EFFICIENT PRIVATE FEDERATED NON-CONVEX OPTIMIZATION WITH SHUFFLED MODEL

Anonymous authors
Paper under double-blind review

ABSTRACT

This paper studies the problem of distributed non-convex optimization under privacy requirements. We develop a differentially private communication efficient algorithm and study its privacy and utility trade-offs. By introducing the shuffled model into our algorithmic design, we are able to achieve strong privacy and utility guarantees without relying on a trusted central server. We further show that our proposed method can achieve improved utility guarantees (faster convergence rates) compared to previous approaches. Additionally, we present preliminary experimental results to corroborate our theoretical findings.

1 INTRODUCTION

We consider the following distributed optimization problem with M clients:

$$\min_{x\in\mathbb{R}^d} F(x) := \frac{1}{M}\sum_{m=1}^{M} F_m(x) := \frac{1}{Mn}\sum_{m=1}^{M}\sum_{i=1}^{n} f(x; z_i^m), \tag{1.1}$$

where F_m is the objective for client m with its own local dataset $D_m = \{z_1^m, \ldots, z_n^m\}$. Our goal is to find a model parameter x that can minimize the problem in equation 1.1 while acheiving differential privacy (DP) (see Definition 2.4) for each dataset D_m, where $m \in [M]$. We are interested in the non-convex heterogeneous case, where f is non-convex and clients have different data distributions.

To solve the distributed optimization problem in equation 1.1, we consider the **intermittent communication** (IC) setting (Stich, 2018; Dieuleveut & Patel, 2019; Woodworth et al., 2021; Bullins et al., 2021; Patel et al., 2022), where M clients work in parallel over R communication rounds, and each client can sequentially compute T stochastic gradient estimates between two communication rounds. In particular, we are interested in the federated learning (FL) framework (McMahan et al., 2016; Kairouz et al., 2021b), where there exists a **central server** that will communicate with clients at each communication round to allow the information sharing among clients.

We want to achieve DP for each individual data record in D_m for client $m \in [M]$, i.e., **record-level DP**. This is in contrast to many previous works (McMahan et al., 2018; Kairouz et al., 2021a) for FL that try to achieve DP for the whole data records in D_m, i.e., **client-level DP**. The client-level DP is useful for cross-device FL, where each device/client (such as mobile phone) maintains data records from a single individual. However, the record-level DP is useful for cross-silo FL, where each silo/client (e.g., hospital) has data records from many different individuals (e.g., patients), and it has been considered in many previous works (Murata & Suzuki, 2023; Lowy et al., 2023; Girgis et al., 2021) for FL. In addition, we assume the central server is **not trustworthy** and thus we want to protect against the curious central server. This is motivated by several findings (Boenisch et al., 2023) highlighting the significant privacy leakage issues caused by the central server in FL.

Many attempts have been made to deal with the **untrusted server**. For example, one can use the secure aggregation techniques, such as multi-party computation (MPC), in FL (Jayaraman et al., 2018). However, these techniques often suffer from huge computational and memory costs, particularly in scenarios involving a large number of clients, extensive data records, and significant model sizes. Another straightforward way is to achieve local DP (LDP) (Beimel et al., 2008; Duchi et al., 2013) for each client. Nevertheless, LDP mechanisms often lead to poor privacy and utility trade-offs (Duchi et al., 2013). A recent line of study (Cheu et al., 2019; Balle et al., 2019; Feldman et al., 2023) shows that one can significantly improve the privacy and utility trade-offs of LDP

Table 1: Comparison of convergence rates for different (ϵ, δ)-DP algorithms in the IC setting. The results are presented by ignoring numerical constants and the logarithmic dependence. τ is the heterogeneity (see Assumption 2.3) of the problem and $\tau \leq 2L$. b is the batch size used by local update algorithms for their local updates, K is the number of local steps. * DIFF2 requires $Mn \geq G^2 d^{1/2}/(LD_F\epsilon)$, which implies $(dD_FLG)^{2/3}/(Mn\epsilon)^{4/3} \geq dG^2/(Mn\epsilon)^2$. Red terms indicate additional errors introduced by using mini-batch gradients instead of full gradients at the server and $\ell = Mn\epsilon/(RdMKb)^{1/2}$. † SDP FEDPROX-SPIDER has an extra $\log(R/\delta)$ dependency compared our method due to the advanced composition used in its privacy guarantees (Abadi et al., 2016).

