Understanding Clipping for Federated Learning: Convergence and Client-Level Differential Privacy

Xinwei Zhang [1] **Xiangyi Chen** [1] **Mingyi Hong** [1] **Zhiwei Steven Wu** [2] **Jinfeng Yi** [3]

Abstract

Providing privacy protection has been one of the primary motivations of Federated Learning (FL). Recently, there has been a line of work on incorporating the formal privacy notion of differential privacy with FL. To guarantee the client-level differential privacy in FL algorithms, the clients' transmitted model updates have to be *clipped* before adding privacy noise. Such clipping operation is substantially different from its counterpart of gradient clipping in the centralized differentially private SGD and has not been well-understood. In this paper, we first empirically demonstrate that the clipped FedAvg can perform surprisingly well even with substantial data heterogeneity when training neural networks, which is partly because the clients' updates become *similar* for several popular deep architectures. Based on this key observation, we provide the convergence analysis of a differential private (DP) FedAvg algorithm and highlight the relationship between clipping bias and the distribution of the clients' updates. To the best of our knowledge, this is the first work that rigorously investigates theoretical and empirical issues regarding the clipping operation in FL algorithms.

1. Introduction

First proposed by Konečnỳ et al. (2016), *Federated Learning* (FL) is a distributed learning framework that aims to reduce communication complexity and to provide privacy protection during training. The popular FedAvg algorithm (Konečnỳ et al., 2016) has been proposed to reduce the communication cost by using periodic averaging and client sampling. There have been many extensions of this algorithm, mostly by modifying the local update directions (Karimireddy et al., 2020; Zhang et al., 2020; Liang et al., 2019). Even though FL algorithms have the goal of privacy protection, recent works have shown that they are vulnerable to inference attacks and leak local information during training (Zhao et al., 2020; Zhu & Han, 2020; Wei et al., 2020b). As a result, striking a balance between *formal* privacy guarantees and desirable optimization performance remains one of the fundamental challenges in FL.

Recently, various FL algorithms (Geyer et al., 2017; Truex et al., 2020; 2019; Wang et al., 2020b; Triastcyn & Faltings, 2019) have been proposed to provide the formal guarantees of *differential privacy* (DP) (Dwork et al., 2006). In these algorithms, the clients perform multiple local updates between two communication steps, and then perturbation mechanisms are applied to aggregate updates across individual clients. In order for the perturbation mechanism to have formal privacy guarantees, each client's model update needs to have a bounded norm, which is ensured by applying a clipping operation that shrinks individual model updates when their norm exceeds a given threshold. While there has been prior work that studies the clipping effects on stochastic gradients (Bassily et al., 2014; Chen et al., 2020; Song et al., 2021) in the differentially private SGD (Abadi et al., 2016), there has not been any work on providing understanding how clipping the model updates affect the optimization performance of FL subject to DP. Our work provides the first in-depth study on such clipping effects.

Contributions. In this work, we will conduct rigorous theoretical analysis and provide extensive empirical evidence to understand how to best protect client-level DP for FL algorithms. Specifically, we make the following contributions:

1) We analyze the existing model and difference clipping strategies for clipping-enabled FedAvg and prove that difference clipping outperforms model clipping. Our result provides theoretical insight into designing FL algorithms with clipping operation.

[1]Department of Electrical and Computer Engineering, University of Minnesota, MN, United States [2]School of Computer Science, Carnegie Mellon University, PA, United States [3]JD.com, Inc., Shanghai, China. Correspondence to: Xinwei Zhang <zhan6234@umn.edu>, Mingyi Hong <mhong@umn.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

2) We empirically show that the performance of the clipping-enabled FedAvg depends on the structure of the neural network being used – when the structure of the network induces *concentrated* clients' updates, the performance drop becomes negligible.

3) We provide the convergence analysis of the clipping-enabled FedAvg algorithm and highlight the relationship between clipping bias and the distribution of the clients' updates. Our result leads to a natural guarantee of client-level DP for FedAvg.

To the best of our knowledge, this is the first work that rigorously investigates theoretical and empirical issues regarding the clipping operation in FL algorithms.

1.1. Preliminaries & Related Work

Federated learning typically considers the following optimization problem:

$$\min_{\mathbf{x}} \left[f(\mathbf{x}) \triangleq \sum_{i=1}^{N} f_i(\mathbf{x}) \right], \text{ where } f_i(\mathbf{x}) = \mathbb{E}_{\xi \sim \mathcal{D}_i} F(\mathbf{x}; \xi), \tag{1}$$

where N is the number of participating clients; the i^{th} client optimizes a local model f_i, which is the expectation of a loss function $F(\mathbf{x}; \xi)$, where $\mathbf{x}_i$ denotes the model parameters and ξ denotes the data sample, and the expectation is taken over local data distribution $\mathcal{D}_i$. At each communication round t, the server samples a subset of clients $\mathcal{P}_t$ and broadcasts the global model parameters $\mathbf{x}^t$. The sampled clients perform Q steps of SGD updates and compute the total update differences $\Delta \mathbf{x}_i^t$'s, and then the server aggregates the update differences to update the global model. In Algorithm 1, we present a slightly generalized FedAvg algorithm from Karimireddy et al. (2020); Yang et al. (2021), in which the server uses a stepsize η_g to perform its update. When $\eta_g = 1$, the algorithm becomes the same as the original FedAvg.

Algorithm 1 FedAvg Algorithm

Initialize: $\mathbf{x}_i^0 \triangleq \mathbf{x}^0, i = 1, \ldots, N$
for $t = 0, \ldots, T-1$ *(stage)* **do**
  for $i \in \mathcal{P}_t \subseteq [N]$ in parallel **do**
    Update agents' $\mathbf{x}_i^{t,0} = \mathbf{x}^t$
    for $q = 0, \ldots, Q-1$ *(iteration)* **do**
      Compute stochastic gradient $g_i^{t,q}$ with $\mathbb{E}[g_i^{t,q}] = \nabla f_i(x_i^{t,q})$
      Local update: $\mathbf{x}_i^{t,q+1} = \mathbf{x}_i^{t,q} - \eta_l g_i^{t,q}$
    end for
  end for
  Global averaging: $\Delta \mathbf{x}_i^t = \mathbf{x}_i^{t,Q} - \mathbf{x}^t, \quad \mathbf{x}^{t+1} = \mathbf{x}^t + \eta_g \frac{1}{|\mathcal{P}_t|} \sum_{i \in \mathcal{P}_t} \Delta \mathbf{x}_i^t$
end for

In this work, we study FL subject to the rigorous privacy guarantees of *Differential Privacy* (DP) (Dwork et al., 2006), whose formal definition is given below.

Definition 1.1. (Dwork et al., 2006) An algorithm $\mathcal{M}$ is (ϵ, δ)-differentially private if

$$P(\mathcal{M}(\mathcal{D}) \in \mathcal{S}) \leq e^{\epsilon} P(\mathcal{M}(\mathcal{D}') \in \mathcal{S}) + \delta, \tag{2}$$

where $\mathcal{D}$ and $\mathcal{D}'$ are neighboring datasets, $\mathcal{S}$ is an arbitrary subset of outputs of $\mathcal{M}$.

The common mechanism used to protect DP in centralized training is straightforward: 1) clip the stochastic gradient with the so-called clipping operation (3); 2) add a random perturbation $\mathbf{z} \sim \mathcal{N}(0, \sigma^2 I)$ to the clipped quantity (Abadi et al., 2016; McMahan et al., 2017; Andrew et al., 2019; Pichapati et al., 2019). The clipping operation is the key step to guarantee DP as the noise level σ^2 is determined by the clipping threshold c (Dwork & Roth, 2014):

$$\text{clip}(g^t, c) = g^t \cdot \min\left\{1, \frac{c}{\|g^t\|}\right\}. \tag{3}$$

However, DP is more complex in FL than that in centralized training. Two key factors distinguishing FL from existing DP machine learning framework are:

- *Data distribution*: unlike centralized training, in FL the data are naturally distributed on the clients, and the clients can potentially have very different data distributions. In the centralized setting, the recent work (Chen et al., 2020) has shown that the distribution of the samples affects the performance of the DP-SGD, but how heterogeneous data distribution affects the design and analysis of FL algorithm that protects DP is unclear.
- *Local updates*: as described in Algorithm 1, the clients will perform multiple local update steps before sending the model to the server, and it is well-known that when $Q > 1$, the data heterogeneity will cause performance degradation in FedAvg even without clipping and perturbation (Khaled et al., 2019). Although there are multiple alternatives of how the DP mechanism can be applied to FL algorithms, none of those mechanisms has a rigorous theoretical guarantee, and it is not clear how to properly balance the optimization performance and privacy guarantees.

These two factors result in different *definitions* and *clipping operations* in FL.

DP definitions in FL: Based on the distribution pattern of the client and local datasets, two DP definitions corresponding to the neighboring datasets in Definition 1.1, are commonly considered in FL algorithm design:

- Sample-level differential privacy (SL-DP): SL-DP directly follows the centralized DP and protects each local sample so that the server could not identify one sample from the

union of all local datasets, i.e., $\mathcal{D} = \bigcup_{i=1}^{N} \mathcal{D}_i$, and $\mathcal{D}, \mathcal{D}'$ differ by one sample ξ. SL-DP fits in the cross-silo FL scenario that has a relatively small number of clients, each with a large dataset. E.g., SL-DP is used in medical image classification application to protect patients' personal information (Choudhury et al., 2019). However, in the Google Keyboard application (Hard et al., 2018) where each client is an application user, SL-DP that only protects one sample (i.e., an input record) will not be sufficient to protect the user's personal information.

- Client-level differential privacy (CL-DP): CL-DP has a stricter privacy guarantee compared with SL-DP. It requires that the server cannot identify the participation of one client by observing the output of the local updates, i.e., $\mathcal{D} = \{\mathcal{D}_i\}_{i=1}^{N}$, and $\mathcal{D}, \mathcal{D}'$ differ by one dataset $\mathcal{D}_i$. CL-DP is suitable for the cross-device FL scenario such as the Google Keyboard application, which has a large number of distributed clients.

Clipping operation in FL: Based on different DP requirements and the algorithm structures, a number of FL algorithms have been proposed which protect DP to some extent.

To protect SL-DP, Truex et al. (2019) proposes to clip and inject noise to every local update. That is, some Gaussian noise is added to the stochastic gradients $g_i^{t,q}$ given in Algorithm 1. However, as intermediate updates are kept local and private, the clipping and perturbation to the local steps appear to be unnecessary, and such operations result in significant performance degradation. Moreover, it is not clear how such kind of operation impact other aspects of the algorithm performance (such as algorithm convergence, quality of solutions, etc.)

To protect CL-DP, Wei et al. (2020a) proposes to clip the local models to be transmitted directly. Similarly, Truex et al. (2020) assumes that the model parameters are upper and lower bounded by some constant and directly apply perturbations to the local models. However, this scheme also significantly reduces the training and test accuracy empirically and has no theoretical convergence guarantee. Recently, Geyer et al. (2017); McMahan et al. (2017) propose to clip the difference between the input model and the output models of the FedAvg algorithm. In particular, one can replace the update directions $\Delta \mathbf{x}_i^t$'s of line 8 in Algorithm 1 by their clipped versions as expressed below:

$$\begin{aligned} \text{clip}(\Delta \mathbf{x}_i^t, c) &= \Delta \mathbf{x}_i^t \cdot \min\left\{1, \frac{c}{\|\Delta \mathbf{x}_i^t\|}\right\}, \\ \mathbf{x}^{t+1} &= \mathbf{x}^t + \eta_g \frac{1}{|\mathcal{P}_t|} \sum_{i \in \mathcal{P}_t} \text{clip}(\Delta \mathbf{x}_i^t, c). \end{aligned} \tag{4}$$

It is shown that such a scheme has better numerical performance than model clipping, but no convergence proof for the algorithm is given. Reference Triastcyn & Faltings (2019) also clips the update difference and proposed Bayesian DP to measure the privacy loss and only demonstrates the numerical performance of the proposed algorithm. D2P-Fed (Wang et al., 2020b) follows the same clipping strategy and further apply compression and quantization during communication to improve communication efficiency while having DP guarantee, but its convergence guarantee only applies to the non-clipping version.

In summary, despite extensive recent research about DP-enabled FL, there are still a number of technical challenges and open research questions in this area. First, it is not clear how various kinds of clipping operations can affect the performance of FL algorithms. Second, it is not clear how to add noise to balance the convergence of FL algorithms and its CL-DP guarantee.

