:

$$\begin{aligned}
\frac{1}{T} \sum_{t=0}^{T-1} \mathbb{E} \|\nabla f(\mu_t)\|^2 &\leq 2(n+1) \frac{f(\mu_0) - f_*}{Ts\eta} \\
&+ \left( 8L^2\eta^2 K^2 + \frac{Ls\eta}{n(n+1)} \sum_i^n \left( \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]} \right) + \frac{720L^2 s^2 \eta^2}{n} \sum_i^n \left( \frac{1}{\mathbb{E}[E_{t+1}^i \wedge K]} + \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{K^2 \mathbb{E}[E_{t+1}^i \wedge K]} \right) \right) \sigma^2 \\
&+ \left( 32L^2\eta^2 K^2 + \frac{8Ls\eta}{(n+1)} \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) + 5600L^2 s^2 \eta^2 \max_i \left( \frac{\mathbb{E}[(E_{t+1}^i \wedge K)^2]}{\mathbb{E}[E_{t+1}^i \wedge K]} \right) \right) G^2.
\end{aligned}$$

## C Detailed simulation environment

From Algorithm 1, one must note that local weights are reset with the central model only when being contacted by the central server. Hence initially we have  $w_0^i = w_0$ , but at time  $t$  we may have  $w_t^i \neq w_t$ , see Figure 4.

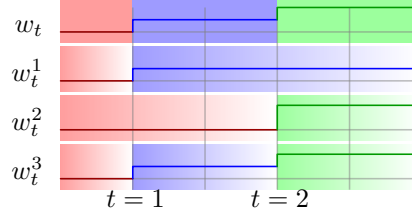


Figure 4: Example of asynchronous updates with  $n = 3$  nodes and selection size  $s = 2$ . At  $t = 0$ , all clients are initialized with the same value. At time  $t = 1$ , clients  $\{1, 3\}$  are selected, and at time  $t = 2$ , clients  $\{2, 3\}$ . At time  $t = 2$ , client 2 is reporting updates computed on outdated parameter.

### C.1 Implementation of concurrent works

In Section 5 we have simulated experiments and run the code for the concurrent approaches FedAvg, QuAFL, and FedBuff. FedAvg is a standard synchronous method. At the beginning of each round, the central node  $s$  selects clients uniformly at random and broadcast its current model. Each of these clients take the central server value and then performs exactly  $K$  local steps, and then sends the resulting model progress back to the server. The server then computes the average of the  $s$  received models and updates its model. In this synchronous structure, the server must wait in each round for the slowest client to complete its update. QuAFL is an asynchronous method that randomly selects  $s$  clients at each server invocation. The server then replaces its model with a convex combination of the received models and its current model. Also, the  $s$  receiving clients replace their local model with a convex combination between their current model and the model of the receiving server. In FedBuff, clients compute local training asynchronously as well, with the help of a buffer. Once the buffer is filled with  $Z$  different client updates, the server averages the buffer updates and performs a gradient step on the computed average. Then the buffer is reset to zero and the available clients get the server model as a new starting point.

### C.2 Discussion on simulated runtime

We based our simulations mainly on the code developed by Nguyen et al. (2022): we assume a server and  $n$  clients, each of which initially has a model copy. We assume that, at each time step  $t$  (for the central server), a batch of  $s$  clients are sampled at random without replacement. For the client  $i$ , the inter-arrival time of two successive requests are therefore independent and distributed according to a geometric distributions of parameter  $s/n$ . The time elapsed from the last renewal is distributed according to the stationary distribution of the age process (assuming that the renewal is stationary), which is also distributed according to a geometric random variable with the same parameter  $s/n$ . We assume that the clients have different computational speeds. For this purpose  $E_t^i$  is distributed according to a geometrical distribution of parameter  $\lambda^i$ :  $E_t^i \sim \text{Geom}(\lambda^i)$ . The parameter  $\lambda^i$  is  $1/2$  for fast clients and  $1/16$  for slow clients; the expected running time  $\mathbb{E}[E_t^i]$  is 2 and 16, respectively. The training dataset is distributed among the clients so that each of them has access to an equal portion of the training data (whether it is IID or non-I IID). We track the performance of each algorithm by evaluating the server’s model against an unseen validation dataset. We measure the loss and accuracy of the model in terms of simulation time, server steps, and total local steps taken by clients.

To adequately capture the time spent on the server side for computations and orchestration of centralized learning, two quantities are implemented: the server waiting time (the time the server waits between two consecutive calls ) and the server interaction time (the time the server takes to send and receive the required data). In all experiments, they are set to 4 and 3, respectively.