Method (Reference)	Convergence Rate $\mathbb{E}\|\nabla F(\widehat{x})\|_2^2 \leq$	Trusted Server	Full Gradients at Server
DIFF2-GD* (Murata & Suzuki, 2023)	$\frac{D_F L}{R} + \frac{D_F L\sqrt{d}}{Mn\epsilon\sqrt{R}} + \frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}}$	Yes	Yes
MB-PSGM-FG† Theorem A.4 SDP FEDPROX-SPIDER† (Lowy et al., 2023)	$\frac{D_F L}{R} + \frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}} + \frac{dG^2}{M^2n^2\epsilon^2}$	No	Yes
MB-PSGM Theorem A.3	$\frac{D_F L}{R} + \frac{(D_F LG)^{2/3}}{(MKb)^{2/3}} + \frac{G^2}{MKbR}$ $+\big(1+\ell+\ell^{1/3}\big)\frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}} + \frac{dG^2}{M^2n^2\epsilon^2}$	No	No
DIFF2-BVR-LSGD* (Murata & Suzuki, 2023)	$\frac{D_F \tau}{R} + \frac{D_F L}{RK} + \frac{D_F L}{R\sqrt{Kb}}$ $+\frac{D_F L\sqrt{d}}{R\sqrt{K\epsilon b}} + \frac{D_F L\sqrt{d}}{\sqrt{RMn\epsilon}} + \frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}}$	Yes	Yes
CE-PSGM-FG Theorem A.2	$\frac{D_F \tau}{R} + \frac{D_F L}{RK} + \frac{D_F L}{R\sqrt{Kb}}$ $+\frac{D_F L\sqrt{d}}{R\sqrt{K\epsilon b}} + \frac{D_F L\sqrt{d}}{\sqrt{RMn\epsilon}} + \frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}} + \frac{dG^2}{M^2n^2\epsilon^2}$	No	Yes
CE-PSGM Theorem 4.2	$\frac{D_F \tau}{R} + \frac{D_F L}{RK} + \frac{D_F L}{R\sqrt{Kb}} + \frac{(D_F LG)^{2/3}}{(MKb)^{2/3}} + \frac{G^2}{MKbR}$ $+\frac{D_F L\sqrt{d}}{R\sqrt{K\epsilon b}} + \frac{D_F L\sqrt{d}}{\sqrt{RMn\epsilon}}$ $+\big(1+\ell+\ell^{1/3}\big)\frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}} + \frac{dG^2}{M^2n^2\epsilon^2}$	No	No

mechanisms through the **shuffled model** by amplifying privacy guarantees through anonymization (the privacy guarantee will be amplified by a factor of $1/\sqrt{M}$). Therefore, we propose to follow the idea of a shuffling model such that clients will first send their local updates to a secure shuffler. The shuffler will then randomly permute clients' updates and send the shuffled messages to the server. See Figure 2 in Appendix A.4 for a simple illustration. Equipped with a secure random shuffler, we can achieve strong privacy and utility trade-offs and handle the untrusted central server.

Related work. Recently, lots of efforts (Noble et al., 2022; Liu et al., 2021; Li et al., 2022; Lowy & Razaviyayn, 2022; Murata & Suzuki, 2023; Lowy et al., 2023; Girgis et al., 2021) have been made towards achieving DP in FL. The works most closely related to ours are those by Murata & Suzuki (2023); Lowy et al. (2023), which also consider solving the problem in equation 1.1 within the non-convex setting while achieving record-level DP. For example, Murata & Suzuki (2023) proposes two differentially private algorithms, including one mini-batch algorithm (i.e., DIFF2-GD) and one local update algorithm (i.e., DIFF2-BVR-LSGD), based on the non-private BVR-L-SGD algorithm (Murata & Suzuki, 2021). For mini-batch algorithms, each machine computes the stochastic/mini-batch gradient estimates at the same point (model parameter) for T times to generate a large mini-batch gradient estimate. On the other hand, in the local update algorithm, each machine computes stochastic/mini-batch gradient estimates at different points for each local step and uses them to update the local model. They provide the convergence guarantees of their

methods, showing that the local update algorithm can achieve a faster convergence rate than the mini-batch algorithm when the problem heterogeneity (see Assumption 2.3) is low. However, their approaches not only require a trustworthy central server but also ask clients to send full gradients to the server for constructing a global differentially private gradient estimator. Lowy et al. (2023) establishes a differentially private mini-batch algorithm, namely SDP FEDPROX-SPIDER, which builds upon on the non-private SPIDER algorithm (Fang et al., 2018). Although their method can handle the untrusted central server through the shuffle model, it fails to benefit from the problem's low heterogeneity and requires full gradient computations from the clients.