2. Clipping Issues in FL

As discussed above, clipping is a key operation in providing DP guarantee for FL algorithms. Therefore, to design algorithms that protect DP in FL, the first step is to understand how clipping affects the convergence performance of a FL algorithm. Towards this end, we start with analyzing two common clipping strategies, and identify their theoretical properties. Then we provide a series of empirical studies to demonstrate how system parameters such as training models, datasets and data distributions can affect the performance of clipping-enabled FedAvg algorithm. These empirical studies will be combined with our theoretical analysis in the next section to provide a comprehensive understanding about the optimization performance and CL-DP guarantees in FL.

2.1. Model clipping versus Difference Clipping

The two major clipping strategies used in protecting CL-DP for FL algorithms are *local model* clipping and *local update difference* clipping, as we describe below.

1. **Model clipping** (Wei et al., 2020a): The clients directly clip the models sent to the server. For FedAvg algorithm, this means performing $\text{clip}(x_i^{t,Q}, c)$. This method appears to be straightforward, but clipping the model directly results in relatively large clipping threshold, so it requires to add larger perturbation.

2. **Difference clipping** (Geyer et al., 2017): The clients clip the local update difference between the initial model and the output model according to (4). This method needs to record the initial model, the update difference typically has smaller magnitudes than the model itself, so the clipping threshold and the perturbation can be smaller than using model clipping. Note that when $Q = 1$, the difference clipping is equivalent to the standard

mini-batch gradient clipping (i.e., the DP-SGD), but in the general case where $Q > 1$, their behaviors are very different.

Below we analyze how they perform on simple quadratic problems. Our results indicate that the difference clipping strategy is more preferable, because it is less likely to have strong impact on the optimization performance. The full proofs of the claims are given in Appendix A.3.

Claim 2.1. *Given any constant clipping threshold c, there exists a convex quadratic problem, for which FedAvg with model clipping does not converge to the global optimal solution with any fixed $Q \geq 1$ and $\eta_l > 0$.*

Claim 2.2. *For all linear regression problem with fixed clipping threshold c, there exist η_l and local update step $Q \geq 1$ such that FedAvg with difference clipping converges to the global optimal solution. Furthermore, there exists a linear regression problem such that under the same c, η_l and Q, FedAvg with difference clipping converges to a better solution with smaller loss than the original FedAvg.*

Remark 1. To prove Claim 2.1, we construct a problem whose magnitude of the optimal solution is larger than the clipping threshold. Then FedAvg with model clipping will converge to a stationary point with magnitude bounded by the clipping threshold, therefore the algorithm will not converge to global optimal solution.

The technique to prove the first part of Claim 2.2 is related to the analysis for centralized gradient clipping algorithms in Song et al. (2020). The main difference is that our algorithm consider Q steps of local update before clipping. We show that by allowing multiple local updates, FedAvg algorithm with difference clipping optimizes the sum of the Huberzied re-weighted local loss functions. By properly choosing the learning rate η_l for each local loss function, we can balance the re-weighting factors so that the optimal solution to the new loss function matches the solution to the original problem. ■

The above claims indicate that the difference clipping should outperform the model clipping in terms of convergence guarantees. Therefore, in the subsequent analysis, we will focus on understanding the difference clipping enabled FL algorithms. In particular, we consider the Clipping-Enabled FedAvg (CE-FedAvg) algorithm described in Algorithm 2, which combines the difference clipping with the slightly generalized FedAvg algorithm described in Algorithm 1 (which uses two stepsizes η_l, η_g, one for local and one for global updates, respectively). The reason to consider such a *bi-level-stepsize* version of FedAvg is that, it has been proved to have superior performance, especially when not all clients participate in each round of communication (Karimireddy et al., 2020; Yang et al., 2021).

Algorithm 2 Clipping-enabled FedAvg Algorithm (CE-FedAvg)

Initialize: $\mathbf{x}_i^0 \triangleq \mathbf{x}^0, i = 1, \ldots, N$
for $t = 0, \ldots, T-1$ *(stage)* **do**
  for $i \in \mathcal{P}_t \subseteq [N]$ in parallel **do**
    Update agents' $\mathbf{x}_i^{t,0} = \mathbf{x}^t$
    for $q = 0, \ldots, Q-1$ *(iteration)* **do**
      Compute stochastic gradient $g_i^{t,q}$ with $\mathbb{E}[g_i^{t,q}] = \nabla f_i(x_i^{t,q})$
      Local update: $\mathbf{x}_i^{t,q+1} = \mathbf{x}_i^{t,q} - \eta_l g_i^{t,q}$
    end for
    Compute update difference: $\Delta \mathbf{x}_i^t = \mathbf{x}_i^{t,Q} - \mathbf{x}_i^{t,0}$
    Clip: $\hat{\Delta}\mathbf{x}_i^t = \text{clip}(\Delta \mathbf{x}_i^t, c)$, where $\text{clip}(\cdot)$ is defined in (3)
  end for
  Global averaging: $\mathbf{x}^{t+1} = \mathbf{x}^t + \eta_g \frac{1}{|\mathcal{P}_t|} \sum_{i \in \mathcal{P}_t} \hat{\Delta}\mathbf{x}_i^t$
end for

2.2. Empirical Results

Experiment Setting. To have a thorough understanding about how the difference clipping can impact the FedAvg, we conduct numerical experiments with different models, datasets and local data distributions. We compare the test accuracies between CE-FedAvg and the original FedAvg. Note that in this set of experiments we do not consider the privacy issues yet, so we do not add perturbation.

To have a fair comparison, we first fix T, N, $|\mathcal{P}_t|$ and optimize the hyper-parameters Q, η_l and η_g for CE-FedAvg and set them to be identical for both FedAvg and CE-FedAvg, so that the difference between the performance of CE-FedAvg and FedAvg can only be larger. We first run the original FedAvg, compute $\|\Delta \mathbf{x}_i^t\|$ and average over all clients i and iterations t to obtain $\bar{\Delta}$ and choose the clipping threshold $c = 0.5\bar{\Delta}$.

We run the algorithm using AlexNet (Krizhevsky et al., 2012) and ResNet-18 (He et al., 2016) with EMNIST dataset (Cohen et al., 2017) and Cifar-10 dataset (Krizhevsky et al., 2009) for comparison. We split the dataset in two different ways: 1) *IID Data* setting, where the samples are uniformly distributed to each client; 2) *Non-IID Data* setting, where the clients have unbalanced samples. Details are described below. For EMNIST digit classification dataset, each client has 500 samples without overlapping. In the IID case, each client has around 50 samples of each class and in the Non-IID case, there are 8 classes (each has around 5 samples) and 2 classes (each has 230 samples) on each client. For the Cifar-10 dataset, in the IID case (resp. Non-IID case), each client also has 500 samples (resp. 50 samples); these samples can overlap with those on the other clients and the samples on each client are uniformly distributed in 10 classes, i.e., each client has 50 samples (resp. 5 samples) from each class.

We also run the algorithm using the LSTM model used in (Reddi et al., 2021) on the NLP problem with Shakespeare

dataset (Caldas et al., 2018) , in addition to the image classification problem. The dataset is also split in two different ways: 1) *IID Data* setting, where samples are uniformly distributed, each client has 3712 samples; 2) Non-IID Data setting where samples are split by the clients according to the way given in (Caldas et al., 2018).

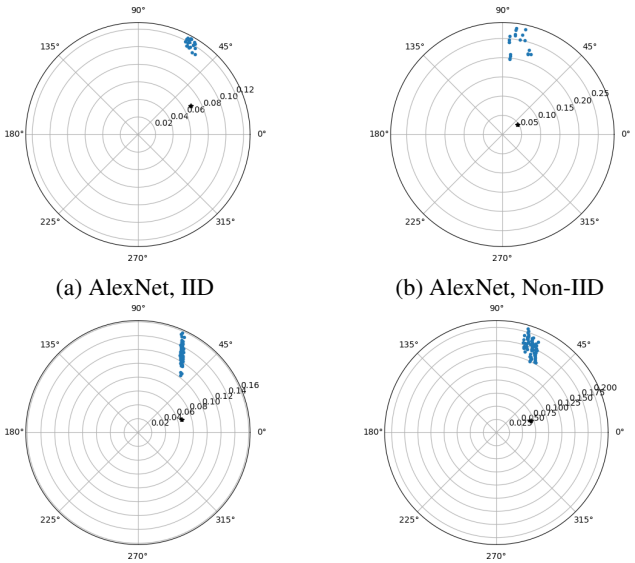
Performance Degradation. In Table 1, we compare the classification results produced by using AlexNet and ResNet-18 on the two datasets.

There are three interesting observations: 1) The data distribution will greatly affect the clipping performance in FL. When data are IID across the clients, clipping has far less impact on the final accuracy, otherwise the clipping will introduce some accuracy drop to the trained models; 2) Clipping has quite different impact on different models – the best accuracy of the models drops 0.10% and 3.60% for ResNet-18 and AlexNet on EMNIST, respectively. The drop is 1.55% for ResNet-18 and 7.30% for AlexNet on Cifar-10, comparing CE-FedAvg with non-clipped version on the Non-IID data; 3) Data complexity also affects the behavior of the CE-FedAvg – the accuracy drop on Cifar-10 dataset is much larger than that on EMNIST dataset.

The empirical experiments show that heterogeneous data distribution among the clients is one of the main causes of the different behavior between the clipped and non-clipped algorithms. And the data heterogeneity issue is unique in FL where the data cannot be shared.

Update Difference Distribution. To further understand the clipping procedure, we plot in Fig. 1, Fig. 2 and Fig. 3 the magnitudes of local updates $\|\Delta \mathbf{x}_i^t\|$ and the cosine angles between the last iteration's global update and $\Delta \mathbf{x}_i^t$: $\cos^{-1}\left(\frac{\left\langle \Delta \mathbf{x}_i^t, \frac{1}{|\mathcal{P}_t|}\sum_{i\in\mathcal{P}_{t-1}} \Delta \mathbf{x}_i^{t-1}\right\rangle}{\|\Delta \mathbf{x}_i^t\|\left\|\frac{1}{|\mathcal{P}_t|}\sum_{i\in\mathcal{P}_{t-1}} \Delta \mathbf{x}_i^{t-1}\right\|}\right)$. Due to page limitation, we only put the distribution of communication round $T = 16$. More detailed results are given in Appendix A.2. In the plots, we mainly focus on the variance of the magnitudes of the clients' update difference (i.e., the blue dots). Larger variance indicates that the updates made by different clients are more different from each other.

Comparing Fig. 1 with Fig. 2 we can see that the update magnitudes on EMNIST dataset are more concentrated than that on Cifar-10 dataset by having smaller mean and variance. Similarly, by comparing Fig. 1a with Fig. 1b or Fig. 1c with Fig. 1d, or Fig. 3a with Fig. 3b, it is clear that the local update magnitudes are more concentrated on IID data than on Non-IID data. Moreover, ResNet-18 has a more concentrated distribution of update magnitudes than AlexNet. Importantly, comparing Table 1 with Fig. 1 and Fig. 2, one can observe that the drop in final accuracy of a model caused by clipping is correlated with *the degree of concentration* of update magnitudes, as AlexNet with less concentrated update magnitudes suffers more from clipping, while ResNet-18 exhibits the opposite behavior.



(a) AlexNet, IID (b) AlexNet, Non-IID

(c) ResNet-18, IID (d) ResNet-18, Non-IID

Figure 1: The distribution of local updates for AlexNet and ResNet-18 on IID and Non-IID data at communication round 16 for EMNIST dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of averaged local update at iteration t.

The above results about the update difference distributions match the accuracy results in Table 1, in the sense that clipping performs worse when update differences distribution has a larger divergence and vise versa. Inspired by this observation, in the next subsection, we will characterize the impact of clipping based on the degree of concentration in local updates and develop the convergence analysis of CE-FedAvg.

3. Convergence Analysis of Clipping-Enabled FedAvg

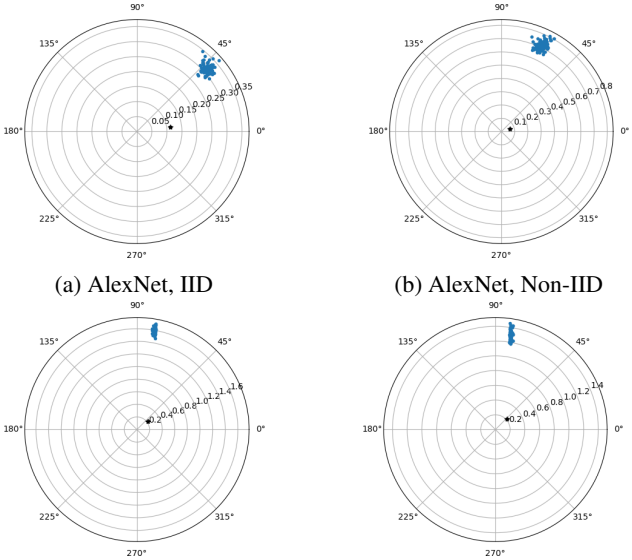
In this section, we analyze the theoretical performance of CE-FedAvg as well as its randomly perturbed version, in order to gain a better understanding of our previous empirical observations and the trade-off between the convergence performance of FedAvg and its DP guarantees.

