SECOND-ORDER GUARANTEES IN FEDERATED LEARNING

Stefan Vlaski, Elsa Rizk and Ali H. Sayed

School of Engineering, École Polytechnique Fédérale de Lausanne

ABSTRACT

Federated learning is a useful framework for centralized learning from distributed data under practical considerations of heterogeneity, asynchrony, and privacy. Federated architectures are frequently deployed in deep learning settings, which generally give rise to non-convex optimization problems. Nevertheless, most existing analysis are either limited to convex loss functions, or only establish first-order stationarity, despite the fact that saddle-points, which are first-order stationary, are known to pose bottlenecks in deep learning. We draw on recent results on the second-order optimality of stochastic gradient algorithms in centralized and decentralized settings, and establish second-order guarantees for a class of federated learning algorithms.

1. INTRODUCTION

Federated learning pursues solutions to global optimization problems over distributed collections of agents by relying on the exchange of model updates in lieu of raw data. Federated architectures are frequently deployed in highly heterogeneous environments, where different agents have access to data of varying quality and varying computational resources. Performance guarantees for federated architectures are generally limited to convex loss functions, or to establishing limiting first-order stationarity on non-convex losses. First-order stationary points include minima, but can be saddle-points or local maxima as well. Saddle-points in particular have been identified as bottlenecks for optimization algorithms in many important applications, such as deep learning [2,3]. It is hence desirable to devise algorithms and performance analyses that ensure efficient escape from saddle-points despite high levels of asynchrony and heterogeneity. Recent works have identified gradient perturbations as playing a key role in guaranteeing efficient saddle-points escape in centralized and fully decentralized architectures [4–10]. Here, we establish analogous results in the federated learning framework, extending recent analysis from [1] to allow for multiple local updates.

Specifically, we consider a collection of K agents, where each agent k is equipped with a risk loss function $J_k(w)$, which is defined as the expectation of a loss $Q(w; \boldsymbol{x}_k)$:

$$J_k(w) \triangleq \mathbb{E}_{\boldsymbol{x}_k} Q(w; \boldsymbol{x}_k) \qquad (1)$$

Here, $Q(w; \boldsymbol{x}_k)$ quantifies the fit of the model parametrization w to the random data $\boldsymbol{x}_k$. Note that we allow for the data $\boldsymbol{x}_k$ to vary with the agent index k, resulting in different risk functions $J_k(w)$ at different agents. It is common in multi-agent settings to pursue a

model w^o that performs well on average by solving:

$$w^o \triangleq \arg\min_w \sum_{k=1}^{K} p_k J_k(w) \qquad (2)$$

where the $\{p_k\}_{k=1}^{K}$ denote non-negative weights, normalized to add up to one without loss of generality. It is common to let $p_k = \frac{1}{K}$, hence giving equal weight to every agent k. In settings where agents are heterogeneous, and exhibit varying amounts of data, or varying computational capabilities, heterogeneous weights p_k can result in improved performance, which we allow for generality. Perhaps the most straightforward approach to pursuing w^o is by means of gradient descent, applied directly to (2), resulting in:

$$w_i = w_{i-1} - \mu \sum_{k=1}^{K} p_k \nabla J_k(w_{i-1}) = w_{i-1} - \mu \nabla J(w_{i-1}) \quad (3)$$

where we defined $J(\cdot) \triangleq \sum_{k=1}^{K} p_k J_k(\cdot)$. This implementation has two important drawbacks, which render it impractical in a federated learning setting. First, it requires full agent participation at every iteration, by means of computation and communication of $\nabla J_k(w_{i-1})$ with a central aggregator. In federated learning applications, where agents may or may not be able to participate in the update at any given iteration, this can cause bottlenecks. Second, evaluation of the exact gradient $\nabla J_k(w_{i-1})$ may be infeasible or costly, since it depends on the full distribution of $\boldsymbol{x}_k$ through its expectation in (1).

1.1. Related Works

Distributed algorithms for solving aggregate optimization problems similar to (2) can be broadly classified into those that involve communication with a centralized parameter server [11–14], and those that operate in a fully decentralized manner through peer-to-peer interactions [15–19]. Federated Averaging (FedAvg) was introduced in [20], and has sparked a number of studies and extensions, including FedDane [21], FedProx [22], hierarchical FedAvg [23], and dynamic FedAvg [24]. While the pursuit of an optimal average model as in (2) is most common, multi-task variations have been introduced as well, both in a federated [25] and decentralized settings [26].

Most prior works on federated learning and the FedAvg algorithm focus on convex risk functions [13, 14, 24], or establish first-order stationarity in non-convex environments [21–23, 27–29]. On the other hand, saddle points, which are first-order stationary, have been identified as bottlenecks in many learning applications, including deep learning [2]. This contrast to the empirical success of deep learning has motivated a number of recent works to consider the ability of gradient descent algorithms to escape saddle-points and find "good" local minima, both in centralized [4–8, 30, 31] and decentralized settings [9, 10, 32, 33]. The broad take-away from these works is that perturbations, either to the initialization or gradient updates,

play a key role in pushing iterates away from strict-saddle points and toward local minimizers. In this work, we extend these results to the federated learning setting, where agents may take an arbitrary number of local steps before communicating with the central parameter server.

2. ALGORITHM FORMULATION

2.1. The Federated Averaging Scheme

The need for full and exact agent participation in evaluating (3) in a federated setting is addressed in the stochastic federated averaging (FedAvg) framework [20]. To this end, the parameter server selects at iteration i a subset of L agents, collected in the set $\mathcal{N}_i$. We introduce a random indicator variable $\mathbb{1}_{k,i}$, which indicates whether agent k participates at time i, i.e., $\mathbb{1}_{k,i} = 1 \iff k \in \mathcal{N}_i$, and 0 otherwise. We assume for simplicity that agents are sampled uniformly at random, resulting in:

$$\Pr\{\mathbb{1}_{k,i} = 1\} = \mathbb{E}\{\mathbb{1}_{k,i}\} = \frac{L}{K} \tag{4}$$

Then, the parameter server provides participating agents with the current aggregate model $\boldsymbol{w}_{i-1}$. They use the model to initialize their local iterate to $\boldsymbol{w}_{k,0} = \boldsymbol{w}_{i-1}$ and then perform E_k local stochastic update steps for $e = 1, \ldots, E_k$:

$$\boldsymbol{w}_{k,e} = \boldsymbol{w}_{k,e-1} - \mu K \mathbb{1}_{k,i} \frac{p_k}{E_k} \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) \tag{5}$$

Here, $\widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1})$ denotes a generic stochastic approximation of the gradient $\nabla J_k(\boldsymbol{w}_{k,e-1})$. Using realizations for the random variable $\boldsymbol{x}_k$, it is common to construct $\widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) \triangleq \nabla Q(\boldsymbol{w}_{k,e-1}, \boldsymbol{x}_{k,e})$, resulting in stochastic gradient descent — we will discuss other constructions and their advantages in Section 2.2 below. The updated models are then fused by the central aggregator according to:

$$\boldsymbol{w}_i = \frac{1}{L} \sum_{k=1}^{K} \mathbb{1}_{k,i} \boldsymbol{w}_{k,E_k} \tag{6}$$

2.2. A General Stochastic Approximation Framework

We now present a number of choices for the stochastic gradient approximation $\widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1})$ to illustrate the generality of (5).