Contributions. The contributions of our work are summarized as follows.

- We develop a differentially private algorithm CE-PSGM for solving the distributed non-convex optimization problem with heterogeneous data. At the core of our algorithm is the non-private communication efficient local update algorithm (Patel et al., 2022) and the shuffled model (Feldman et al., 2023). By using a secure random shuffler, our method is able to address the scenario where the central server is untrustworthy. Additionally, our approach can seamlessly reduce to the mini-batch algorithm (i.e., MB-PSGM), which is widely used in practice, by omitting local updates.
- We show the convergence guarantees of our proposed method. Specifically, our mini-batch algorithm achieves a faster convergence rate than existing mini-batch algorithms (Murata & Suzuki, 2023; Lowy et al., 2023). Furthermore, for the local update algorithm, our method attains the state-of-the-art convergence rate (Murata & Suzuki, 2023). More importantly, we illustrate that the local update algorithm can converge faster than the mini-batch algorithm when the problem heterogeneity is low. Detailed comparisons are provided in Table 1.
- Compared to existing methods (Murata & Suzuki, 2023; Lowy et al., 2023), our proposed method allows clients to transmit mini-batch gradients, as opposed to full gradients, to the central server for constructing the global differentially private gradient estimator, while also achieving fast convergence rates. This advancement is made possible through our new privacy amplification results (Lemma 2.7), which take into account the combined effects of data sampling and random shuffling.

Notation. We use $\mathcal{B}$ to denote the index set. $[n]$ denotes the set $\{1, 2, \ldots, n\}$. We use $\lesssim$ to denote inequality up to numerical constants and poly-logarithmic terms, and let $D_F = F(x_0) - \inf_{x\in\mathbb{R}^d} F(x)$.

2 PRELIMINARIES

Our goal is to find an ε-approximate stationary point of F, i.e., a point $x \in \mathbb{R}^d$ such that $\mathbb{E}\|\nabla F(x)\|_2^2 \leq \varepsilon$, that is also DP. We have the following assumptions on the objective loss functions.

Assumption 2.1. $f(x; z)$ is G-Lipschitz, i.e., $\forall x, y \in \mathbb{R}^d,\ |f(x; z) - f(y; z)| \leq G\|x - y\|_2$.

The Lipchitz assumption is a standard assumption in the differentially private optimization literature (), and it is necessary for us to show the convergence of our method.

Assumption 2.2. $f(x; z)$ is L-smooth, i.e., $\forall x, y \in \mathbb{R}^d,\ \|\nabla f(x; z) - \nabla f(y; z)\|_2 \leq L\|x - y\|_2$.

We make the following assumption that relate the functions of different clients to one another, which is also known as the **"heterogeneity"** of the problem.

Assumption 2.3. The objective function of each client is second-order τ-heterogeneous, i.e., $\forall m \in [M]$, $\sup_{m\in[M],x\in\mathbb{R}^d} \|\nabla^2 F_m(x) - \nabla^2 F(x)\|_2 \leq \tau$.

Assumption 2.3 can always be satified by setting $\tau \geq 2L$ for smooth functions. This second-order heterogeneity assumption has been previously considered in the non-private setting (Karimireddy et al., 2021; Murata & Suzuki, 2021; Patel et al., 2022), and is crucial to show the benifits (Patel et al., 2022) of the local updates when $\tau \ll L$.

We next introduce the notions of differential privacy (Dwork et al., 2006) and Rényi Differential Privacy (RDP) (Mironov, 2017). In our privacy analysis, we use RDP to account for privacy loss and then state our results by converting the RDP guarantee to (ϵ, δ)-DP guarantee.

Definition 2.4 ((ϵ, δ)-DP)**.** A randomized mechanism $\mathcal{M}$ satisfies (ϵ, δ)-differential privacy if for adjacent datasets D, D' differing by one element, and any output subset O, it holds that $\mathbb{P}[\mathcal{M}(D) \in O] \leq e^\epsilon \cdot \mathbb{P}[\mathcal{M}(D') \in O] + \delta$.

Definition 2.5 (RDP). A randomized mechanism $\mathcal{M}$ satisfies (α, ρ)-Rényi differential privacy with $\alpha > 1$ and $\rho > 0$ if for adjacent datasets $D, D' \in \mathcal{D}$ differing by one element, $D_\alpha\big(\mathcal{M}(D)||\mathcal{M}(D')\big) = \log \mathbb{E}_{\mathcal{M}(D')}\big(\mathcal{M}(D)/\mathcal{M}(D')\big)^\alpha/(1-\alpha) \leq \rho$.