Towards this end, we will provide the convergence analysis and privacy guarantees for the DP-FedAvg algorithm (Algorithm 3). Compared to CE-FedAvg, this algorithm further adds a random perturbation $\mathbf{z}_i^t$ to the locally clipped model differences. During the communication, we assume that the attacker can only observe the aggregated update $\sum_{i\in\mathcal{P}_t} \tilde{\Delta}\mathbf{x}_i^t$, and this can be guaranteed by using secure aggregation (Bonawitz et al., 2017) or assuming secure uplinks of the clients.

Despite the similar mechanism used in DPSGD and DP-FedAvg, let us point their major differences: in

Table 1: The testing accuracy of a) FedAvg and clipping-enabled FedAvg, on IID and Non-IID data. The 4th and 6th columns display both the accuracy of clipping-enabled FedAvg, and its difference with FedAvg.

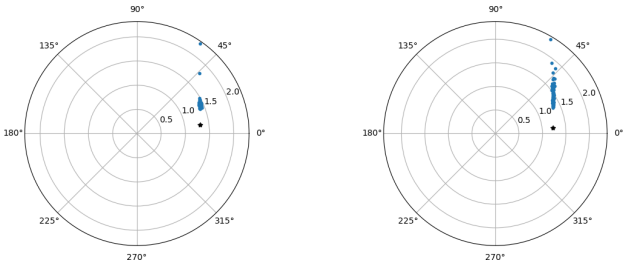
Model	dataset	IID(%)	IID Clipping (diff.)(%)	Non-IID (%)	Non-IID Clipping (diff.)(%)
AlexNet	EMNIST	98.20	98.01 (–0.19)	95.60	92.00 (–3.60)
	Cifar-10	66.01	61.18 (–4.83)	57.14	49.84 (–7.30)
ResNet-18	EMNIST	99.61	99.59 (–0.02)	95.43	95.33 (–0.10)
	Cifar-10	76.36	75.83 (–0.53)	59.46	57.91 (–1.55)



(a) AlexNet, IID (b) AlexNet, Non-IID

(c) ResNet-18, IID (d) ResNet-18, Non-IID

Figure 2: The distribution of local updates for AlexNet and ResNet-18 on IID and Non-IID data at communication round 16 for Cifar-10 dataset.



(a) LSTM, IID (b) LSTM, Non-IID

Figure 3: The distribution of local updates for Stacked LSTM on IID and Non-IID data at communication round 16 for Shakespeare dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of averaged local update at iteration t.

DPSGD, the goal is to protect SL-DP, while DP-FedAvg is to protect CL-DP. The key difference in DP-FedAvg is that the local dataset size is large enough so that after performing multiple local update steps, the resulting model has relatively good performance. By doing so, we can largely reduce the number of communication and the corresponding privacy noise added per communication. Note that DP-FedAvg becomes DPSGD with the following choices of hyperparameters: 1) enlarge the client number to be the same as the size of the dataset, 2) decrease the local dataset size to 1; 3) decrease the number of local update to 1; 4) decrease the privacy noise accordingly.

Algorithm 3 DP-FedAvg Algorithm

Initialize: $\mathbf{x}_i^0 \triangleq \mathbf{x}^0, i = 1, \dots, N$
for $t = 0, \dots, T-1$ *(stage)* **do**
  for $i \in \mathcal{P}_t \subseteq [N]$ in parallel **do**
    Update agents' $\mathbf{x}_i^{t,0} = \mathbf{x}^t$
    for $q = 0, \dots, Q-1$ *(iteration)* **do**
      Compute stochastic gradient $g_i^{t,q}$ with $\mathbb{E}[g_i^{t,q}] = \nabla f_i(x_i^{t,q})$
      Local update: $\mathbf{x}_i^{t,q+1} = \mathbf{x}_i^{t,q} - \eta_l g_i^{t,q}$
    end for
    Compute update difference: $\Delta \mathbf{x}_i^t = \mathbf{x}_i^{t,Q} - \mathbf{x}_i^{t,0}$
    Clip and perturb: $\tilde{\Delta}\mathbf{x}_i^t = \text{clip}(\Delta \mathbf{x}_i^t, c) + \mathbf{z}_i^t$, where $\text{clip}(\cdot)$ is defined in (3)
  end for
  Global averaging: $\mathbf{x}^{t+1} = \mathbf{x}^t + \eta_g \frac{1}{|\mathcal{P}_t|} \sum_{i \in \mathcal{P}_t} \tilde{\Delta}\mathbf{x}_i^t$
end for

3.1. Convergence Analysis

Theorem 3.1 (Convergence of DP-FedAvg)**.** *For Algorithm 3, assume* $\|\nabla f_i(x) - \nabla f_i(y)\| \le L\|x - y\|$, $\forall\, i, x, y$, $\min_x f(x) \ge f^*$; $\mathbb{E}[\|g_i^{t,q} - \nabla f_i(x_i^{t,q})\|^2] \le \sigma_l^2$, $\|g_i^{t,q}\| \le G$, $\forall\ t, q, i$, $\|\nabla f_i(x) - \nabla f(x)\|^2 \le \sigma_g^2$, $\forall i$, *where L is the Lipschitz constant of gradient, σ_l^2 and σ_g^2 are intra-client and inter-client gradient variance, G is the bound on stochastic gradient.*

By letting $\eta_g \eta_l \le \min\{\frac{P}{96Q^2}, \frac{P}{6QL(P-1)}\}$ *and* $\eta_l \le \frac{1}{\sqrt{60}QL}$, *we have*

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\overline{\alpha}^t\|\nabla f(x^t)\|^2] \le \textit{FedAvg terms} + \underbrace{\frac{2\eta_g L d\sigma^2}{\eta_l PQ}}_{\text{caused by privacy noise}}$$
$$+ \underbrace{G^2\frac{4}{T}\sum_{t=1}^{T}\mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}(|\alpha_i^t - \tilde{\alpha}_i^t| + |\tilde{\alpha}_i^t - \overline{\alpha}^t|)\right]}_{\text{caused by clipping}}$$
$$+ \underbrace{\eta_g\eta_l LQG^2\frac{6}{T}\sum_{t=1}^{T}\mathbb{E}\left[\frac{1}{P}\sum_{i=1}^{N}(|\alpha_i^t - \tilde{\alpha}_i^t|^2 + |\tilde{\alpha}_i^t - \overline{\alpha}^t|^2)\right]}_{\text{caused by clipping}}$$

where $P := |\mathcal{P}_t|$, $\alpha_i^t := \frac{c}{\max(c, \eta_l\|\sum_{q=0}^{Q-1} g_i^{t,q}\|)}$, $\tilde{\alpha}_i^t := \frac{c}{\max(c, \eta_l\|\mathbb{E}[\sum_{q=0}^{Q-1} g_i^{t,q}]\|)}$, $\overline{\alpha}^t := \frac{1}{N}\sum_{i=1}^{N}\tilde{\alpha}_i^t$; *$d$ is the dimension of x,* $\gamma_1(T) = \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\overline{\alpha}^t] \le 1$, $\gamma_2(T) = \frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[(\overline{\alpha}^t)^2] \le 1$ *and FedAvg terms* $= \frac{4(f(x^0)-f^*)}{\eta_g\eta_l QT} + \frac{25}{2}\eta_l^2 LQ(\sigma_l^2 + 6Q\sigma_g^2)\gamma_1(T) + \frac{6\eta_g\eta_l L\sigma_l^2}{P}\gamma_2(T)$.

In the bound of Theorem 3.1, the *FedAvg terms* are inherited

from standard FedAvg with two-sided learning rates which can yield a convergence rate of $O(\frac{1}{\sqrt{PQT}} + \frac{1}{T})$ when setting $\eta_g = \sqrt{QP}$ and $\eta_l = \frac{1}{\sqrt{TQL}}$. When there is no clipping bias and privacy noise, Theorem 3.1 exactly recovers the standard convergence bounds for FedAvg up to a constant, see Theorem 1 in (Yang et al., 2021). Aside from *FedAvg terms*, we have two types of extra terms caused by the privacy noise z_i^t and the clipping operation, respectively. We highlight the terms caused by clipping which characterize the estimation bias caused by clipping. The bias can be decomposed into terms caused by $|\alpha_i^t - \tilde{\alpha}_i^t|$ and terms caused by $|\tilde{\alpha}_i^t - \overline{\alpha}^t|$. Notice that since $|\alpha_i^t - \tilde{\alpha}_i^t| \leq \eta_l |\| \sum_{q=0}^{Q-1} g_i^{t,q} \| - \| \mathbb{E}[\sum_{q=0}^{Q-1} g_i^{t,q}] \| |$, it is clear $\mathbb{E}[|\alpha_i^t - \tilde{\alpha}_i^t|]$ will be small if the stochastic local updates have smaller variance in norm. This term characterizes the bias caused by local update variance. In addition, $\mathbb{E}[|\tilde{\alpha}_i^t - \overline{\alpha}^t|]$ will be small if the expected local model updates have similar magnitudes in norm across clients and $\mathbb{E}[|\tilde{\alpha}_i^t - \overline{\alpha}^t|] = 0$ if $\|\mathbb{E}[\Delta x_i^t]\| = \|\mathbb{E}[\Delta x_j^t]\|, \forall i, j$. This term shows the bias caused by cross-client update variance.

Key insight: In FL, sometimes each client will have limited amount of data, and the local model updates can be performed with small σ_l or even $\sigma_l = 0$ (full batch update). Thus, the bias caused by $|\alpha_i^t - \tilde{\alpha}_i^t|$ can be small and is avoidable. However, the bias caused by $|\tilde{\alpha}_i^t - \overline{\alpha}^t|$ is *unavoidable* since this term will not diminish even each client updates its local model with full batch gradient. In addition, this term might be large with heterogeneous data distribution since the heterogeneity may induce quite disparate gradient distributions across clients. Thus, it is crucial to investigate the bias caused by $|\tilde{\alpha}_i^t - \overline{\alpha}^t|$ in practice. Note that $|\tilde{\alpha}_i^t - \overline{\alpha}^t|$ is fully controlled by differences in magnitudes of local model updates when $\sigma_l = 0$ for fixed c. Going back to Fig. 1, we do see that how such differences in update magnitudes can be affected by both the neural network models and data heterogeneity. From another intuitive perspective, clipping operation is similar to changing learning rates in a data-dependent way. Inconsistent learning rate across workers can cause a problem known as objective inconsistency (Wang et al., 2020a) in federated learning, which also support that $|\tilde{\alpha}_i^t - \overline{\alpha}^t|$ can affect model performance.

3.2. Differential Privacy Guarantee

The privacy guarantee of DP-FedAvg can be characterized by standard privacy theorems on Gaussian mechanism. We rephrase Abadi et al. (2016, Theorem 1) for client privacy in Theorem 3.2.

Theorem 3.2 (Privacy of DP-FedAvg)**.** *There exist constants u and v so that given the number of iterations T, for any $\epsilon \leq uq^2T$ with $q = \frac{P}{N}$ and $|\mathcal{P}_t| = P$, $\forall t$, Algorithm 1 is (ϵ, δ)-differentially private for any $\delta > 0$ if* $\sigma^2 \geq v\frac{c^2 PT \ln(\frac{1}{\delta})}{N^2\epsilon^2}$.

The privacy-utility trade-off of DP-FedAvg can be analyzed by substituting σ^2 from Theorem 3.2 into Theorem 3.1. To get more insights on how parameters like T, η_g, η_l and ϵ affect DP-FedAvg, let us consider simplified Theorem 3.1 in Corollary 3.2.1 with $c \geq \eta_l QG$ and σ^2 substituted . If $c' < G$ in Corollary 3.2.1, then there will be extra bias terms inherited from the bound in Theorem 3.1.