Example 1 (Mini-Batch SGD). Given a collection of B_k samples $\{\boldsymbol{x}_{k,e,b}\}_{b=1}^{B_k}$, constructing:

$$\widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) = \frac{1}{B_k} \sum_{b=1}^{B_k} \nabla Q(\boldsymbol{w}_{k,e-1}, \boldsymbol{x}_{k,e,b}) \tag{7}$$

yields mini-batch stochastic gradient descent, or simply stochastic gradient descent when $B_k = 1$. □

Example 2 (Perturbed SGD). It has been observed, both empirically and analytically, that adding additional perturbations to the stochastic gradient update can improve the performance of the gradient descent algorithm in non-convex settings [6]. In the presence of privacy concerns, perturbations to update directions can also be added in order to ensure differential privacy [34]. This corresponds to constructing:

$$\widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) = \nabla Q(\boldsymbol{w}_{k,e-1}, \boldsymbol{x}_{k,e}) + \boldsymbol{v}_{k,e} \tag{8}$$

where $\boldsymbol{v}_{k,e}$ denotes i.i.d. perturbation noise with zero mean, following for example a Gaussian or Laplacian distribution. □

Example 3 (Straggling Agents). Consider a setting where agents may be unreliable, in the sense that, despite being chosen by the parameter server to participate at iteration i, they may fail to return a locally updated model $\boldsymbol{w}_{k,E_k}$ by the time the server needs to re-aggregate models in (6). Such a setting can be modeled via:

$$\widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) = \begin{cases} \frac{1}{\delta_k} \nabla Q(\boldsymbol{w}_{k,e-1}, \boldsymbol{x}_{k,e}) & \text{with prob. } \delta_k, \\ 0 & \text{otherwise.} \end{cases} \tag{9}$$

Here, the scaling factor $\frac{1}{\delta_k}$ has been added to ensure unbiased gradient approximations, by allowing agents who participate less frequently to take larger steps. Alternative stochastic models for asynchronous behavior are possible as well [35]. □

It can be readily verified, that all three constructions in Examples 1–3 are unbiased approximations of the true gradient $\nabla J_k(\boldsymbol{w}_{k,e-1})$, i.e.:

$$\mathbb{E}\left\{\widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) | \boldsymbol{w}_{k,e-1}\right\} = \nabla J_k(\boldsymbol{w}_{k,e-1}) \tag{10}$$

Nevertheless, the stochastic nature of the approximation induces a gradient noise into the evolution of the algorithm, which we denote by:

$$\boldsymbol{s}_{k,e}(\boldsymbol{w}_{k,e-1}) \triangleq \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) - \nabla J_k(\boldsymbol{w}_{k,e-1}) \tag{11}$$

We impose the following general conditions on the stochastic gradient noise process, and hence the construction of the stochastic gradient approximation itself.

Assumption 1 (Gradient Noise Process). *The gradient noise process* (11) *satisfies:*

$$\mathbb{E}\{\boldsymbol{s}_{k,e}(\boldsymbol{w}_{k,e-1}) | \boldsymbol{w}_{k,e-1}\} = 0 \tag{12}$$

$$\mathbb{E}\{\| \boldsymbol{s}_{k,e}(\boldsymbol{w}_{k,e-1})\|^4 | \boldsymbol{w}_{k,e-1}\} \le \beta_k^4 \|\nabla J_k(\boldsymbol{w}_{k,e-1})\|^4 + \sigma_k^4 \tag{13}$$

for some $\beta_k^4, \sigma_k^4 \ge 0$. It is assumed that the gradient noise process is mutually independent over space and time, after conditioning on the current iterate:

$$\mathbb{E}\left\{\boldsymbol{s}_{k_1,e_1}(\boldsymbol{w}) \, \boldsymbol{s}_{k_2,e_2}(\boldsymbol{w})^{\mathsf{T}} | \boldsymbol{w}\right\} = 0 \quad \forall \, k_1 \ne k_2 \text{ or } e_1 \ne e_2 \tag{14}$$

and the gradient noise covariance:

$$R_{s,k}(\boldsymbol{w}_{k,e-1}) \triangleq \mathbb{E}\left\{\boldsymbol{s}_{k,e}(\boldsymbol{w}_{k,e-1}) \, \boldsymbol{s}_{k,e}(\boldsymbol{w}_{k,e-1})^{\mathsf{T}} | \boldsymbol{w}_{k,e-1}\right\} \tag{15}$$

is smooth:

$$\|R_{s,k}(x) - R_{s,k}(y)\| \le \beta_R \|x - y\|^\gamma \tag{16}$$

for some β_R and $0 < \gamma \le 4$, and there is a gradient noise component (in the aggregate) in every direction:

$$R_{s,k}(x) \ge \sigma_\ell^2 I, \quad \forall x \tag{17}$$

□

Relation (12) ensures that the stochastic gradient approximation is unbiased, while (13) imposes a relative bound on the fourth-order moment [17]. In light of Jensen's inequality, it is stronger than imposing a bound on the gradient noise variance, but will allow us to more granularly study the impact of the gradient noise around saddle-points; on the other hand, it is weaker than the more common conditions of bounded noise with probability one, or a sub-Gaussian condition [6, 7]. Relation (16) ensures that the distribution of the stochastic gradient noise process is locally smooth, allowing us to formulate an accurate short-term model around saddle-points [8]. It has previously been utilized to analyze in detail the

steady-state behavior of stochastic gradient algorithms in convex environments [17]. The persistent noise condition (17) will allow recursions to efficiently escape saddle-points by relying on the aggregate effect of the noise coupled with the local instability of saddle-points. It can be relaxed to only require a noise component to be present in the subspace of local descent directions [5, 8]. Since (17) can always be ensured by adding a small amount of isotropic perturbations to the stochastic gradient update as in (8), it will be sufficient, for simplicity, to impose (17) in this work.

3. PERFORMANCE ANALYSIS

3.1. A Perturbed Centralized Gradient Recursion

By iterating (5), we find for the final local update $\boldsymbol{w}_{k,E_k}$ sent back to the parameter server:

$$\boldsymbol{w}_{k,E_k} = \boldsymbol{w}_{i-1} - \mu K \mathbb{1}_{k,i} \frac{p_k}{E_k} \sum_{e=1}^{E_k} \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) \tag{18}$$

and after aggregation in (6):

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} - \mu \frac{K}{L} \sum_{k=1}^{K} \mathbb{1}_{k,i} \frac{p_k}{E_k} \sum_{e=1}^{E_k} \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) \tag{19}$$

We can reformulate this recursion to resemble the deterministic recursion (3) as:

$$\boldsymbol{w}_i = \boldsymbol{w}_{i-1} - \mu \sum_{k=1}^{K} p_k \nabla J_k(\boldsymbol{w}_{i-1}) - \mu \boldsymbol{s}_i - \mu \boldsymbol{d}_i \tag{20}$$

where $\boldsymbol{s}_i$ and $\boldsymbol{d}_i$ are perturbation terms:

$$\boldsymbol{s}_i \triangleq \frac{K}{L} \sum_{k=1}^{K} \mathbb{1}_{k,i} \frac{p_k}{E_k} \sum_{e=1}^{E_k} \widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1}) - \nabla J(\boldsymbol{w}_{i-1}) \tag{21}$$

$$\boldsymbol{d}_i \triangleq \frac{K}{L} \sum_{k=1}^{K} \mathbb{1}_{k,i} \frac{p_k}{E_k} \sum_{e=1}^{E_k} \left(\widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) - \widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1}) \right) \tag{22}$$

Comparing (20) with (3), we observe that the FedAvg implementation can be viewed as a perturbed gradient descent recursion. Perturbations have recently been shown to be instrumental in allowing local descent algorithms to escape from saddle-points and converge to local minima of non-convex loss functions. However, those studies are generally limited to assuming unbiased perturbations. In contrast, employing (5) with $E_k > 1$ results in biased gradient perturbations resulting from the term $\boldsymbol{d}_i$, rendering current analyses inapplicable. In this work, we generalize recent results on the second-order guarantees of stochastic gradient algorithms [8] to allow for biased gradient perturbations, and recover second-order guarantees for the FedAvg algorithm for heterogeneous agents. We describe and discuss the dependence of these guarantees on the various parameters of the architecture, such as agent participation rate, levels heterogeneity, asynchrony, and computational capabilities.

We introduce the following smoothness conditions to ensure that the impact of the perturbations (21)–(22) is limited.