For each global step, the FedAvg runtime is the sum of the server interaction time (see above) and the local step runtime of the slowest selected client times the number of maximum epochs  $K$  (we wait

until all clients have computed all their local epochs in this synchronous setting). For QuAFL and FAVAS, the duration of a global step is simply the sum of the server interaction time and the server waiting time. For FedBuff, the runtime is the sum of the server interaction time and the time spent feeding the buffer of size  $Z$ . The waiting time for feeding the buffer depends on the respective local runtimes of the slow and fast clients, as well as on the ratio between slow and fast clients: in the code, we reset a counter at the beginning of each global step and read the runtime when the  $Z^{th}$  local update arrives.

### C.3 Detailed results

Below we provide further insight into the experiments described in Section 5. We present figures for loss, variance ( $\sum_{i=1}^n \|w_t^i - w_t\|^2$ ), but also for accuracy (evaluated on the held-out test set on the server side) in terms of time, but also in terms of total local steps and total server steps.

We find that FAVAS and other asynchronous methods, when time rather than the number of server steps FAVAS - and more generally asynchronous methods - can achieve significant speedups on these metrics compared to FedAvg. This is due to asynchronous communication allowing rounds to complete faster without always having to wait for slower nodes to complete their local computations. Although this behaviour is simulated, we believe it reflects the practical potential of FAVAS.

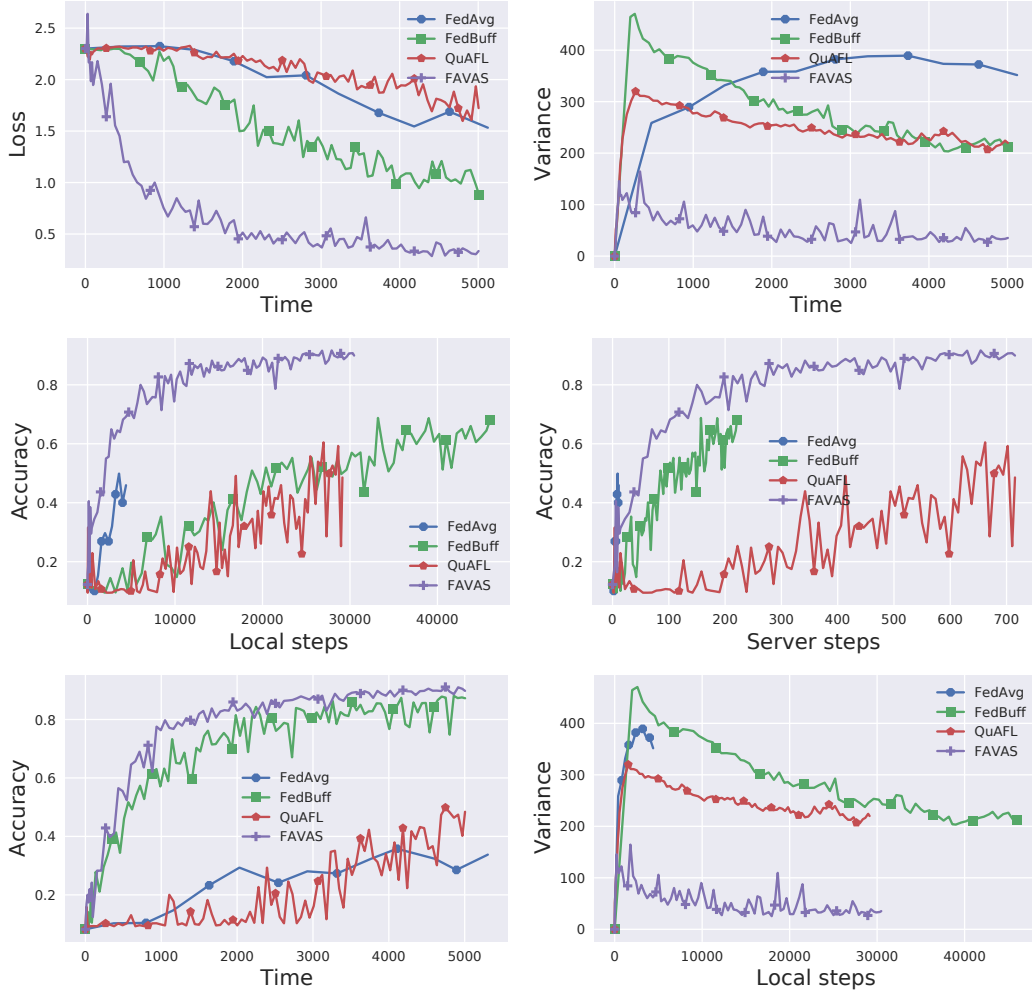


Figure 5: Validation loss/accuracy and variance on the MNIST dataset with a non-iid split in between  $n = 100$  total nodes. In this particular experiment, one ninth of the clients are defined as fast.