Corollary 3.2.1 (Convergence with privacy guarantee)**.** Assume all assumptions in Theorem 3.1, for any clipping threshold $c = \eta_l Q c'$ with $c' \geq G$, and set σ^2 as in Theorem 3.2, for any (ϵ, δ) satisfying the constraints in Theorem 3.2, we have

$$\frac{1}{T}\sum_{t=1}^{T}\mathbb{E}[\|\nabla f(x^t)\|^2] \tag{5}$$
$$\leq \underbrace{O\left(\frac{1}{\eta_g\eta_l QT} + \eta_l^2 Q^2 + \frac{\eta_g\eta_l}{P}\right)}_{\text{standard terms for FedAvg}} + \underbrace{O\left(\frac{\eta_g\eta_l QTd\ln(\frac{1}{\delta})}{N^2\epsilon^2}\right)}_{\text{caused by privacy noise}}$$

and the best rate one can get from the above bound is $\tilde{O}(\frac{\sqrt{d}}{N\epsilon})$ by optimizing η_g, η_l, Q, T.

A direct implication of Corollary 3.2.1 is that the big-O convergence rate of DP-FedAvg is the same as differentially private SGD (DP-SGD) in terms of d, ϵ, and N (the number of samples in DP-SGD).

4. Numerical Experiments

In the experiment, we compare the performance of FedAvg, CE-FedAvg and DP-FedAvg on two datasets. In both experiments, we set client number $N = 1920$, the number of client participates in each round $|\mathcal{P}_t| = 80$, $\forall\ t$, the number of local iterations $Q = 32$ and the mini-batch size 64. The clipping threshold is set to 50% of the average (over clients and iterations) of local update magnitudes recorded in FedAvg. For DP-FedAvg we set the clipping threshold the same as in CE-FedAvg, we fix the number of communication rounds and privacy budget for the algorithms to obtain the noise variance that needs to be added. These hyper-parameters are optimized for DP-FedAvg Among all the experiments, we fix privacy budget $\delta = 10^{-5}$.

EMNIST dataset. We use the digit part of the EMNIST dataset, which has 240K training samples and 40K testing samples. We distribute the data in the Non-IID way described in Section II and each client has 125 samples. We conduct experiments on a 2-layer MLP with one hidden layer, AlexNet, ModelNetV2 (Sandler et al., 2018) and ResNet-18. The results are listed in Table 3 and Figure 4.

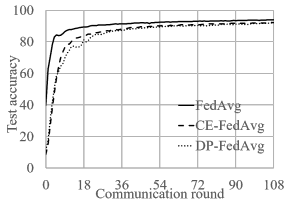
Cifar-10 dataset. The dataset we use is the Cifar-10 dataset, which has 50K training samples and 10K testing samples. We distribute the data in the IID way described in Section II

Table 2: The accuracy difference between a) FedAvg and CE-FedAvg and b) CE-FedAvg and DP-FedAvg on IID Cifar-10 dataset. The clipping threshold is 0.5 of the average magnitude and privacy budget $\epsilon = 1.5$ for MLP, AlexNet and ResNet-18.

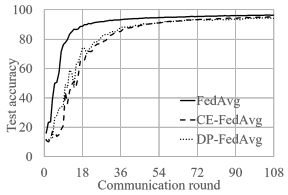
Model	# Parameters	# Layers	Accuracy (%)	Clipping (diff.)(%)	DP (diff.)(%)
MLP	616K	2	51.90	44.51 (–7.39)	43.60 (–0.90)
AlexNet	3.3M	7	66.01	61.18 (–4.83)	61.36 (**+0.18**)
ResNet-18	11.1M	18	76.36	75.83 (**–0.53**)	70.68 (–5.15)

Table 3: The accuracy difference between a) FedAvg and clip-enabled FedAvg and b) clip-enabled FedAvg and DP-FedAvg on Non-IID EMNIST dataset. The clipping threshold is 0.5 of the average magnitude and privacy budget $\epsilon = 1.5$ for MLP, AlexNet and MobileNetV2 and $\epsilon = 5$ for ResNet-18.

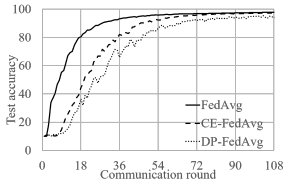
Model	# Parameters	# Layers	Accuracy (%)	Clipping (diff.)(%)	DP (diff.)(%)
MLP	159K	2	94.0	93.1 (–1.84)	92.8 (–0.29)
AlexNet	3.3M	7	96.4	94.9 (–1.47)	94.7 (**–0.16**)
MobileNetV2	2.3M	24	97.8	97.4 (–0.35)	95.8 (–1.62)
ResNet-18	11.1M	18	95.2	95.3 (**+0.15**)	91.5 (–3.76)*



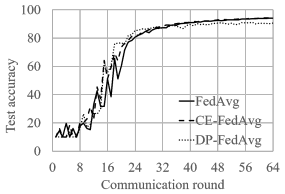
(a) MLP, $\epsilon = 1.5$



(b) AlexNet, $\epsilon = 1.5$



(c) MobileNetV2, $\epsilon = 1.5$



(d) ResNet-18, $\epsilon = 5$

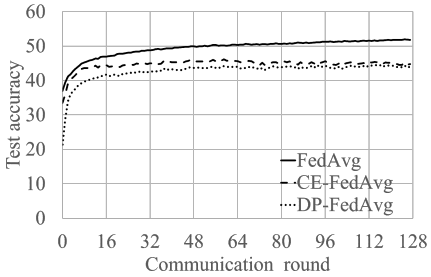
Figure 4: The test accuracy of FedAvg, CE-FedAvg and DP-FedAvg on different models on EMNIST. The privacy budgets for MLP, AlexNet and MobileNet are $\epsilon = 1.5$ while for ResNet, we set $\epsilon = 5$.

and each client has 500 samples. We conduct experiments on a 2-layer MLP with one hidden layer, AlexNet and ResNet-18. The results are listed in Table 2 and Figure 5.

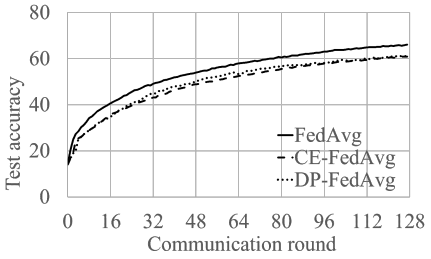
Discussion. Let us discuss the relation between our empirical observations and the theoretical results.

1) It appears that when the underlying machine learning model is *structured* (e.g., many layers, has convolution layers, skip connections, etc), the update difference of FedAvg becomes *concentrated*, yielding a better clipping performance (as suggested by the terms related to clipping in Theorem 3.1);

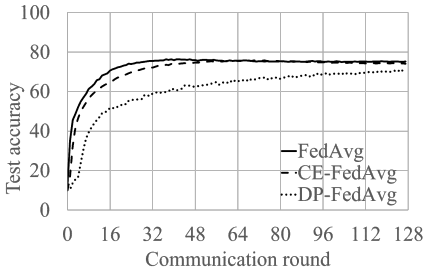
2) When the model has too many parameters and/or layers, they are sensitive to privacy noise. This is reasonable since the error term caused by privacy noise in Theorem 3.1 is linearly dependent on the size of the model d and the square of the Lipschitz constant L (note, that $\eta_\ell \propto 1/L$). From Herrera et al. (2020, Corollary 3.3), we know that L increases exponentially with the number of layers. Therefore, larger and deeper models are potentially more



(a) MLP, $\epsilon = 1.5$



(b) AlexNet, $\epsilon = 1.5$



(c) ResNet-18, $\epsilon = 1.5$

Figure 5: The test accuracy of FedAvg, CE-FedAvg and DP-FedAvg on different models on Cifar-10. The privacy budgets for MLP, AlexNet and ResNet are $\epsilon = 1.5$.

sensitive to privacy noise.

3) We conjecture that, to ensure good performance of DP-FedAvg, we need to pick a neural network that is structured enough, while not having too many variables and too many layers.

Acknowledgements

We thank the anonymous reviewers for valuable feedback on the merit of the work, and helpful suggestions on improving the presentation. Z. S. Wu was supported in part by the NSF CNS #2120603, a CMU CyLab 2021 grant, a Google Faculty Research Award, and a Mozilla Research Grant. M. Hong, X. Chen and X. Zhang are supported in part by NSF grants CIF-1910385, CMMI-1727757 and AFOSR grant 19RT0424.

References

Abadi, M., Chu, A., Goodfellow, I., McMahan, H. B., Mironov, I., Talwar, K., and Zhang, L. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.

Andrew, G., Thakkar, O., McMahan, H. B., and Ramaswamy, S. Differentially private learning with adaptive clipping. *arXiv preprint arXiv:1905.03871*, 2019.

Bassily, R., Smith, A., and Thakurta, A. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th Annual Symposium on Foundations of Computer Science*, pp. 464–473. IEEE, 2014.

Bonawitz, K., Ivanov, V., Kreuter, B., Marcedone, A., McMahan, H. B., Patel, S., Ramage, D., Segal, A., and Seth, K. Practical secure aggregation for privacy-preserving machine learning. In *proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*, pp. 1175–1191, 2017.

Caldas, S., Duddu, S. M. K., Wu, P., Li, T., Konečnỳ, J., McMahan, H. B., Smith, V., and Talwalkar, A. Leaf: A benchmark for federated settings. *arXiv preprint arXiv:1812.01097*, 2018.

Chen, X., Wu, S. Z., and Hong, M. Understanding gradient clipping in private sgd: A geometric perspective. *Advances in Neural Information Processing Systems*, 33, 2020.

Choudhury, O., Gkoulalas-Divanis, A., Salonidis, T., Sylla, I., Park, Y., Hsu, G., and Das, A. Differential privacy-enabled federated learning for sensitive health data. *arXiv preprint arXiv:1910.02578*, 2019.

Cohen, G., Afshar, S., Tapson, J., and Van Schaik, A. EMNIST: Extending MNIST to handwritten letters. In *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 2921–2926. IEEE, 2017.

Dwork, C. and Roth, A. The algorithmic foundations of differential privacy. *Foundations and Trends in Theoretical Computer Science*, 9(3-4):211–407, 2014.

Dwork, C., McSherry, F., Nissim, K., and Smith, A. *Calibrating Noise to Sensitivity in Private Data Analysis*, pp. 265–284. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006. ISBN 978-3-540-32732-5. doi: 10.1007/11681878_14. URL http://dx.doi.org/10.1007/11681878_14.

Geyer, R. C., Klein, T., and Nabi, M. Differentially private federated learning: A client level perspective. *arXiv preprint arXiv:1712.07557*, 2017.

Hard, A., Rao, K., Mathews, R., Ramaswamy, S., Beaufays, F., Augenstein, S., Eichner, H., Kiddon, C., and Ramage, D. Federated learning for mobile keyboard prediction. *arXiv preprint arXiv:1811.03604*, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Herrera, C., Krach, F., and Teichmann, J. Estimating full lipschitz constants of deep neural networks. *arXiv preprint arXiv:2004.13135*, 2020.

Karimireddy, S. P., Kale, S., Mohri, M., Reddi, S., Stich, S., and Suresh, A. T. Scaffold: Stochastic controlled averaging for federated learning. In *International Conference on Machine Learning*, pp. 5132–5143. PMLR, 2020.

Khaled, A., Mishchenko, K., and Richtárik, P. First analysis of local gd on heterogeneous data. *arXiv preprint arXiv:1909.04715*, 2019.

Konečnỳ, J., McMahan, H. B., Yu, F. X., Richtárik, P., Suresh, A. T., and Bacon, D. Federated learning: Strategies for improving communication efficiency. *arXiv preprint arXiv:1610.05492*, 2016.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Krizhevsky, A., Sutskever, I., and Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25: 1097–1105, 2012.

Liang, X., Shen, S., Liu, J., Pan, Z., Chen, E., and Cheng, Y. Variance reduced local sgd with lower communication complexity. *arXiv preprint arXiv:1912.12844*, 2019.

McMahan, H. B., Ramage, D., Talwar, K., and Zhang, L. Learning differentially private recurrent language models. *arXiv preprint arXiv:1710.06963*, 2017.

Pichapati, V., Suresh, A. T., Yu, F. X., Reddi, S. J., and Kumar, S. Adaclip: Adaptive clipping for private sgd. *arXiv preprint arXiv:1908.07643*, 2019.

Reddi, S., Charles, Z., Zaheer, M., Garrett, Z., Rush, K., Konečnỳ, J., Kumar, S., and McMahan, H. B. Adaptive federated optimization. *International Conference on Learning Representations*, 2021.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-C. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4510–4520, 2018.

Song, S., Thakkar, O., and Thakurta, A. Characterizing private clipped gradient descent on convex generalized linear problems. *arXiv preprint arXiv:2006.06783*, 2020.