Assumption 2 (Smoothness). *The local costs $J_k(w)$ are assumed to be smooth:*

$$\|\nabla J_k(x) - \nabla J_k(y)\| \leq \delta \|x - y\| \tag{23}$$

$$\left\|\nabla^2 J_k(x) - \nabla^2 J_k(y)\right\| \leq \rho \|x - y\| \tag{24}$$

Heterogeneity between agents is quantified by their gradient disagreement:

$$\|\nabla J_k(x) - \nabla J_\ell(x)\| \leq G \tag{25}$$

Furthermore, the costs themselves are assumed to be Lipschitz, implying uniformly bounded gradient:

$$\|\nabla J_k(x)\| \leq U \tag{26}$$

and the stochastic approximations of the gradient are Lipschitz in the mean-fourth sense:

$$\mathbb{E}\left\{ \left\|\widehat{\nabla J}_k^e(\boldsymbol{x}) - \widehat{\nabla J}_k^e(\boldsymbol{y})\right\|^4 \mid \boldsymbol{x}, \boldsymbol{y} \right\} \leq \hat{\delta}^4 \|\boldsymbol{x} - \boldsymbol{y}\|^4 \tag{27}$$

$\square$

3.2. Perturbation Bounds

Under the conditions on the stochastic gradient construction in Assumption 1, and the smoothness conditions in Assumption 2 we can bound the perturbations (21)–(22).

Lemma 1 (Perturbation Bounds). *The perturbations to recursion (20) are bounded as:*

$$\mathbb{E}\left\{ \boldsymbol{s}_i \mid \boldsymbol{w}_{i-1} \right\} = 0 \tag{28}$$

$$\mathbb{E}\left\{ \|\boldsymbol{s}_i\|^4 \mid \boldsymbol{w}_{i-1} \right\} \leq \overline{\beta}^4 \|\nabla J(\boldsymbol{w}_{i-1})\|^4 + \overline{\sigma}^4 \tag{29}$$

$$\mathbb{E}\left\{ \|\boldsymbol{d}_i\|^4 \mid \boldsymbol{w}_{i-1} \right\} \leq \mu^4 \sum_{k=1}^{K} p_k^5 \frac{K^6}{L^2} \hat{\delta}^4 8 \left(U^4 + \beta_k^4 U^4 + \sigma_k^4 \right) \tag{30}$$

where we introduced the constants:

$$\overline{\beta}^4 \triangleq \sum_{k=1}^{K} p_k \overline{\beta}_k^4 \tag{31}$$

$$\overline{\sigma}^4 \triangleq \sum_{k=1}^{K} p_k \overline{\sigma}_k^4 \tag{32}$$

$$\overline{\beta}_k^4 \triangleq 192 \frac{K^3}{L^3} \frac{\beta_k^4}{E_k^2} + 64 \frac{L}{K} \left(\frac{K-L}{L} \right)^4 + 64 \frac{K-L}{K} \tag{33}$$

$$\overline{\sigma}_k^4 \triangleq \left(192 \frac{K^3}{L^3} \frac{\beta_k^4}{E_k^2} + 64 \frac{L}{K} \left(\frac{K-L}{L} \right)^4 + 64 \frac{K-L}{K} \right) G^4$$
$$+ 24 \frac{K^3}{L^3} \frac{\sigma_k^4}{E_k^2} \tag{34}$$

The covariance of the aggregate gradient noise $\boldsymbol{s}_i$ evaluates to:

$$\mathbb{E}\left\{ \boldsymbol{s}_i \boldsymbol{s}_i^\mathsf{T} \mid \boldsymbol{w}_{i-1} \right\}$$
$$= \frac{K}{L} \sum_{k=1}^{K} \frac{p_k^2}{E_k} R_{s,k}(\boldsymbol{w}_{i-1}) + \frac{K}{L} \frac{K-L}{K-1} \sum_{k=1}^{K} t(\boldsymbol{w}_{i-1}) t(\boldsymbol{w}_{i-1})^\mathsf{T}$$
$$\geq \overline{\sigma}_\ell^2 I \triangleq \left(\sum_{k=1}^{K} \frac{p_k^2}{E_k} \right) \sigma_\ell^2 I \tag{35}$$

where $t(\boldsymbol{w}_{i-1})$ denotes the deviation:

$$t(\boldsymbol{w}_{i-1}) \triangleq p_k \nabla J_k(\boldsymbol{w}_{i-1}) - \frac{1}{K} \nabla J(\boldsymbol{w}_{i-1}) \tag{36}$$

Proof. Appendix A. $\square$

3.3. Second-Order Guarantees

Examination of the bounds (28)–(30) reveals that the aggregate zero-mean component s_i arising from the use of stochastic gradient approximations continues to be bounded in a manner similar to the local approximations (13), where the aggregate constant bounds are determined by the quality of local approximations $\{\beta_k^4, \sigma_k^4\}_{k=1}^K$, the participation rate $\frac{L}{K}$, the weights $\{p_k\}_{k=1}^K$, the level of heterogeneity G, and number of local updates taken E_k. The bias d_i induced by employing multiple local updates, on the other hand, does not have zero-mean. The bound on its fourth-order moment (30), however, is proportional to μ^4, causing its effect to be small for small step-sizes when compared to s_i, which is independent in μ. The fact that d_i is biased renders traditional second-order analysis of stochastic gradient algorithms [1, 4–6, 8] inapplicable to this setting, while the fact that its fourth-order moment is small compared to s_i makes it possible to extend the arguments of [1, 8].

Theorem 1. *Suppose the aggregate loss $J(w)$ is bounded from below by $J(w) \geq J^o$. Then, with probability $1 - 2\pi$:*

$$\|\nabla J(\boldsymbol{w}_{i^o})\|^2 \leq \mu \frac{\delta \overline{\sigma}^2}{1 - 2\mu\delta(1 + \overline{\beta}^2)}\left(1 + \frac{1}{\pi}\right) + O(\mu^2) \quad (37)$$

and $\lambda_{\min}\left(\nabla^2 J(\boldsymbol{w}_{i^o})\right) \geq -\tau$ in at most i^o iterations, where

$$i^o \leq \frac{2\left(J(w_0) - J^o\right)}{\mu^2 \delta \overline{\sigma}^2} i^s \quad (38)$$

and i^s denotes the saddle-point escape time:

$$i^s = \frac{\log\left(2M\frac{\overline{\sigma}^2}{\overline{\sigma}_\ell^2} + 1 + O(\mu)\right)}{\log\left(1 + 2\mu\tau\right)} \quad (39)$$

Proof. The argument is an adjustment of [8] by bounding away the effect of d_i. Details omitted due to space limitations. $\square$

This result ensures that, with probability $1 - 2\pi$, the FedAvg algorithm will return a second-order stationary point with $\|\nabla J(\boldsymbol{w}_{i^o})\|^2 \leq O(\mu)$ and $\lambda_{\min}\left(\nabla^2 J(\boldsymbol{w}_{i^o})\right) \geq -\tau$ in at most i^o iterations, where i^o scales polynomially with all problem parameters. Every second-order stationary point, in light of $\|\nabla J(\boldsymbol{w}_{i^o})\|^2 \leq O(\mu)$ is also first-order stationary, but the additional condition $\lambda_{\min}\left(\nabla^2 J(\boldsymbol{w}_{i^o})\right) \geq -\tau$ allows for the exclusion of strict saddle-points by choosing τ sufficiently small.