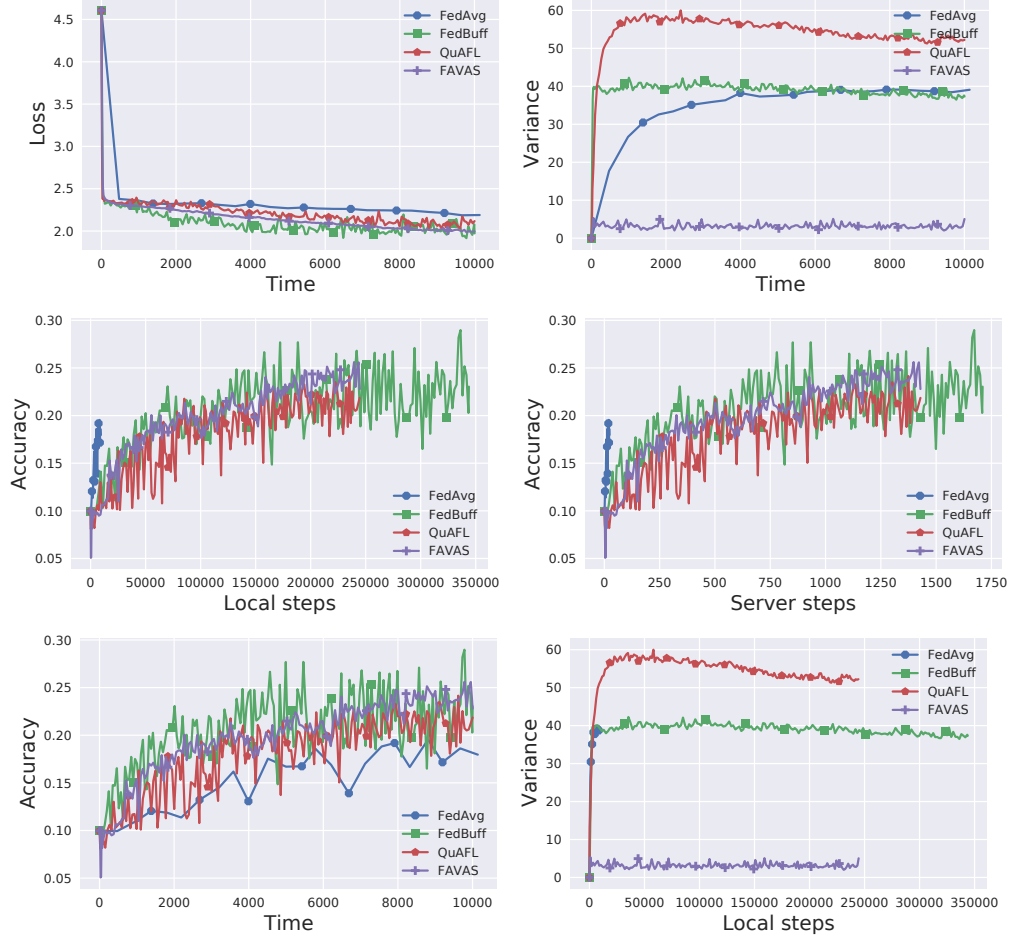


Figure 6: Validation loss/accuracy and variance on the CIFAR-10 dataset with a non-iid split in between  $n = 100$  total nodes.

731 We refer to FAVAS[QNN] when training a neural network with low bit precision arithmetic. We ran the  
 732 code<sup>3</sup> from LUQ (Chmiel et al., 2021) and adapted it to our datasets and the FL framework. During  
 733 FAVAS[QNN] training, 3-bits quantization for weights and activation are used, 4 bits quantization for  
 734 neural gradients is used.

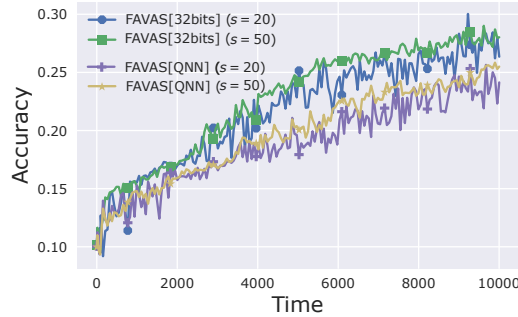


Figure 7: Validation accuracy on the CIFAR-10 dataset with a non-iid split in between  $n = 100$  total nodes. The amount  $s$  of selected clients at each round is varied. FAVAS[QNN] is the quantized version of FAVAS[32bits].

<sup>3</sup><https://openreview.net/forum?id=clwYez4n8e8>

735 In Figure 7 we analyse the effects of quantization and the influence of the number of randomly  
736 selected clients  $s$  on the convergence behaviour. As expected, we find that higher  $s$  improve the  
737 performance of FAVAS. Quantizing the neural network degrades the convergence behaviour of the  
738 algorithm, but, even if the weights and activation functions are highly quantized - as in the scenario  
739 we are considering-, the results are close to their full-precision counterpart.