Song, S., Steinke, T., Thakkar, O., and Thakurta, A. Evading the curse of dimensionality in unconstrained private glms. In *International Conference on Artificial Intelligence and Statistics*, pp. 2638–2646. PMLR, 2021.

Triastcyn, A. and Faltings, B. Federated learning with bayesian differential privacy. In *2019 IEEE International Conference on Big Data (Big Data)*, pp. 2587–2596. IEEE, 2019.

Truex, S., Baracaldo, N., Anwar, A., Steinke, T., Ludwig, H., Zhang, R., and Zhou, Y. A hybrid approach to privacy-preserving federated learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, pp. 1–11, 2019.

Truex, S., Liu, L., Chow, K.-H., Gursoy, M. E., and Wei, W. LDP-Fed: Federated learning with local differential privacy. In *Proceedings of the Third ACM International Workshop on Edge Systems, Analytics and Networking*, pp. 61–66, 2020.

Wang, J., Liu, Q., Liang, H., Joshi, G., and Poor, H. V. Tackling the objective inconsistency problem in heterogeneous federated optimization. *arXiv preprint arXiv:2007.07481*, 2020a.

Wang, L., Jia, R., and Song, D. D2p-fed: Differentially private federated learning with efficient communication. *arXiv preprint arXiv:2006.13039*, 2020b.

Wei, K., Li, J., Ding, M., Ma, C., Yang, H. H., Farokhi, F., Jin, S., Quek, T. Q., and Poor, H. V. Federated learning with differential privacy: Algorithms and performance analysis. *IEEE Transactions on Information Forensics and Security*, 15:3454–3469, 2020a.

Wei, W., Liu, L., Loper, M., Chow, K.-H., Gursoy, M. E., Truex, S., and Wu, Y. A framework for evaluating gradient leakage attacks in federated learning. *arXiv preprint arXiv:2004.10397*, 2020b.

Yang, H., Fang, M., and Liu, J. Achieving linear speedup with partial worker participation in Non-IID federated learning. *International Conference on Learning Representations*, 2021.

Zhang, X., Hong, M., Dhople, S., Yin, W., and Liu, Y. FedPD: A federated learning framework with optimal rates and adaptivity to Non-IID data, 2020.

Zhao, B., Mopuri, K. R., and Bilen, H. idlg: Improved deep leakage from gradients. *arXiv preprint arXiv:2001.02610*, 2020.

Zhu, L. and Han, S. Deep leakage from gradients. In *Federated Learning*, pp. 17–31. Springer, 2020.

A. Appendix

A.1. Proof of Theorem 3.1

By Lipschitz smoothness, we have

$$f(x_{t+1}) \leq f(x_t) + \langle \nabla f(x_t), x_{t+1} - x_t \rangle + \frac{L}{2}\|x_{t+1} - x_t\|^2. \tag{6}$$

Before we proceed, we define following quantities to simplify notation:

$$\begin{aligned}
&\alpha_i^t := \frac{c}{\max(c, \eta_l \|\sum_{q=0}^{Q-1} g_i^{t,q}\|)}, \quad \tilde{\alpha}_i^t := \frac{c}{\max(c, \eta_l \|\mathbb{E}[\sum_{q=0}^{Q-1} g_i^{t,q}]\|)}, \quad \overline{\alpha}^t := \frac{1}{N}\sum_{i=1}^N \tilde{\alpha}_i^t, \\
&\Delta_i^t := -\eta_l \sum_{q=0}^{Q-1} g_i^{t,q} \cdot \alpha_i^t, \quad \tilde{\Delta}_i^t := -\eta_l \sum_{q=0}^{Q-1} g_i^{t,q} \cdot \tilde{\alpha}_i^t, \\
&\overline{\Delta}_i^t := -\eta_l \sum_{q=0}^{Q-1} g_i^{t,q} \cdot \overline{\alpha}^t, \quad \breve{\Delta}_i^t := -\eta_l \sum_{q=0}^{Q-1} \nabla f_i(x_i^{t,q}) \cdot \overline{\alpha}^t \quad P := |\mathcal{P}_t|,
\end{aligned} \tag{7}$$

where the expectation in $\tilde{\alpha}_i^t$ is taken over all possible randomness.

By using the above definitions, the model difference between two consecutive iterations can be expressed as:

$$x_{t+1} - x_t = \eta_g \frac{1}{P} \sum_{i \in \mathcal{P}_t} (\Delta_i^t + z_i^t),$$

with $z_i^t \sim \mathcal{N}(0, \sigma^2 I)$. Using the above expressions, and take an conditional expectation of (6) (conditioned on x_t), we obtain:

$$\begin{aligned}
\mathbb{E}[f(x_{t+1})] &\leq f(x_t) + \eta_g \left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{P}\sum_{i\in\mathcal{P}_t} \Delta_i^t + z_i^t\right] \right\rangle + \frac{L}{2}\eta_g^2 \mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t} \Delta_i^t + z_i^t\right\|^2\right] \\
&= f(x_t) + \eta_g \left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{P}\sum_{i\in\mathcal{P}_t} \Delta_i^t\right] \right\rangle + \frac{L}{2}\eta_g^2 \mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t} \Delta_i^t\right\|^2\right] + \frac{L}{2}\eta_g^2 \frac{1}{P}\sigma^2 d,
\end{aligned} \tag{8}$$

where d in the last expression represents dimension of x_t; in the last equation we use the fact that z_i^t is zero mean.

Next, we will analyze the bias caused by clipping, through analyzing the first order term in (8). Towards this end, we have the following series of relations:

$$\begin{aligned}
&\left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{P}\sum_{i\in\mathcal{P}_t} \Delta_i^t\right] \right\rangle \\
&\overset{(i)}{=} \left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{P}\mathbb{E}_i[\sum_{i\in\mathcal{P}_t} \Delta_i^t]\right] \right\rangle = \left\langle \nabla f(x_t), \frac{1}{P} P \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \Delta_i^t\right] \right\rangle \\
&= \left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \Delta_i^t - \tilde{\Delta}_i^t\right] \right\rangle + \left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \tilde{\Delta}_i^t - \overline{\Delta}_i^t\right] \right\rangle \\
&+ \left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \overline{\Delta}_i^t\right] \right\rangle
\end{aligned} \tag{9}$$

where (i) we takes expectation on the randomness of the client sampling, i.e., $\mathbb{E}_i\, \Delta_i^t = \frac{1}{N}\sum_{i=1}^N \Delta_i^t$. The first two terms of RHS of the above equality can be viewed as bias caused by clipping. The first order predicted descent can be analyzed from

the last term by completing the square:

$$
\begin{aligned}
&\left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^{N}\overline{\Delta}_i^t\right]\right\rangle \\
\overset{(i)}{=}&\mathbb{E}\left[\left\langle \nabla f(x_t), \frac{1}{N}\sum_{i=1}^{N}\breve{\Delta}_i^t\right]\right\rangle \\
\overset{(ii)}{=}&\frac{-\eta_l\overline{\alpha}^t Q}{2}\|\nabla f(x_t)\|^2 - \frac{\eta_l\overline{\alpha}^t}{2Q}\mathbb{E}\left[\left\|\frac{1}{\eta_l N\overline{\alpha}^t}\sum_{i=1}^{N}\breve{\Delta}_i^t\right\|^2\right] \\
&+ \frac{\eta_l\overline{\alpha}^t}{2}\underbrace{\mathbb{E}\left[\left\|\sqrt{Q}\nabla f(x_t) - \frac{1}{\sqrt{Q}}\frac{1}{\eta_l N\overline{\alpha}^t}\sum_{i=1}^{N}\breve{\Delta}_i^t\right\|^2\right]}_{A_1},
\end{aligned}
\tag{10}
$$

where (i) comes from $\mathbb{E}\,\overline{\Delta}_i^t = \breve{\Delta}_i^t$, (ii) is because $\langle a, b\rangle = -\frac{1}{2}\|a\|^2 - \frac{1}{2}\|b\|^2 + \frac{1}{2}\|a-b\|^2$ holds true for any vector a, b. We further upper bound A_1 as

$$
\begin{aligned}
A_1 =&Q\mathbb{E}\left[\left\|\nabla f(x_t) - \frac{1}{QN}\sum_{i=1}^{N}\sum_{q=0}^{Q-1}\nabla f_i(x_i^{t,q})\right\|^2\right] \\
=&Q\mathbb{E}\left[\left\|\frac{1}{QN}\sum_{i=1}^{N}\sum_{q=0}^{Q-1}\nabla f_i(x^t) - \nabla f_i(x_i^{t,q})\right\|^2\right] \\
\leq&\frac{1}{N}\sum_{i=1}^{N}\sum_{q=0}^{Q-1}\mathbb{E}[\|\nabla f_i(x^t) - \nabla f_i(x_i^{t,q})\|^2] \\
\leq&\frac{1}{N}\sum_{i=1}^{N}\sum_{q=0}^{Q-1}L^2\mathbb{E}[\|x^t - x_i^{t,q}\|^2] \\
\leq&L^2 5Q^2\eta_l^2(\sigma_l^2 + 6Q\sigma_g^2) + L^2 30Q^3\eta_l^2\|\nabla f(x_t)\|^2
\end{aligned}
\tag{11}
$$

where the first inequality comes from Jensen's inequality, the second inequality comes from L-smoothness and the last inequality is due to Lemma 3 in (Reddi et al., 2021).

Now we turn to upper bounding the second order term in (8), as follows

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\Delta_i^t\right\|^2\right] \\
\leq&3\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\Delta_i^t - \tilde{\Delta}_i^t\right\|^2\right] + 3\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\tilde{\Delta}_i^t - \overline{\Delta}_i^t\right\|^2\right] + 3\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\overline{\Delta}_i^t\right\|^2\right].
\end{aligned}
\tag{12}
$$

We can bound the expectation in the last term of (12) as follows:

$$
\begin{aligned}
&\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\overline{\Delta}_i^t\right\|^2\right] \\
=&\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\left(\eta_l\sum_{q=0}^{Q-1}g_i^{t,q}\cdot\overline{\alpha}^t\right)\right\|^2\right] \\
\leq&\eta_l^2\mathbb{E}\left[2\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\sum_{q=0}^{Q-1}\nabla f(x_i^{t,q})\cdot\overline{\alpha}^t\right\|^2+2\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\sum_{q=0}^{Q-1}(\nabla f(x_i^{t,q})-g_i^{t,q})\cdot\overline{\alpha}^t\right\|^2\right] \\
\leq&2\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\breve{\Delta}_i^t\right\|^2\right]+\frac{2}{P}\eta_l^2\overline{\alpha}^2Q\sigma_l^2
\end{aligned}
\tag{13}
$$

where the last inequality is because the assumption that $\mathbb{E}[\|g_i^{t,q}-\nabla f_i(x_i^{t,q})\|^2]\leq\sigma_l^2$. Let us further bound the expectation in the first term of (13) as:

$$
\begin{aligned}
\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\breve{\Delta}_i^t\right\|^2\right]&=\frac{1}{P^2}\mathbb{E}\left[\left\|\sum_{i\in\mathcal{P}_t}\breve{\Delta}_i^t\right\|^2\right] \\
&\overset{(i)}{=}\frac{1}{P^2}\mathbb{E}\left[\mathbb{E}_i\sum_{i\in\mathcal{P}_t}\left\|\breve{\Delta}_i^t\right\|^2+\mathbb{E}_{i,j}\sum_{i\neq j\in\mathcal{P}_t}\left\langle\breve{\Delta}_i^t,\breve{\Delta}_j^t\right\rangle\right] \\
&\overset{(ii)}{=}\frac{1}{P^2}\mathbb{E}\left[\frac{P}{N}\sum_{i=1}^N\left\|\breve{\Delta}_i^t\right\|^2+P(P-1)\left\langle\mathbb{E}_i\,\breve{\Delta}_i^t,\mathbb{E}_j\,\breve{\Delta}_j^t\right\rangle\right] \\
&=\frac{1}{P^2}\mathbb{E}\left[\frac{P}{N}\sum_{i=1}^N\left\|\breve{\Delta}_i^t\right\|^2+P(P-1)\left\|\frac{1}{N}\sum_{i=1}^N\breve{\Delta}_i^t\right\|^2\right],
\end{aligned}
\tag{14}
$$

where in (i) we expand the square and take expectation on the randomness of client sampling, and (ii) is due to independent sampling the clients *with* replacement so that $\mathbb{E}_{i,j}\left\langle\Delta_i^t,\Delta_j^t\right\rangle=\left\langle\mathbb{E}_i\,\Delta_i^t,\mathbb{E}_j\,\Delta_j^t\right\rangle$.