4. NUMERICAL RESULTS

We illustrate the ability of the FedAvg algorithm to escape saddle-points for:

$$Q(w_1, W_2; \boldsymbol{\gamma}, \boldsymbol{h}) \triangleq \log\left(1 + e^{-\gamma w_1^\top W_2 \boldsymbol{h}}\right) \quad (40)$$

$$J(w_1, W_2) \triangleq \mathbb{E}\, Q(w_1, W_2; \boldsymbol{\gamma}, \boldsymbol{h}) + \frac{\rho}{2}\|w_1\|^2 + \frac{\rho}{2}\|W_2\|^2 \quad (41)$$

This loss arises when training a neural network with a single, linear hidden layer to predict the class label $\boldsymbol{\gamma}$ from $\boldsymbol{h}$ using cross-entropy, and exhibits a strict saddle-point at $w_1 = W_2 = 0$, making it suitable as a simplified benchmark — see [9] for a discussion and motivation. For a total of $K = 100$ agents, we vary the rate of participation from $L = 1$ to $L = 100$. Agents are chosen uniformly, and
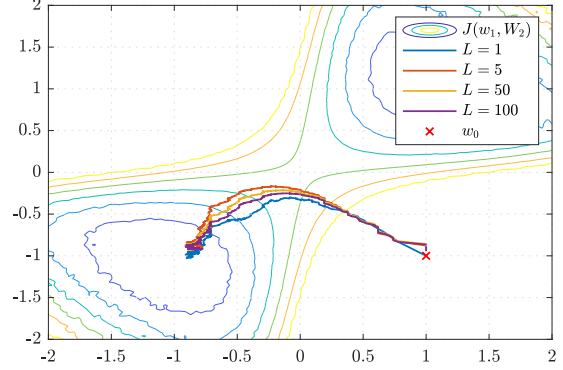
Fig. 1: Evolution of the aggregate model for various choices of the participation rate $\frac{L}{K}$. All implementations escape the saddle-point and find a local minimum.
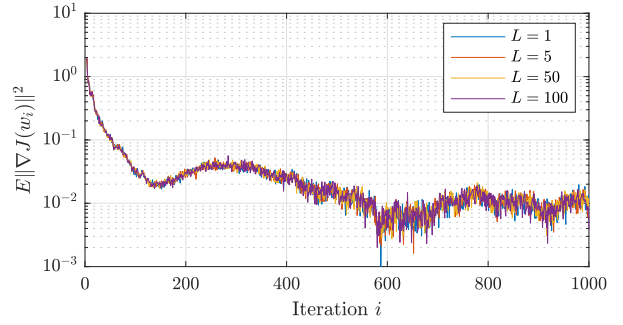
Fig. 2: Evolution of the gradient norm for varying participation rates.

participating agents perform $E = 10$ local updates constructed as a combination of Examples 2 and 3, namely:

$$\widehat{\nabla J_k^e}(\boldsymbol{w}_{k,e-1}) = \frac{1}{\delta_k}\left(\nabla Q(\boldsymbol{w}_{k,e-1}, \boldsymbol{x}_{k,e}) - \rho\, \boldsymbol{w}_{k,e-1}\right) \quad (42)$$

with probability $p_k = 0.5$, and $\widehat{\nabla J_k^e}(\boldsymbol{w}_{k,e-1}) = 0$ otherwise. Evolution of iterates and the gradient norm are shown in Figures 1 and 2 respectively.

5. CONCLUSION

In this work, we considered a highly heterogeneous and asynchronous variant of the Federated Averaging (FedAvg) algorithm, where agents may be using varying, potentially unreliable, stochastic gradient approximations with varying quality, and take a different number E_k of local update steps, and established convergence to second-order stationary points. Despite high levels of heterogeneity and asynchrony, the algorithm continues to escape saddle-points and return second-order stationary points in polynomial time, shedding light on the success of deep learning, which is frequently employed in federated learning settings.

6. REFERENCES

[1] S. Vlaski and A. H. Sayed, "Second-order guarantees in centralized, federated and decentralized non-convex optimiza-

tion," *Communications in Information Systems*, vol. 20, pp. 353 – 388, 2020, also available as arXiv:2003.14366.

[2] A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, and Y. LeCun, "The Loss Surfaces of Multilayer Networks," in *Proc. International Conference on Artificial Intelligence and Statistics*, San Diego, May 2015, pp. 192–204.

[3] K. Kawaguchi, "Deep learning without poor local minima," in *Advances in Neural Information Processing Systems*, pp. 586–594. 2016.

[4] R. Ge, F. Huang, C. Jin, and Y. Yuan, "Escaping from saddle points—online stochastic gradient for tensor decomposition," in *Proc. of Conference on Learning Theory*, Paris, France, 2015, pp. 797–842.

[5] H. Daneshmand, J. Kohler, A. Lucchi, and T. Hofmann, "Escaping saddles with stochastic gradients," in *Proc. International Conference on Machine Learning*, Jul 2018, pp. 1155–1164.

[6] C. Jin, P. Netrapalli, R. Ge, S. M. Kakade and M. I. Jordan, "Stochastic gradient descent escapes saddle points efficiently," *available as arXiv:1902.04811*, Feb. 2019.

[7] C. Fang, Z. Lin, and T. Zhang, "Sharp analysis for nonconvex SGD escaping from saddle points," in *Proc. Conference on Learning Theory*, Jun 2019, pp. 1192–1234.

[8] S. Vlaski and A. H. Sayed, "Second-order guarantees of stochastic gradient descent in non-convex optimization," *available as arXiv:1908.07023*, August 2019.

[9] S. Vlaski and A. H. Sayed, "Distributed learning in non-convex environments – Part I: Agreement at a Linear rate," *available as arXiv:1907.01848*, 2021.

[10] S. Vlaski and A. H. Sayed, "Distributed learning in non-convex environments – Part II: Polynomial escape from saddle-points," to appear in *IEEE Transactions on Signal Processing, also available as arXiv:1907.01849*, July 2019.

[11] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Athena Scientific, 1997.

[12] A. Agarwal and J. C. Duchi, "Distributed delayed stochastic optimization," in *Advances in Neural Information Processing Systems*, 2011, vol. 24, pp. 873–881.

[13] S. U. Stich, "Local SGD converges fast and communicates little," in *Proc. Conference on Learning Representations*, New Orleans, LA, USA, May 2019.

[14] A. Khaled, K. Mishchenko, and P. Richtárik, "First analysis of local GD on heterogeneous data," *available as arXiv:1909.04715*, 2019.

[15] D. P. Bertsekas, "A new class of incremental gradient methods for least squares problems," *SIAM J. Optim.*, vol. 7, no. 4, pp. 913–926, April 1997.

[16] A. Nedić and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Automatic Control*, vol. 54, no. 1, pp. 48–61, Jan 2009.

[17] A. H. Sayed, "Adaptation, learning, and optimization over networks," *Foundations and Trends in Machine Learning*, vol. 7, no. 4-5, pp. 311–801, July 2014.

[18] A. H. Sayed, "Adaptive networks," *Proceedings of the IEEE*, vol. 102, no. 4, pp. 460–497, April 2014.

[19] J. C. Duchi, A. Agarwal, and M. J. Wainwright, "Dual averaging for distributed optimization: Convergence analysis and network scaling," *IEEE Transactions on Automatic Control*, vol. 57, no. 3, pp. 592–606, March 2012.

[20] H. B. McMahan, E. Moore, D. Ramage, S. Hampson, and B. Agüera y Arcas, "Communication-efficient learning of deep networks from decentralized data," *Proc. International Conference on Artificial Intelligence and Statistics*, vol. 54, pp. 1273–1282, April 2017.

[21] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smithy, "FedDANE: A federated newton-type method," in *Proc. Asilomar Conference on Signals, Systems, and Computers*, 2019, pp. 1227–1231.

[22] T. Li, A. K. Sahu, M. Zaheer, M. Sanjabi, A. Talwalkar, and V. Smith, "Federated optimization in heterogeneous networks," in *Proceedings of Machine Learning and Systems*, I. Dhillon, D. Papailiopoulos, and V. Sze, Eds., 2020, vol. 2, pp. 429–450.

[23] L. Liu, J. Zhang, S. H. Song, and K. B. Letaief, "Client-edge-cloud hierarchical federated learning," in *Proc. IEEE ICC*, 2020, pp. 1–6.

[24] E. Rizk, S. Vlaski, and A. H. Sayed, "Dynamic federated learning," in *Proc. IEEE SPAWC*, 2020, pp. 1–5.