We can use the following Gaussian Mechanism (Mironov, 2017) to acheive RDP.

Lemma 2.6. Given a function q, the Gaussian Mechanism $\mathcal{M} = q(D) + \mathbf{u}$, where $\mathbf{u} \sim N(0, \sigma^2\mathbf{I})$, satisfies $(\alpha, \alpha S_2^2/(2\sigma^2))$-RDP, where S_2 is the ℓ_2-sensitivity of q and is defined as $S_2 = \sup_{D,D'} \|q(D) - q(D')\|_2$ for two adjacent datasets D, D' differing by one element.

In our algorithmic design, we propose to use both data sampling and random shuffling (see line 5 to line 12 in Algorithm 1) at r-th communication round to construct the global differentially private gradient estimator, and we denote this process by $\mathcal{M}_r$ (see detailed definition in Appendix A.3). Therefore, we provide the following new privacy amplification result that captures the combined effect of data sampling and shuffling on the privacy guarantees.

Lemma 2.7. Let $\gamma = B/n$ and M_L is an ϵ_0-LDP mechanism. Then the mechanism $\mathcal{M}_r$ is $(\alpha, 26\gamma^2\alpha\rho)$-RDP with $\rho = ce_0^\epsilon/(MB)$ for all $\alpha \in [2, MB/(4c\epsilon_0 e^{\epsilon_0})]$ and some constant c when $\gamma \leq 0.1$, $\rho \leq 0.1$, $\epsilon_0 \geq 1$, $c \geq 16\epsilon_0 \log(1/\gamma)$, $6ce^{\epsilon_0} \leq MB\log(1/\delta)$.

3 METHODS

The proposed algorithm, i.e., CE-PSGM, is illustrated in Algorithm 1.

Algorithm 1 Communication Efficient Private Stochastic Gradient Method (CE-PSGM)

input Initialization x_0, iteration number R, step size η, local steps H, batch sizes B_0, B_1, B_2, weight parameters β, clipping parameters C_1, C_2, C_3, noise parameter σ^2, LDP parameter ϵ_0

1: Let $x_0 = x_{-1}$, $\bar{v}_{-1} = 0$
2: **for** $r = 0, 1, \ldots, R-1$ **do**
3: **if** $r = 0$ set $\rho = 1$, $Q = 1$, $B = B_0$ **else** set $\rho = \beta$, $Q = H$, $B = B_1$
4: **Send (communicate)** (x_r, x_{r-1}) to clients
5: **on client** $m \in [M]$ **do**
6: Data sampling: choose B samples uniformly at random indexed by $\mathcal{B}_r^m$
7: Compute stochastic gradients: $g_{m,r}^{1,i} = \nabla f(x_r; z_i^m) - \nabla f(x_{r-1}; z_i^m)$ and $g_{m,r}^{2,i} = \nabla f(x_{r-1}; z_i^m)$, where $i \in \mathcal{B}_r^m$
8: **LDP estimators:** $\bar{g}_{m,r}^{1,i} = \mathcal{M}_L\big(\text{CLIP}(g_{m,r}^{1,i}, C_r^1)\big)$ and $\bar{g}_{m,r}^{2,i} = \mathcal{M}_L\big(\text{CLIP}(g_{m,r}^{2,i}, C_r^2)\big)$, where $C_r^1 = C_1\|x_r - x_{r-1}\|_2$, $C_r^2 = C_2$
9: **Send (communicate)** $\big(\{\bar{g}_{m,r}^{1,i}\}_{i\in\mathcal{B}_r^m}, \{\bar{g}_{m,r}^{2,i}\}_{i\in\mathcal{B}_r^m}\big)$ to the shuffler
10: **end on client**
11: **shuffling:** Shuffler randomly shuffles $\{\bar{g}_{m,r}^{1,i}\}_{i\in\mathcal{B}_r^m, m\in[M]}$ and $\{\bar{g}_{m,r}^{2,i}\}_{i\in\mathcal{B}_r^m, m\in[M]}$ and **Send (communicate)** them to the server
12: **Private VR term:** $\bar{v}_r = (1-\rho)\bar{v}_{r-1} + \frac{1}{MB}\sum_{m=1}^M\sum_{i=1}^B\big((1-\rho)\bar{g}_{m,r}^{1,i} + \rho\bar{g}_{m,r}^{2,i}\big)$
13: **Send (communicate)** $(x_r, \bar{v}_r)$ to client $\widetilde{m}_r$, where $\widetilde{m}_r$ is $(r \bmod M + 1)$-th client
14: **on client** $\widetilde{m}_r$ **do**
15: Set: $w_{r+1,1}^{\widetilde{m}_r} := w_{r+1,0}^{\widetilde{m}_r} := x_r$, $\bar{v}_{r,0}^{\widetilde{m}_r} := \bar{v}_r$
16: **for** $k = 1, \ldots, H$ **do**
17: Data sampling: choose B_2 samples uniformly at random indexed by $\mathcal{B}_{r,k}^{\widetilde{m}}$
18: $d_{k-1}^{\widetilde{m}} = \frac{1}{B_2}\sum_{i\in\mathcal{B}_{r,k}^{\widetilde{m}}}\text{CLIP}(\nabla f(w_{r+1,k}^{\widetilde{m}_r}; z_i^{\widetilde{m}}) - \nabla f(w_{r+1,k-1}^{\widetilde{m}_r}; z_i^{\widetilde{m}}), C_r^k)$, where $C_r^k = C_3\|w_{r+1,k}^{\widetilde{m}_r} - w_{r+1,k-1}^{\widetilde{m}_r}\|_2$
19: **Private gradient estimator:** $\bar{v}_{r,k}^{\widetilde{m}_r} = \bar{v}_{r,k-1}^{\widetilde{m}_r} + d_{k-1}^{\widetilde{m}} + \nu$, where $\nu \sim C_r^k \cdot N(0, \sigma^2\mathbf{I})$
20: Update: $w_{r+1,k+1}^{\widetilde{m}_r} = w_{r+1,k}^{\widetilde{m}_r} - \eta\bar{v}_{r,k}^{\widetilde{m}_r}$
21: **end for**
22: **Send (communicate)** $\big(w_{r+1,H+1}^{\widetilde{m}_r}\big)$ to the server
23: **end on client**
24: Let $x_{r+1} = w_{r+1,H+1}^{\widetilde{m}_r}$
25: **end for**