Additionally, note we have:

$$
\begin{aligned}
\mathbb{E}\sum_{i=1}^N\left\|\breve{\Delta}_i^t\right\|^2&\overset{(i)}{=}\mathbb{E}\sum_{i=1}^N\eta_l^2(\overline{\alpha}^t)^2\left\|\sum_{q=0}^{Q-1}\nabla f_i(x^t)+\nabla f_i(x_i^{t,q})-\nabla f_i(x^t)\right\|^2 \\
&\overset{(ii)}{\leq}2\eta_l^2\overline{\alpha}^t\sum_{i=1}^N\left(Q^2\sum_{q=0}^{Q-1}\mathbb{E}\left\|\nabla f_i(x^t)+\nabla f_i(x_i^{t,q})\right\|^2+Q^2\sum_{q=0}^{Q-1}\left\|\nabla f_i(x^t)\right\|^2\right) \\
&\overset{(iii)}{\leq}2\eta_l^2\overline{\alpha}^tN\Big(L^2 5Q^2\eta_l^2(\sigma_l^2+6Q\sigma_g^2)+L^2 30Q^3\eta_l^2\|\nabla f(x_t)\|^2+2Q^3\|\nabla f(x_t)\|^2+2Q^3\sigma_g^2\Big) \\
&=10N\eta_l^4\overline{\alpha}^tL^2Q^2\sigma_l^2+4N\eta_l^2\overline{\alpha}^tQ^3(15L^2\eta_l^2+1)(\|\nabla f(x_t)\|^2+\sigma_g^2).
\end{aligned}
\tag{15}
$$

where (i) comes from the definition of $\breve{\Delta}_i^t$; (ii) comes from the fact that $\|a+b\|^2\leq2(\|a\|^2+\|b\|^2)$; in (iii) we apply (11) to the first term and bound the second term by the assumption that $\|\nabla f_i(x)-\nabla f(x)\|^2\leq\sigma_g^2$.

Combining (8)-(15), we have

$$\begin{aligned}\mathbb{E}[f(x_{t+1})] \leq & f(x_t) - \frac{\eta_g\eta_l\overline{\alpha}^t Q}{2}\|\nabla f(x_t)\|^2 - \frac{\eta_g\eta_l\overline{\alpha}^t}{2Q}\mathbb{E}\left[\left\|\frac{1}{\eta_l N\overline{\alpha}^t}\sum_{i=1}^N \breve{\Delta}_i^t\right\|^2\right] \\ & + \frac{\eta_g\eta_l\overline{\alpha}^t}{2}(5L^2Q^2\eta_l^2(\sigma_l^2+6Q\sigma_g^2) + 30L^2Q^3\eta_l^2\|\nabla f(x_t)\|^2) \\ & + \eta_g\left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \Delta_i^t - \tilde{\Delta}_i^t\right]\right\rangle + \eta_g\left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \tilde{\Delta}_i^t - \overline{\Delta}_i^t\right]\right\rangle \\ & + \frac{3L\eta_g^2(P-1)}{P}\mathbb{E}\left[\left\|\frac{1}{N}\sum_{i=1}^N \breve{\Delta}_i^t\right\|^2\right] + \frac{3L}{P}\eta_g^2\eta_l^2(\overline{\alpha}^t)^2Q\sigma_l^2 + \frac{L}{2}\eta_g^2\frac{1}{P}\sigma^2 d \\ & + \frac{30}{P}\eta_l^4\eta_g^2\overline{\alpha}^t L^2Q^2\sigma_l^2 + \frac{12}{P}\eta_l^2\eta_g^2\overline{\alpha}^t Q^3(15L^2\eta_l^2+1)(\|\nabla f(x_t)\|^2+\sigma_g^2) \\ & + \frac{3L}{2}\eta_g^2\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\Delta_i^t - \tilde{\Delta}_i^t\right\|^2\right] + \frac{3L}{2}\eta_g^2\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\tilde{\Delta}_i^t - \overline{\Delta}_i^t\right\|^2\right] \end{aligned} \tag{16}$$

When $\eta_g\eta_l \leq \min\{\frac{P}{96Q^2}, \frac{P}{6QL(P-1)}\}$ and $\eta_l \leq \frac{1}{\sqrt{60}QL}$, the above inequality simplifies to

$$\begin{aligned}\mathbb{E}[f(x_{t+1})] \leq & f(x_t) - \frac{\eta_g\eta_l\overline{\alpha}^t Q}{4}\|\nabla f(x_t)\|^2 \\ & + \frac{5\eta_g\eta_l^3\overline{\alpha}^t}{2}(1+\frac{12\eta_l\eta_g}{P})L^2Q^2(\sigma_l^2+6Q\sigma_g^2) \\ & + \eta_g\left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \Delta_i^t - \tilde{\Delta}_i^t\right]\right\rangle + \eta_g\left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \tilde{\Delta}_i^t - \overline{\Delta}_i^t\right]\right\rangle \\ & + \frac{3L}{N}\eta_g^2\eta_l^2(\overline{\alpha}^t)^2Q\sigma_l^2 + \frac{L}{2}\eta_g^2\frac{1}{P}\sigma^2 d \\ & + \frac{3L}{2}\eta_g^2\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\Delta_i^t - \tilde{\Delta}_i^t\right\|^2\right] + \frac{3L}{2}\eta_g^2\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\tilde{\Delta}_i^t - \overline{\Delta}_i^t\right\|^2\right] \end{aligned} \tag{17}$$

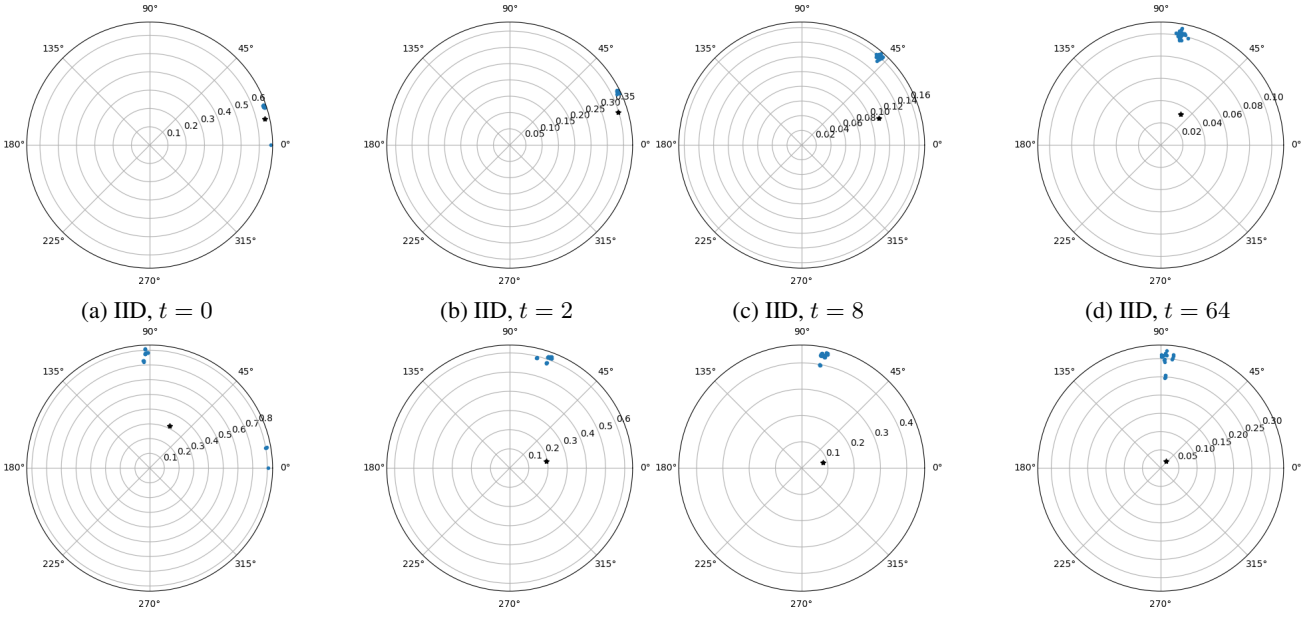
Sum over t from 1 to T, divide both sides by $T\eta_g\eta_l Q/4$, and rearrange, we have

$$\begin{aligned} & \frac{1}{T}\sum_{t=1}^T \mathbb{E}[\overline{\alpha}^t\|\nabla f(x_t)\|^2] \\ \leq & \frac{4}{T\eta_g\eta_l Q}(\mathbb{E}[f(x_1)] - \mathbb{E}[f(x_{T+1})]) \\ & + 10\eta_l^2L^2Q(1+\frac{12\eta_l\eta_g}{P})(\sigma_l^2+6Q\sigma_g^2)\frac{1}{T}\sum_{t=1}^T\overline{\alpha}^t + \frac{12L}{P}\eta_g\eta_l\sigma_l^2\frac{1}{T}\sum_{t=1}^T(\overline{\alpha}^t)^2 + 2L\frac{\eta_g}{\eta_l QP}d\sigma^2 \\ & + \frac{1}{T}\sum_{t=1}^T\frac{4}{\eta_l Q}\mathbb{E}\left[\left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \Delta_i^t - \tilde{\Delta}_i^t\right]\right\rangle + \left\langle \nabla f(x_t), \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N \tilde{\Delta}_i^t - \overline{\Delta}_i^t\right]\right\rangle\right] \\ & + \frac{6L}{\eta_l Q}\eta_g\frac{1}{T}\sum_{t=1}^T\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\Delta_i^t - \tilde{\Delta}_i^t\right\|^2\right] + \frac{6L}{\eta_l Q}\eta_g\frac{1}{T}\sum_{t=1}^T\mathbb{E}\left[\left\|\frac{1}{P}\sum_{i\in\mathcal{P}_t}\tilde{\Delta}_i^t - \overline{\Delta}_i^t\right\|^2\right]. \end{aligned} \tag{18}$$

Upper-bounding the last four terms using $\|g_i^{t,q}\| \leq G$ yields the desired result. □

A.2. Additional Numerical Experiments

In this section, we provide additional numerical results which cannot be placed in the main paper due to page limitation.



(a) IID, $t = 0$ (b) IID, $t = 2$ (c) IID, $t = 8$ (d) IID, $t = 64$

(e) Non-IID, $t = 0$ (f) Non-IID, $t = 2$ (g) Non-IID, $t = 8$ (h) Non-IID, $t = 64$

Figure 6: The distribution of local updates for MLP on IID and Non-IID data at different communication rounds for EMNIST dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of global model update at iteration t.

We run the algorithm using AlexNet (Krizhevsky et al., 2012) and ResNet-18 (He et al., 2016) with EMNIST dataset (Cohen et al., 2017) and Cifar-10 dataset (Krizhevsky et al., 2009) for comparison. In addition to the image classification problem, we also run the algorithm using the stacked LSTM model used in (Reddi et al., 2021) with Shakespeare dataset (Caldas et al., 2018) on the NLP problem.

We plot the change of the distributions of the update differences of different algorithms listed in the main paper. Notice that in all models and datasets, the distributions of the magnitude in the IID cases are more concentrated than the corresponding Non-IID cases. Also, the distributions of the same model trained on EMNIST dataset are more concentrated than trained on Cifar-10 dataset.