[25] V. Smith, C.-K. Chiang, M. Sanjabi, and A. S Talwalkar, "Federated multi-task learning," in *Advances in Neural Information Processing Systems*, 2017, vol. 30, pp. 4424–4434.

[26] R. Nassif, S. Vlaski, C. Richard, J. Chen, and A. H. Sayed, "Multitask learning over graphs: An approach for distributed, streaming machine learning," *IEEE Signal Processing Magazine*, vol. 37, no. 3, pp. 14–25, 2020.

[27] J. Wang and G. Joshi, "Cooperative SGD: A unified framework for the design and analysis of communication-efficient SGD algorithms," *available as arXiv:1808.07576*, 2018.

[28] F. Zhou and G. Cong, "On the convergence properties of a k-step averaging stochastic gradient descent algorithm for non-convex optimization," in *Proc. International Joint Conference on Artificial Intelligence*, July 2018, pp. 3219–3227.

[29] H. Yu, S. Yang, and S. Zhu, "Parallel restarted sgd with faster convergence and less communication: Demystifying why model averaging works for deep learning," *Proc. AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 5693–5700, Jul. 2019.

[30] S. Gelfand and S. Mitter, "Recursive stochastic algorithms for global optimization in $\mathbb{R}^d$," *SIAM Journal on Control and Optimization*, vol. 29, no. 5, pp. 999–1018, 1991.

[31] S. S. Du, C. Jin, J. D. Lee, M. I. Jordan, B. Póczos, and A. Singh, "Gradient descent can take exponential time to escape saddle points," in *Proc. International Conference on Neural Information Processing Systems*, 2017, pp. 1067–1077.

[32] A. Daneshmand, G. Scutari, and V. Kungurtsev, "Second-order guarantees of distributed gradient algorithms," *SIAM Journal on Optimization*, vol. 30, no. 4, pp. 3029–3068, 2020.

[33] B. Swenson, S. Kar, H. V. Poor and J. M. F. Moura, "Annealing for distributed global optimization," *available as arXiv:1903.07258*, March 2019.

[34] C. Dwork and A. Roth, "The algorithmic foundations of differential privacy," *Found. Trends Theor. Comput. Sci.*, vol. 9, no. 3–4, pp. 211–407, Aug. 2014.

[35] X. Zhao and A. H. Sayed, "Asynchronous adaptation and learning over networks – Part I: Modeling and stability analysis," *IEEE Transactions on Signal Processing*, vol. 63, no. 4, pp. 811–826, 2015.

A. PROOF OF LEMMA 1

We begin by establishing that $\boldsymbol{s}_i$ has conditional zero-mean:

$$\mathbb{E}\left\{\boldsymbol{s}_i \,|\, \boldsymbol{w}_{i-1}\right\}$$

$$\overset{(21)}{=} \frac{K}{L}\sum_{k=1}^{K}\frac{p_k}{E_k}\sum_{e=1}^{E_k}\mathbb{E}\left\{\mathbb{1}_{k,i}\widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1})\,|\,\boldsymbol{w}_{i-1}\right\} - \nabla J(\boldsymbol{w}_{i-1})$$

$$\overset{(a)}{=}\frac{K}{L}\sum_{k=1}^{K}\frac{p_k}{E_k}\sum_{e=1}^{E_k}\mathbb{E}\left\{\mathbb{1}_{k,i}\right\}\mathbb{E}\left\{\widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1})|\,\boldsymbol{w}_{i-1}\right\}-\nabla J(\boldsymbol{w}_{i-1})$$

$$\overset{(b)}{=}\frac{K}{L}\sum_{k=1}^{K}\frac{p_k}{E_k}\sum_{e=1}^{E_k}\frac{L}{K}\nabla J_k(\boldsymbol{w}_{i-1})-\nabla J(\boldsymbol{w}_{i-1})$$

$$=\sum_{k=1}^{K}p_k\nabla J_k(\boldsymbol{w}_{i-1})-\nabla J(\boldsymbol{w}_{i-1})=0 \tag{43}$$

where (a) follows because participation $\mathbb{1}_{k,i}$ is independent of $\boldsymbol{w}_{i-1}$ and the data available at time i, and hence $\widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1})$. Step (b) follows from $\mathbb{E}\left\{\mathbb{1}_{k,i}\right\}=\frac{L}{K}$ and (12). We now proceed to evaluate the aggregate gradient noise covariance. For brevity, we define:

$$\boldsymbol{g}_{k,i}\triangleq\frac{1}{E_k}\sum_{e=1}^{E_k}\widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1}) \tag{44}$$

Then:

$$\boldsymbol{s}_i\triangleq\frac{K}{L}\sum_{k=1}^{K}\mathbb{1}_{k,i}p_k\boldsymbol{g}_{k,i}-\nabla J(\boldsymbol{w}_{i-1}) \tag{45}$$

For the aggregate gradient noise covariance, we have:

$$\mathbb{E}\left\{\boldsymbol{s}_i\,\boldsymbol{s}_i^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$=\mathbb{E}\left\{\left(\frac{K}{L}\sum_{k=1}^{K}\mathbb{1}_{k,i}p_k\boldsymbol{g}_{k,i}-\nabla J(\boldsymbol{w}_{i-1})\right)\right.$$
$$\left.\times\left(\frac{K}{L}\sum_{k=1}^{K}\mathbb{1}_{k,i}p_k\boldsymbol{g}_{k,i}-\nabla J(\boldsymbol{w}_{i-1})\right)^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$\overset{(a)}{=}\mathbb{E}\left\{\left(\frac{K}{L}\sum_{k=1}^{K}\mathbb{1}_{k,i}p_k\boldsymbol{g}_{k,i}\right)\left(\frac{K}{L}\sum_{k=1}^{K}\mathbb{1}_{k,i}p_k\boldsymbol{g}_{k,i}\right)^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$
$$-\nabla J(\boldsymbol{w}_{i-1})\nabla J(\boldsymbol{w}_{i-1})^{\mathsf{T}}$$

$$=\mathbb{E}\left\{\left(\frac{K^2}{L^2}\sum_{k=1}^{K}\sum_{\ell=1}^{K}\mathbb{1}_{k,i}\mathbb{1}_{\ell,i}p_kp_\ell\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}\right)|\,\boldsymbol{w}_{i-1}\right\}$$
$$-\nabla J(\boldsymbol{w}_{i-1})\nabla J(\boldsymbol{w}_{i-1})^{\mathsf{T}} \tag{46}$$

where (a) follows after multiplying and simplifying cross-terms by noting that:

$$\mathbb{E}\left\{\frac{K}{L}\sum_{k=1}^{K}\mathbb{1}_{k,i}p_k\boldsymbol{g}_{k,i}|\,\boldsymbol{w}_{i-1}\right\}\overset{(43)}{=}\nabla J(\boldsymbol{w}_{i-1}) \tag{47}$$

The challenge in evaluating (46) lies in the fact that, while the approximations $\boldsymbol{g}_{k,i}$ and $\boldsymbol{g}_{\ell,i}$ are mutually independent by Assumption 1, the same does not hold for the participation indicators $\mathbb{1}_{k,i}$ and $\mathbb{1}_{\ell,i}$, since agents are sampled without replacement. We can nevertheless evaluate:

$$\mathbb{E}\left\{\left(\frac{K^2}{L^2}\sum_{k=1}^{K}\sum_{\ell=1}^{K}\mathbb{1}_{k,i}\mathbb{1}_{\ell,i}p_kp_\ell\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}\right)|\,\boldsymbol{w}_{i-1}\right\}$$

$$\overset{(a)}{=}\frac{K^2}{L^2}\sum_{k=1}^{K}\sum_{\ell=1}^{K}\mathbb{E}\left\{\mathbb{1}_{k,i}\mathbb{1}_{\ell,i}p_kp_\ell\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1},\mathbb{1}_{k,i}=1\right\}$$
$$\times\Pr\left\{\mathbb{1}_{k,i}=1\right\}$$