A.3. Quadratic Example

A.3.1. Proof of Claim 2.1

Given a fixed clipping threshold c, consider the following quadratic problem

$$f(x) = \sum_{i=1}^{3} \frac{1}{2}(x - b_i)^2,$$

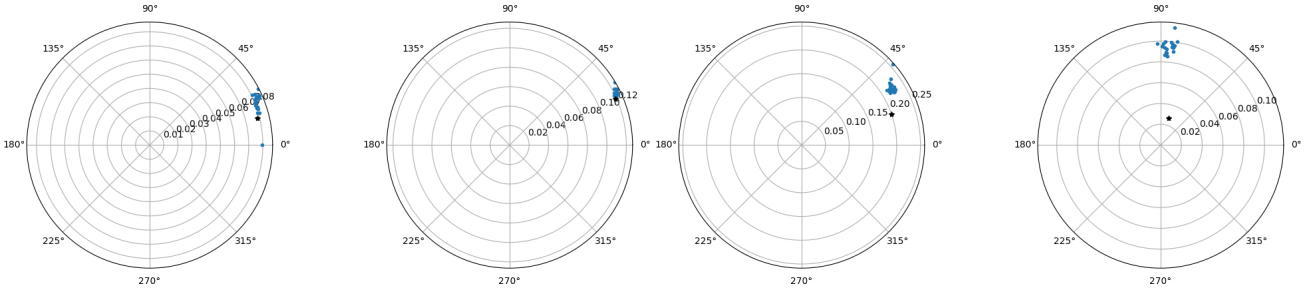
where we have $N = 3$ clients. By applying model clipping to FedAvg, one round update can be expressed as:

$$\begin{aligned} x^+ &= \frac{1}{3}\sum_{i=1}^{3} \text{clip}(\lambda x + (1-\lambda)b_i, c), \\ \lambda &= (1-\eta_l)^Q \in (0,1), \end{aligned} \tag{19}$$

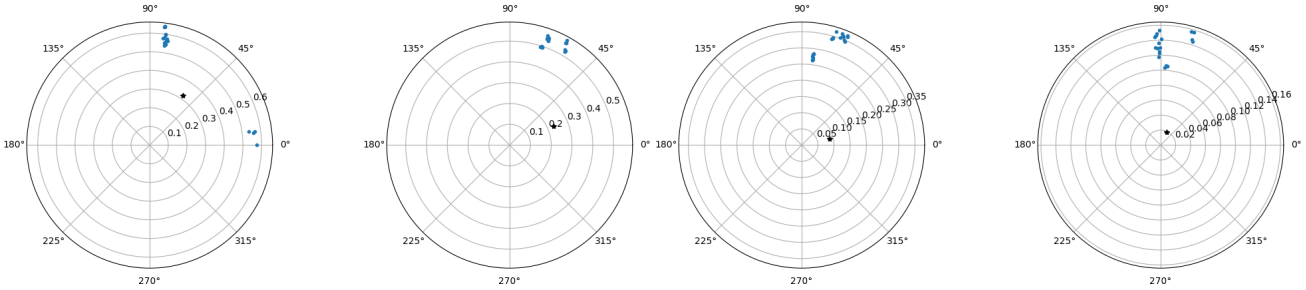
where η_l is the local stepsize.

Suppose that the algorithm converges, then we will have solution $x^+ = x = x^\infty$. This implies that

$$\frac{1}{3}\sum_{i=1}^{3} \text{clip}(\lambda x^\infty + (1-\lambda)b_i, c) = x^\infty. \tag{20}$$

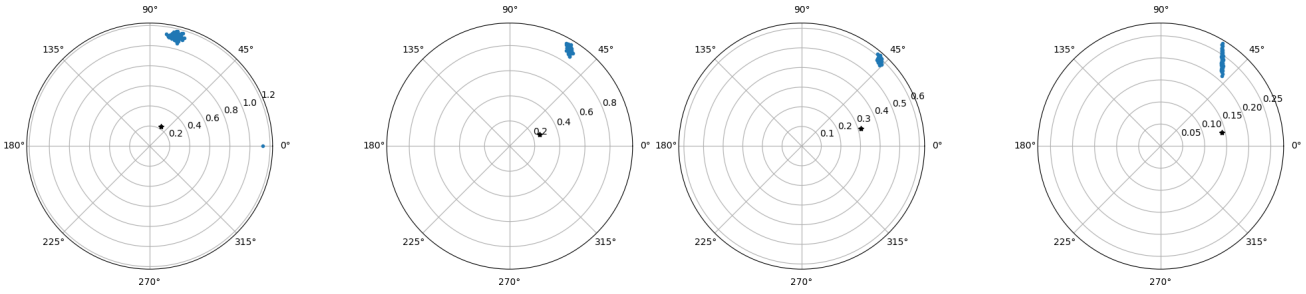


(a) IID, $t = 0$ (b) IID, $t = 2$ (c) IID, $t = 8$ (d) IID, $t = 64$

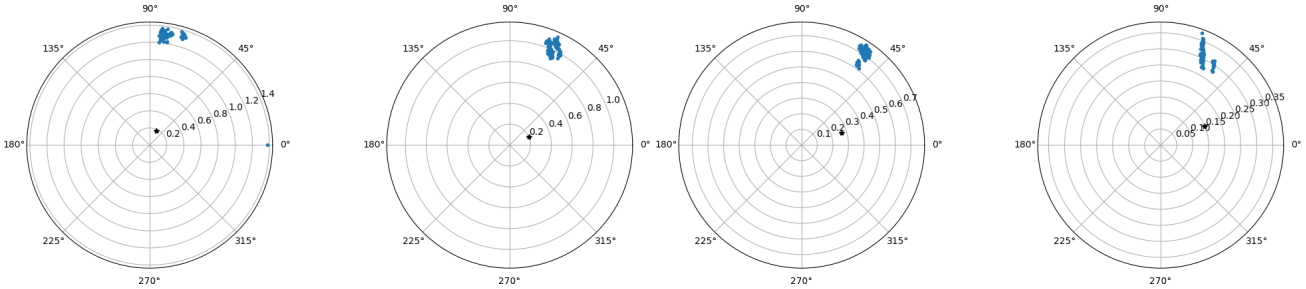


(e) Non-IID, $t = 0$ (f) Non-IID, $t = 2$ (g) Non-IID, $t = 8$ (h) Non-IID, $t = 64$

Figure 7: The distribution of local updates for AlexNet on IID and Non-IID data at different communication rounds for EMNIST dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of global local model update at iteration t.



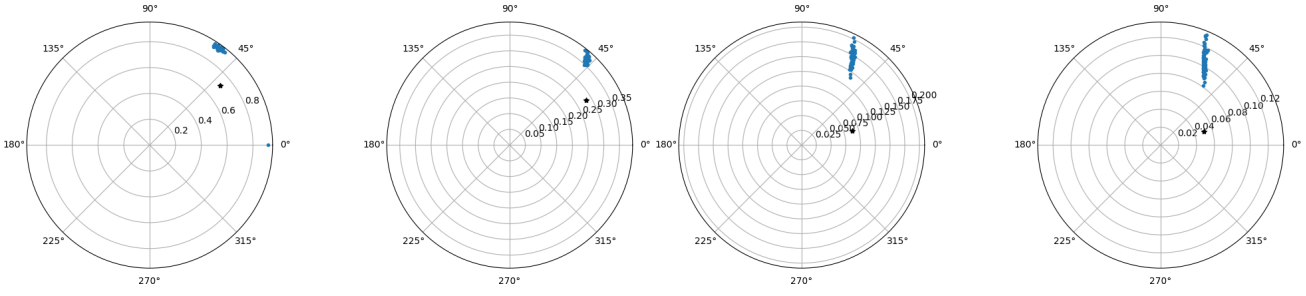
(a) IID, $t = 0$ (b) IID, $t = 2$ (c) IID, $t = 8$ (d) IID, $t = 32$



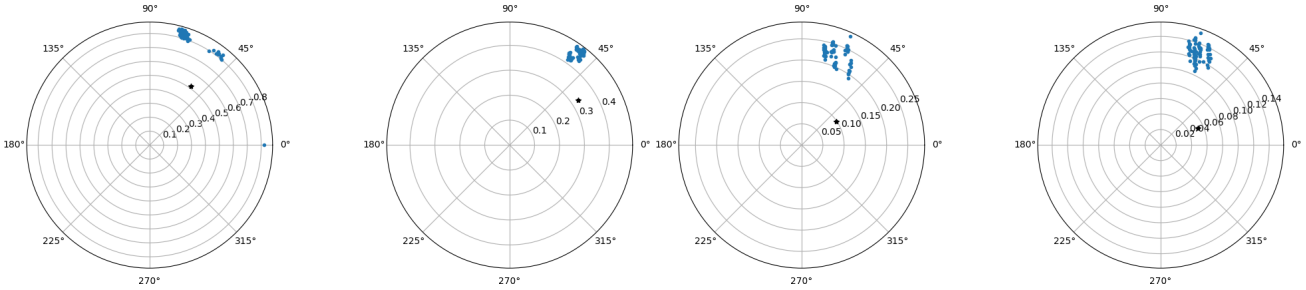
(e) Non-IID, $t = 0$ (f) Non-IID, $t = 2$ (g) Non-IID, $t = 8$ (h) Non-IID, $t = 32$

Figure 8: The distribution of local updates for MobileNetV2 on IID and Non-IID data at different communication rounds for EMNIST dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of global local model update at iteration t.

Let us set $b_1 = b_2 = -0.5c, b_3 = kc$, then it is easy to verify that the optimal solution of the problem is given by $x^\star = \frac{(k-1)c}{3} > 0$. However, when $k > 4$, from (20) we can see that $x^\infty \leq c$ and $x^\star > c$. Therefore, the only possibility is that $x^\infty = \frac{\lambda}{3-2\lambda} c \leq c \neq x^\star$, and this holds true for any $\lambda \in (0, 1)$. So the stationary solution of FedAvg with model clipping to this problem will not converge to the original optimal solution no matter how we choose Q and η_l.

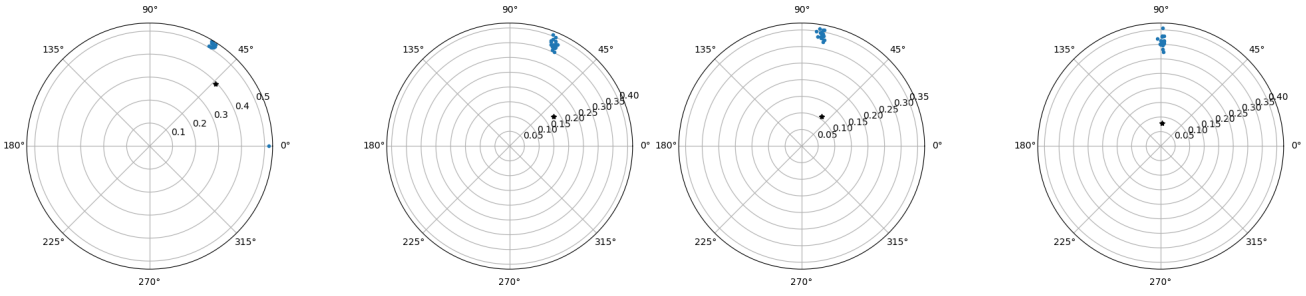


(a) IID, $t = 0$ (b) IID, $t = 2$ (c) IID, $t = 8$ (d) IID, $t = 32$

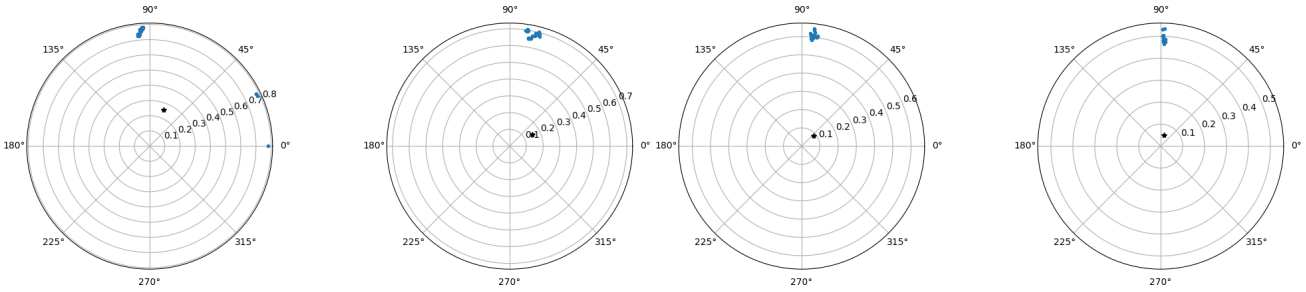


(e) Non-IID, $t = 0$ (f) Non-IID, $t = 2$ (g) Non-IID, $t = 8$ (h) Non-IID, $t = 32$

Figure 9: The distribution of local updates for ResNet-18 on IID and Non-IID data at different communication rounds for EMNIST dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of global local model update at iteration t.

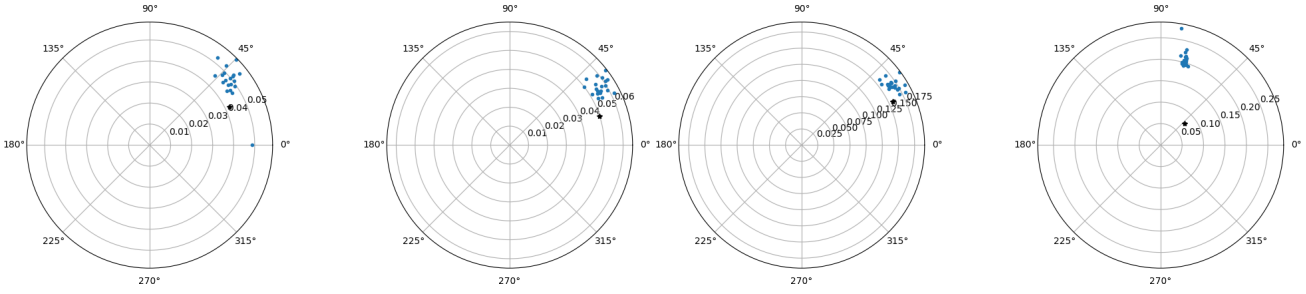


(a) IID, $t = 0$ (b) IID, $t = 2$ (c) IID, $t = 8$ (d) IID, $t = 64$

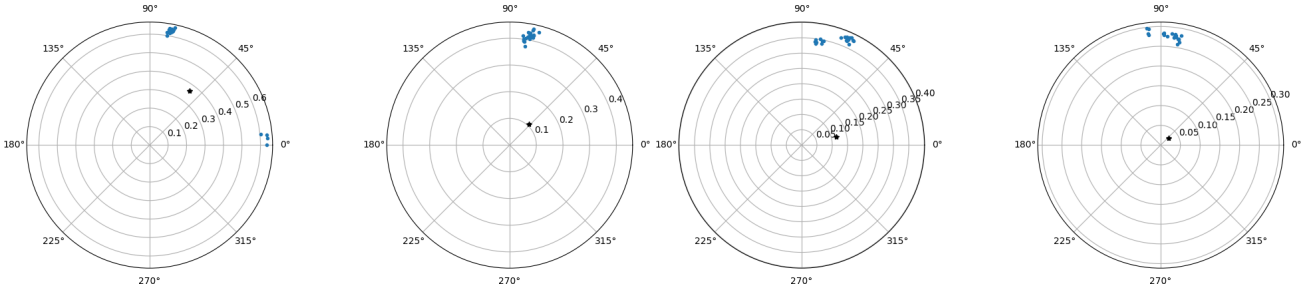


(e) Non-IID, $t = 0$ (f) Non-IID, $t = 2$ (g) Non-IID, $t = 8$ (h) Non-IID, $t = 64$

Figure 10: The distribution of local updates for MLP on IID and Non-IID data at different communication rounds for Cifar-10 dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of global local model update at iteration t.

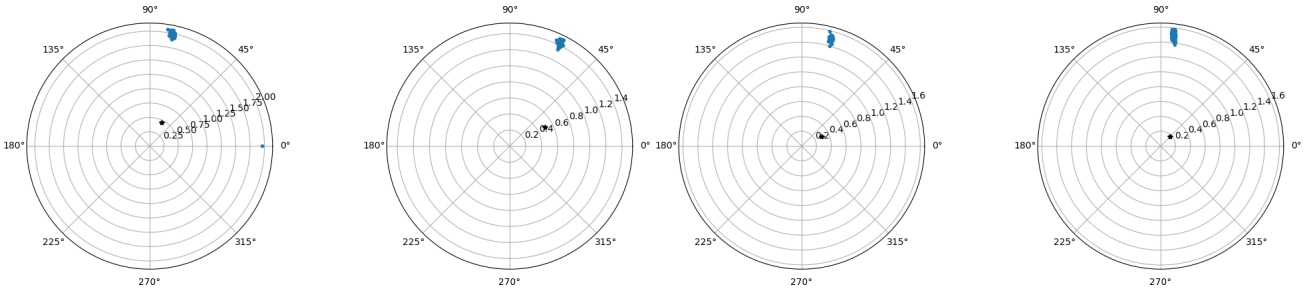


(a) IID, $t = 0$ (b) IID, $t = 2$ (c) IID, $t = 8$ (d) IID, $t = 64$

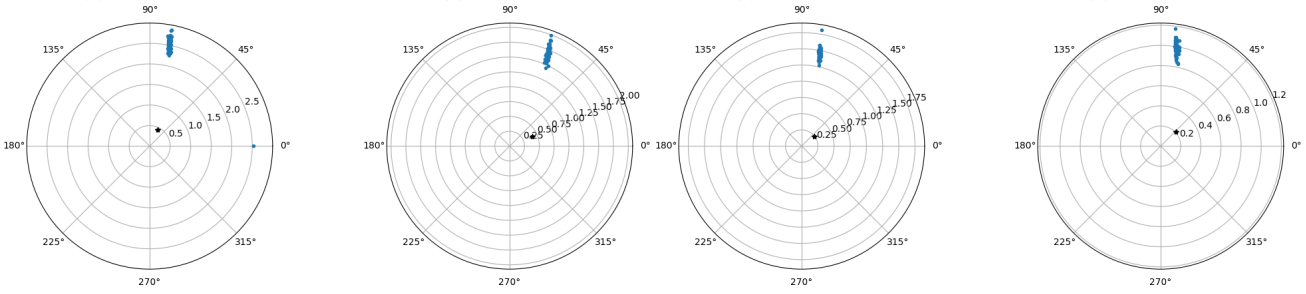


(e) Non-IID, $t = 0$ (f) Non-IID, $t = 2$ (g) Non-IID, $t = 8$ (h) Non-IID, $t = 64$

Figure 11: The distribution of local updates for AlexNet on IID and Non-IID data at different communication rounds for Cifar-10 dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of global local model update at iteration t.



(a) IID, $t = 0$ (b) IID, $t = 2$ (c) IID, $t = 8$ (d) IID, $t = 32$



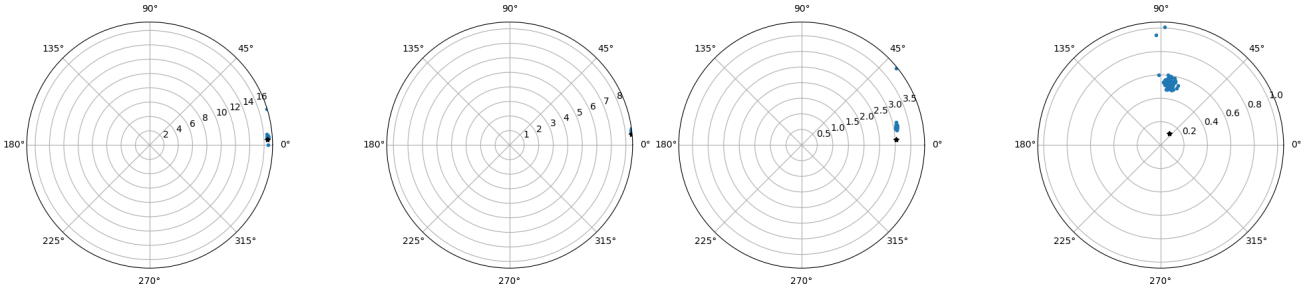
(e) Non-IID, $t = 0$ (f) Non-IID, $t = 2$ (g) Non-IID, $t = 8$ (h) Non-IID, $t = 32$

Figure 12: The distribution of local updates for ResNet-18 on IID and Non-IID data at different communication rounds for Cifar-10 dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of global local model update at iteration t.

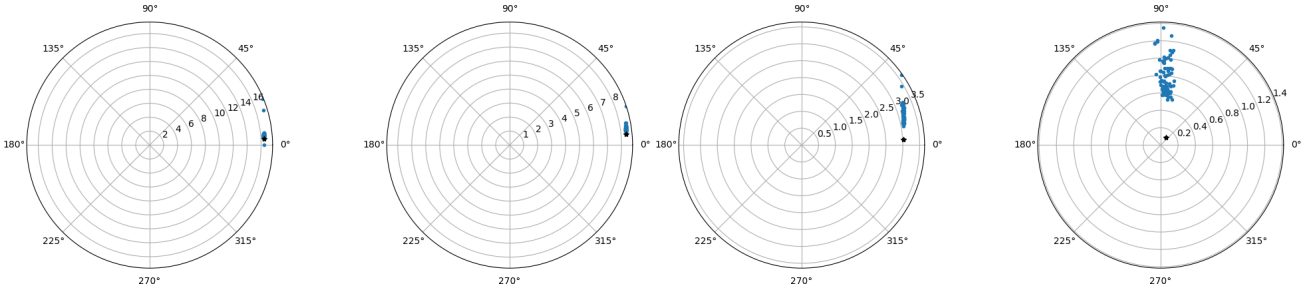
A.3.2. Proof of Claim 2.2

First, we prove that using difference clipping, FedAvg can converge to global optimal by carefully selecting Q and η_l. Consider the following convex quadratic problem

$$f(x) = \sum_{i=1}^{N} \frac{1}{2}(A_i x - b_i)^2.$$



(a) IID, $t = 0$ (b) IID, $t = 2$ (c) IID, $t = 8$ (d) IID, $t = 64$



(e) Non-IID, $t = 0$ (f) Non-IID, $t = 2$ (g) Non-IID, $t = 8$ (h) Non-IID, $t = 64$

Figure 13: The distribution of local updates for Stacked LSTM on IID and Non-IID data at different communication rounds for Shakespeare dataset. Each blue dot corresponds to the local update from one client. The black dot shows the magnitude and the cosine angle of global local model update at iteration t.

By applying FedAvg with update difference clipping, one round of update can be expressed as:

$$\begin{aligned} x^+ &= x - \frac{1}{N}\sum_{i=1}^{N} \text{clip}(\Lambda_i \nabla f_i(x), c) \\ \Lambda_i &= (I - (I - \eta_l A_i^T A_i)^Q)(A_i^T A_i)^{-1}. \end{aligned} \tag{21}$$

In order for the problem to converge to the original problem, it is easy to verify that the following condition has to hold:

$$\sum_{i=1}^{N} \text{clip}(\Lambda_i \nabla f_i(x^\star), c) = 0.$$

The above example can be viewed as using gradient descent to optimize a problem with the following gradient

$$\nabla f_i'(x) = \begin{cases} \Lambda_i \nabla f_i(x) & \|\Lambda_i \nabla f_i(x)\| \le c, \\ \frac{c\Lambda_i \nabla f_i(x)}{\|\Lambda_i \nabla f_i(x)\|} & \text{otherwise.} \end{cases} \tag{22}$$

Note that in general it is hard to write down the exact local problems f_i' that satisfies the above condition, but when $x \in \mathbb{R}$ is a scalar, $f_i'(x)$ is the Huberized loss of $\Lambda_i f_i(x)$ (Song et al., 2021)

$$f_i'(x) = \begin{cases} \Lambda_i f_i(x) & \text{if } |\Lambda_i A_i (A_i x - b_i)| \le c, \\ c\left|\frac{\Lambda_i}{A_i} f_i(x)\right| - \frac{1}{2}c^2 & \text{otherwise.} \end{cases} \tag{23}$$

In general, the re-weighted problem does not have the same solution as the original problem, but we can select η_l and Q (determined by on $x^\star$ and f_i's) so that $f'(x)$ has the same solution as $f(x)$. For example, one set of parameters that satisfy the above requirement is $Q = 1, \eta_l = 1/\max_i\{\|\nabla f_i(x^\star)\|\}$. In this case, $\Lambda_i = I\eta_l$, and when η_l is small enough, the clipping will not be activate when $x = x^\star$ and $\sum_{i=1}^N \text{clip}(\Lambda_i \nabla f_i(x^\star), c) = \sum_{i=1}^N \eta_l \nabla f_i(x^\star) = 0$.

Next, we show that Clipping-enabled FedAvg can outperform the non-clipped version. Note that when $Q > 1$, even when η is small such that the clipping is not activated, the algorithm will not converge to the original solution. So in general

	$Q=1$	$Q=\infty$
$c=\infty$	$x^\infty=0$	$x^\infty=\frac{13}{9}$
$c=1$	$x^\infty=\frac{1}{2}$	$x^\infty=\frac{2}{3}$

Table 4: Stationary points of FedAvg with gradient clipping for (24) under different parameter settings.

one cannot draw the conclusion about whether clipping helps or hurts the performance of FedAvg. Consider the following problem:

$$\begin{aligned} f(x) &= \sum_{i=1}^{3} f_i(x), \\ f_1(x) &= \frac{1}{2}(x-4)^2,\ f_2(x) = \frac{1}{2}(2x-1)^2,\ f_3(x) = \frac{1}{2}(6x+1)^2. \end{aligned} \tag{24}$$

As $\nabla f(x) = (x-4) + (4x-2) + (36x+6) = 41x$, the optimal solution of this problem is $x^\star = 0$. Table 4 show the stationary points of FedAvg under different choice of parameters. When $Q=1$, FedAvg is equivalent to SGD and clipping hurts the performance of FedAvg. However, when Q is large, clipped FedAvg has a better performance than the non-clipped version, in the sense that the stationary solution it obtains are closer to the global optimal solution $x^* = 0$.