$$=\frac{L}{K}\frac{K^2}{L^2}\sum_{k=1}^{K}\sum_{\ell=1}^{K}\mathbb{E}\left\{\mathbb{1}_{\ell,i}p_kp_\ell\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1},\mathbb{1}_{k,i}=1\right\}$$

$$\overset{(b)}{=}\frac{K}{L}\sum_{k=1}^{K}\sum_{\ell=1}^{K}\mathbb{E}\left\{\mathbb{1}_{\ell,i}|\mathbb{1}_{k,i}=1\right\}p_kp_\ell\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$\overset{(c)}{=}\frac{K}{L}\sum_{k=1}^{K}\mathbb{E}\left\{\mathbb{1}_{k,i}|\mathbb{1}_{k,i}=1\right\}p_k^2\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{k,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$+\frac{K}{L}\sum_{k=1}^{K}\sum_{\ell\neq k}\mathbb{E}\left\{\mathbb{1}_{\ell,i}|\mathbb{1}_{k,i}=1\right\}p_kp_\ell\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$\overset{(d)}{=}\frac{K}{L}\sum_{k=1}^{K}p_k^2\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{k,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$+\frac{K}{L}\frac{L-1}{K-1}\sum_{k=1}^{K}\sum_{\ell\neq k}p_kp_\ell\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\} \tag{48}$$

where (a) follows from Bayes' theorem and (b) is due to the fact that $\mathbb{1}_{k,i}$ and $\boldsymbol{w}_{i-1}$ are independent, (c) separates cross-terms and (d) results from $\mathbb{E}\left\{\mathbb{1}_{\ell,i}|\mathbb{1}_{k,i}=1\right\}=\frac{L-1}{K-1}$. We return to (46):

$$\mathbb{E}\left\{\boldsymbol{s}_i\,\boldsymbol{s}_i^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$=\frac{K}{L}\sum_{k=1}^{K}p_k^2\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{k,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$-\left(\sum_{k=1}^{K}p_k\nabla J_k(\boldsymbol{w}_{i-1})\right)\left(\sum_{\ell=1}^{K}p_\ell\nabla J_\ell(\boldsymbol{w}_{i-1})\right)^{\mathsf{T}}$$

$$+\frac{K}{L}\frac{L-1}{K-1}\sum_{k=1}^{K}\sum_{\ell\neq k}p_kp_\ell\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$\overset{(a)}{=}\frac{K}{L}\sum_{k=1}^{K}p_k^2\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{k,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}-\sum_{k=1}^{K}p_k^2\nabla J_k(\boldsymbol{w}_{i-1})\nabla J_k(\boldsymbol{w}_{i-1})^{\mathsf{T}}$$

$$-\sum_{k=1}^{K}\sum_{\ell\neq k}p_kp_\ell\nabla J_k(\boldsymbol{w}_{i-1})\nabla J_\ell(\boldsymbol{w}_{i-1})^{\mathsf{T}}$$

$$+\frac{K}{L}\frac{L-1}{K-1}\sum_{k=1}^{K}\sum_{\ell\neq k}p_kp_\ell\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$\overset{(b)}{=}\frac{K}{L}\sum_{k=1}^{K}p_k^2\,\mathbb{E}\left\{\left(\boldsymbol{g}_{k,i}-\nabla J_k(\boldsymbol{w}_{i-1})\right)\right.$$
$$\left.\times\left(\boldsymbol{g}_{k,i}-\nabla J_k(\boldsymbol{w}_{i-1})\right)^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$+\frac{K-L}{L}\sum_{k=1}^{K}p_k^2\nabla J_k(\boldsymbol{w}_{i-1})\nabla J_k(\boldsymbol{w}_{i-1})^{\mathsf{T}}$$

$$+\left(\frac{K}{L}\frac{L-1}{K-1}-1\right)\sum_{k=1}^{K}\sum_{\ell\neq k}p_kp_\ell\nabla J_k(\boldsymbol{w}_{i-1})\nabla J_\ell(\boldsymbol{w}_{i-1})^{\mathsf{T}}$$

$$-\frac{K}{L}\frac{L-1}{K-1}\sum_{k=1}^{K}\sum_{\ell\neq k}p_kp_\ell\nabla J_k(\boldsymbol{w}_{i-1})\nabla J_\ell(\boldsymbol{w}_{i-1})^{\mathsf{T}}$$

$$+\frac{K}{L}\frac{L-1}{K-1}\sum_{k=1}^{K}\sum_{\ell\neq k}p_kp_\ell\,\mathbb{E}\left\{\boldsymbol{g}_{k,i}\boldsymbol{g}_{\ell,i}^{\mathsf{T}}|\,\boldsymbol{w}_{i-1}\right\}$$

$$\overset{(c)}{=} \frac{K}{L} \sum_{k=1}^{K} \frac{p_k^2}{E_k} R_{s,k}(\boldsymbol{w}_{i-1})$$

$$+ \frac{K-L}{L} \sum_{k=1}^{K} p_k^2 \nabla J_k(\boldsymbol{w}_{i-1}) \nabla J_k(\boldsymbol{w}_{i-1})^\mathsf{T}$$

$$- \frac{K-L}{L(K-1)} \sum_{k=1}^{K} \sum_{\ell \neq k} p_k p_\ell \nabla J_k(\boldsymbol{w}_{i-1}) \nabla J_\ell(\boldsymbol{w}_{i-1})^\mathsf{T}$$

$$\overset{(d)}{=} \frac{K}{L} \sum_{k=1}^{K} \frac{p_k^2}{E_k} R_{s,k}(\boldsymbol{w}_{i-1})$$

$$+ \frac{K-L}{L} \sum_{k=1}^{K} p_k^2 \nabla J_k(\boldsymbol{w}_{i-1}) \nabla J_k(\boldsymbol{w}_{i-1})^\mathsf{T}$$

$$+ \frac{K-L}{L(K-1)} \sum_{k=1}^{K} p_k^2 \nabla J_k(\boldsymbol{w}_{i-1}) \nabla J_k(\boldsymbol{w}_{i-1})^\mathsf{T}$$

$$- \frac{K-L}{L(K-1)} \nabla J(\boldsymbol{w}_{i-1}) \nabla J(\boldsymbol{w}_{i-1})^\mathsf{T}$$

$$= \frac{K}{L} \sum_{k=1}^{K} \frac{p_k^2}{E_k} R_{s,k}(\boldsymbol{w}_{i-1})$$

$$+ \frac{K}{L} \frac{K-L}{K-1} \sum_{k=1}^{K} p_k^2 \nabla J_k(\boldsymbol{w}_{i-1}) \nabla J_k(\boldsymbol{w}_{i-1})^\mathsf{T}$$

$$- \frac{K-L}{L(K-1)} \nabla J(\boldsymbol{w}_{i-1}) \nabla J(\boldsymbol{w}_{i-1})^\mathsf{T}$$

$$\overset{(e)}{=} \frac{K}{L} \sum_{k=1}^{K} \frac{p_k^2}{E_k} R_{s,k}(\boldsymbol{w}_{i-1})$$

$$+ \frac{K}{L} \frac{K-L}{K-1} \sum_{k=1}^{K} \left(\left(p_k \nabla J_k(\boldsymbol{w}_{i-1}) - \frac{1}{K} \nabla J(\boldsymbol{w}_{i-1}) \right) \right.$$

$$\left. \times \left(p_k \nabla J_k(\boldsymbol{w}_{i-1}) - \frac{1}{K} \nabla J(\boldsymbol{w}_{i-1}) \right)^\mathsf{T} \right) \quad (49)$$

where (a) multiplies $\left(\sum_{k=1}^{K} p_k \nabla J_k(\boldsymbol{w}_{i-1}) \right) \left(\sum_{\ell=1}^{K} p_\ell \nabla J_\ell(\boldsymbol{w}_{i-1}) \right)^\mathsf{T}$ and separates cross-terms, (b) combines terms using the fact that $\mathbb{E}\{\boldsymbol{g}_{k,i} | \boldsymbol{w}_{i-1}\} = \nabla J_k(\boldsymbol{w}_{i-1})$, (c) follows from (44), (15) and the fact the $\boldsymbol{g}_{k,i}$ are mutually independent. Step (d) completes the square to obtain $\nabla J(\boldsymbol{w}_{i-1}) \nabla J(\boldsymbol{w}_{i-1})^\mathsf{T}$ and (e) can be verified by multiplying out the result. For the fourth-order moment, we have following the argument in [1, Example 7]:

$$\mathbb{E}\left\{ \| \boldsymbol{s}_i \|^4 | \boldsymbol{w}_{i-1} \right\}$$

$$\overset{(21)}{=} \mathbb{E}\left\{ \left\| \frac{K}{L} \sum_{k=1}^{K} \frac{p_k}{E_k} \sum_{e=1}^{E_k} \mathbb{1}_{k,i} \widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1}) - \nabla J(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$= \mathbb{E}\left\{ \left\| \frac{K}{L} \sum_{k=1}^{K} \frac{p_k}{E_k} \sum_{e=1}^{E_k} \mathbb{1}_{k,i} \widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1}) \right.\right.$$

$$\left.\left. - \sum_{k=1}^{K} p_k \nabla J_k(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$\overset{(a)}{\leq} \sum_{k=1}^{K} p_k \mathbb{E}\left\{ \left\| \frac{K}{L} \frac{1}{E_k} \sum_{e=1}^{E_k} \mathbb{1}_{k,i} \widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1}) \right.\right.$$

$$\left.\left. - \nabla J_k(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\} \quad (50)$$

where (a) follows by Jensen's inequality. We proceed with the individual terms of the sum:

$$\mathbb{E}\left\{ \left\| \frac{K}{L} \frac{1}{E_k} \sum_{e=1}^{E_k} \mathbb{1}_{k,i} \widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1}) - \nabla J_k(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$= \mathbb{E}\left\{ \left\| \frac{K}{L} \frac{1}{E_k} \sum_{e=1}^{E_k} \mathbb{1}_{k,i} \left(\widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1}) - \nabla J_k(\boldsymbol{w}_{i-1}) \right) \right.\right.$$

$$\left.\left. + \left(\frac{K}{L} \mathbb{1}_{k,i} - 1 \right) \nabla J_k(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$\overset{(11)}{=} \mathbb{E}\left\{ \left\| \frac{K}{L} \frac{1}{E_k} \sum_{e=1}^{E_k} \mathbb{1}_{k,i} \boldsymbol{s}_{k,e}(\boldsymbol{w}_{i-1}) \right.\right.$$

$$\left.\left. + \left(\frac{K}{L} \mathbb{1}_{k,i} - 1 \right) \nabla J_k(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$\overset{(a)}{\leq} 8 \mathbb{E}\left\{ \left\| \frac{K}{L} \frac{1}{E_k} \sum_{e=1}^{E_k} \mathbb{1}_{k,i} \boldsymbol{s}_{k,e}(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$+ 8 \mathbb{E}\left\{ \left\| \left(\frac{K}{L} \mathbb{1}_{k,i} - 1 \right) \nabla J_k(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$\overset{(b)}{=} 8 \frac{K^4}{L^4} \mathbb{E}\left\{ \left\| \frac{1}{E_k} \sum_{e=1}^{E_k} \mathbb{1}_{k,i} \boldsymbol{s}_{k,e}(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$+ 8 \mathbb{E}\left\{ \left(\frac{K}{L} \mathbb{1}_{k,i} - 1 \right)^4 \right\} \| \nabla J_k(\boldsymbol{w}_{i-1}) \|^4$$

$$\overset{(c)}{=} 8 \frac{K^4}{L^4} \Pr\{\mathbb{1}_{k,i} = 1\} \mathbb{E}\left\{ \left\| \frac{1}{E_k} \sum_{e=1}^{E_k} \boldsymbol{s}_{k,e}(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$+ 8 \Pr\{\mathbb{1}_{k,i} = 1\} \mathbb{E}\left\{ \left(\frac{K}{L} - 1 \right)^4 \right\} \| \nabla J_k(\boldsymbol{w}_{i-1}) \|^4$$

$$+ 8 \Pr\{\mathbb{1}_{k,i} = 0\} \mathbb{E}\left\{ (-1)^4 \right\} \| \nabla J_k(\boldsymbol{w}_{i-1}) \|^4$$

$$\overset{(d)}{=} 8 \frac{K^4}{L^4} \frac{L}{K} \mathbb{E}\left\{ \left\| \frac{1}{E_k} \sum_{e=1}^{E_k} \boldsymbol{s}_{k,e}(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$+ 8 \left(\frac{L}{K} \left(\frac{K-L}{L} \right)^4 + \frac{K-L}{K} \right) \| \nabla J_k(\boldsymbol{w}_{i-1}) \|^4$$

$$\overset{(e)}{\leq} 24 \frac{K^3}{L^3} \frac{1}{E_k^2} \left(\beta_k^4 \| \nabla J_k(\boldsymbol{w}_{i-1}) \|^4 + \sigma_k^4 \right)$$

$$+ 8 \left(\frac{L}{K} \left(\frac{K-L}{L} \right)^4 + \frac{K-L}{K} \right) \| \nabla J_k(\boldsymbol{w}_{i-1}) \|^4$$

$$= \left(24 \frac{K^3}{L^3} \frac{\beta_k^4}{E_k^2} + 8 \frac{L}{K} \left(\frac{K-L}{L} \right)^4 + 8 \frac{K-L}{K} \right) \| \nabla J_k(\boldsymbol{w}_{i-1}) \|^4$$

$$+ 24 \frac{K^3}{L^3} \frac{\sigma_k^4}{E_k^2}$$

$$= \left(24 \frac{K^3}{L^3} \frac{\beta_k^4}{E_k^2} + 8 \frac{L}{K} \left(\frac{K-L}{L} \right)^4 + 8 \frac{K-L}{K} \right)$$

$$\times \|\nabla J(\boldsymbol{w}_{i-1}) + \nabla J_k(\boldsymbol{w}_{i-1}) - \nabla J(\boldsymbol{w}_{i-1})\|^4$$
$$+ 24 \frac{K^3}{L^3} \frac{\sigma_k^4}{E_k^2}$$
$$\overset{(f)}{\leq} \left(24 \frac{K^3}{L^3} \frac{\beta_k^4}{E_k^2} + 8 \frac{L}{K} \left(\frac{K-L}{L} \right)^4 + 8 \frac{K-L}{K} \right)$$
$$\times \left(8 \|\nabla J(\boldsymbol{w}_{i-1})\|^4 + 8 \|\nabla J_k(\boldsymbol{w}_{i-1}) - \nabla J(\boldsymbol{w}_{i-1})\|^4 \right)$$
$$+ 24 \frac{K^3}{L^3} \frac{\sigma_k^4}{E_k^2}$$
$$\overset{(25)}{\leq} \left(24 \frac{K^3}{L^3} \frac{\beta_k^4}{E_k^2} + 8 \frac{L}{K} \left(\frac{K-L}{L} \right)^4 + 8 \frac{K-L}{K} \right)$$
$$\times \left(8 \|\nabla J(\boldsymbol{w}_{i-1})\|^4 + 8G^4 \right) + 24 \frac{K^3}{L^3} \frac{\sigma_k^4}{E_k^2}$$
$$= \left(192 \frac{K^3}{L^3} \frac{\beta_k^4}{E_k^2} + 64 \frac{L}{K} \left(\frac{K-L}{L} \right)^4 + 64 \frac{K-L}{K} \right) \|\nabla J(\boldsymbol{w}_{i-1})\|^4$$
$$+ \left(192 \frac{K^3}{L^3} \frac{\beta_k^4}{E_k^2} + 64 \frac{L}{K} \left(\frac{K-L}{L} \right)^4 + 64 \frac{K-L}{K} \right) G^4$$
$$+ 24 \frac{K^3}{L^3} \frac{\sigma_k^4}{E_k^2} \tag{51}$$

where (a) and (f) follow from Jensen's inequality, (b) uses the fact that $\mathbb{1}_{k,i}$ is independent of $\widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1})$, (c) applies Bayes' theorem and (d) uses $\Pr\{\mathbb{1}_{k,i} = 1\} = \mathbb{E}\{\mathbb{1}_{k,i}\} = \frac{L}{K}$. Step (e) follows from (13) and:

$$\mathbb{E}\left\{ \left\| \frac{1}{E_k} \sum_{e=1}^{E_k} \boldsymbol{s}_{k,e}(\boldsymbol{w}_{i-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$\leq \frac{3 - \frac{2}{E_k}}{E_k^2} \left(\beta_k^4 \|\nabla J_k(\boldsymbol{w}_{i-1})\|^4 + \sigma_k^4 \right), \tag{52}$$

which can be verified by induction over E_k [1]. Next, we bound the fourth-moment of the term $\boldsymbol{d}_i$, arising from the fact that agents take $E_k > 1$ local gradient steps before returning the updated estimate to the parameter server. We introduce $\boldsymbol{d}_{k,e-1} \triangleq \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,e-1}) - \widehat{\nabla J}_k^e(\boldsymbol{w}_{i-1})$ for brevity. Then, we have:

$$\mathbb{E}\left\{ \|\boldsymbol{d}_i\|^4 \big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(22)}{=} \mathbb{E}\left\{ \left\| \frac{K}{L} \sum_{k=1}^{K} \mathbb{1}_{k,i} \frac{p_k}{E_k} \sum_{e=1}^{E_k} \boldsymbol{d}_{k,e-1} \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(a)}{\leq} \sum_{k=1}^{K} p_k \mathbb{E}\left\{ \left\| \frac{K}{L} \frac{\mathbb{1}_{k,i}}{E_k} \sum_{e=1}^{E_k} \boldsymbol{d}_{k,e-1} \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$= \sum_{k=1}^{K} p_k \frac{K^4}{L^4} \mathbb{E}\left\{ \left\| \frac{\mathbb{1}_{k,i}}{E_k} \sum_{e=1}^{E_k} \boldsymbol{d}_{k,e-1} \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(b)}{=} \sum_{k=1}^{K} p_k \frac{K^4}{L^4} \Pr\{\mathbb{1}_{k,i} = 1\} \mathbb{E}\left\{ \left\| \frac{1}{E_k} \sum_{e=1}^{E_k} \boldsymbol{d}_{k,e-1} \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(c)}{=} \sum_{k=1}^{K} p_k \frac{K^3}{L^3} \mathbb{E}\left\{ \left\| \frac{1}{E_k} \sum_{e=1}^{E_k} \boldsymbol{d}_{k,e-1} \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$

$$\overset{(d)}{\leq} \sum_{k=1}^{K} p_k \frac{K^3}{L^3} \frac{1}{E_k} \sum_{e=1}^{E_k} \mathbb{E}\left\{ \|\boldsymbol{d}_{k,e-1}\|^4 \big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(27)}{\leq} \sum_{k=1}^{K} p_k \frac{K^3}{L^3} \frac{\hat{\delta}^4}{E_k} \sum_{e=1}^{E_k} \mathbb{E}\left\{ \|\boldsymbol{w}_{k,e-1} - \boldsymbol{w}_{i-1}\|^4 \big| \boldsymbol{w}_{i-1} \right\} \tag{53}$$

where (a) and (d) follow from Jensen's inequality, (b) applies a Bayes' decomposition and (c) follows form $\Pr\{\mathbb{1}_{k,i} = 1\} = \frac{L}{K}$. We now bound the deviation of estimates over one epoch. For $e = 1$, we have $\boldsymbol{w}_{k,e-1} = \boldsymbol{w}_{k,0} = \boldsymbol{w}_{i-1}$ and hence $\mathbb{E}\left\{ \|\boldsymbol{w}_{k,e-1} - \boldsymbol{w}_{i-1}\|^4 \big| \boldsymbol{w}_{i-1} \right\} = 0$. For $e \geq 2$, iterating (5), we find:

$$\mathbb{E}\left\{ \|\boldsymbol{w}_{k,e-1} - \boldsymbol{w}_{i-1}\|^4 \big| \boldsymbol{w}_{i-1} \right\}$$
$$= \mathbb{E}\left\{ \left\| \mu K \mathbb{1}_{k,i} \frac{p_k}{E_k} \sum_{j=1}^{e-1} \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,j-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$= \mu^4 p_k^4 K^4 \mathbb{E}\left\{ \left\| \mathbb{1}_{k,i} \frac{1}{E_k} \sum_{j=1}^{e-1} \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,j-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(a)}{=} \mu^4 p_k^4 K^4 \Pr\{\mathbb{1}_{k,i} = 1\} \mathbb{E}\left\{ \left\| \frac{1}{E_k} \sum_{j=1}^{e-1} \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,j-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(b)}{=} \mu^4 p_k^4 K^4 \frac{L}{K} \mathbb{E}\left\{ \left\| \frac{1}{E_k} \sum_{j=1}^{e-1} \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,j-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$= \mu^4 p_k^4 K^4 \frac{L}{K} \mathbb{E}\left\{ \left\| \frac{e-1}{E_k} \frac{1}{e-1} \sum_{j=1}^{e-1} \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,j-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$= \mu^4 p_k^4 K^4 \frac{L}{K} \frac{(e-1)^4}{E_k^4} \mathbb{E}\left\{ \left\| \frac{1}{e-1} \sum_{j=1}^{e-1} \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,j-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(c)}{\leq} \mu^4 p_k^4 K^4 \frac{L}{K} \frac{(e-1)^3}{E_k^4} \sum_{j=1}^{e-1} \mathbb{E}\left\{ \left\| \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,j-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(d)}{\leq} \mu^4 p_k^4 K^4 \frac{L}{K} \frac{1}{E_k} \sum_{j=1}^{e-1} \mathbb{E}\left\{ \left\| \widehat{\nabla J}_k^e(\boldsymbol{w}_{k,j-1}) \right\|^4 \Big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(e)}{\leq} \mu^4 p_k^4 K^4 \frac{L}{K} \frac{1}{E_k} \sum_{j=1}^{e-1} 8 \mathbb{E}\left\{ \|\nabla J_k(\boldsymbol{w}_{k,j-1})\|^4 \big| \boldsymbol{w}_{i-1} \right\}$$
$$+ \mu^4 p_k^4 K^4 \frac{L}{K} \frac{1}{E_k} \sum_{j=1}^{e-1} 8 \mathbb{E}\left\{ \|\boldsymbol{s}_{k,j-1}(\boldsymbol{w}_{k,j-1})\|^4 \big| \boldsymbol{w}_{i-1} \right\}$$
$$\overset{(f)}{\leq} \mu^4 p_k^4 K^4 \frac{L}{K} 8U^4 + \mu^4 p_k^4 K^4 \frac{L}{K} 8 \left(\beta_k^4 U^4 + \sigma_k^4 \right)$$
$$= \mu^4 p_k^4 K^4 \frac{L}{K} 8 \left(U^4 + \beta_k^4 U^4 + \sigma_k^4 \right) \tag{54}$$

where (a) and (b) follow from Bayes' theorem and $\Pr\{\mathbb{1}_{k,i} = 1\} = \frac{L}{K}$, (c) and (e) follows from (11) and Jensen's inequality, (d) follows from $e - 1 \leq E_k$, and (f) follows from (13) and (26) and the fact that $e - 1 \leq E_k$. Returning to (53), we have:

$$\mathbb{E}\left\{ \|\boldsymbol{d}_i\|^4 \big| \boldsymbol{w}_{i-1} \right\} \overset{(53)}{\leq} \mu^4 \sum_{k=1}^{K} p_k^5 \frac{K^6}{L^2} \hat{\delta}^4 8 \left(U^4 + \beta_k^4 U^4 + \sigma_k^4 \right) \tag{55}$$