output Choose $\widetilde{x}$ uniformly from $\{w_{r,k}^{\widetilde{m}_r}\}_{r\in[R],k\in[H]}$

Our method is motivated by the communication efficient local update algorithm (Patel et al., 2022). At the central server, we construct a private global gradient estimator $\bar{v}_r$ (see line 5 to line 12) through a secure shuffler. Specifically, each client $m \in [M]$ first computes stochastic gradients

$\{g_{m,r}^{1,i}, g_{m,r}^{2,i}\}_{i\in\mathcal{B}_r^m}$ for a random subset $\mathcal{B}_r^m$ of $B_1 \leq n$ samples (line 7). Then, the client applies the ϵ_0-LDP mechanism $\mathcal{M}_L$ to these stochastic gradients (line 8). Each client sends the LDP stochastic gradients $\{\bar{g}_{m,r}^{1,i}, \bar{g}_{m,r}^{2,i}\}_{i\in\mathcal{B}_r^m}$ to the secure shuffler. The shuffler outputs a random permutation of the received gradients and sends them to the server (line 11). Finally, the server will aggregate the received gradients to construct $\bar{v}_r$ (line 12), a private variant of the STORM gradient estimator (Cutkosky & Orabona, 2019). For the local updates, we propose to use the private gradient estimator $\bar{v}_{r,k}^{m_r}$ (line 19), which is constructed by adding random Gaussian noise to the SARAH gradient estimator (Nguyen et al., 2017). Given these two private gradient estimators, we are able to make use of the low heterogeneity of the problem to achieve a faster convergence rate (Patel et al., 2022). In addition, the secure random shuffler allows us to not only achieve strong privacy guarantees through anonymization (Feldman et al., 2023) but also deal with the untrustworthy central server. For the ϵ_0-LDP mechanism $\mathcal{M}_L$, we propose to use the method proposed by Duchi et al. (2018) for the private mean estimation problem. The detailed algorithm can be found in Appendix A.4.

If we want to implement our algorithm in the IC setting with parameter T, we can set $H = K, B_2 = b, B_1 = Kb$ in Algorithm 1, and let $Kb = T$. In addition, if we set $H = 1$, our method reduces to the mini-batch algorithm (MB-PSGM), i.e., the algorithm without local updates on the clients. By setting $B_1 = Kb$, we can also implement MB-PSGM in the IC setting with parameter T. Note that if we are using full gradients to construct the private gradient estimator $\bar{v}_r$ at the central server, i.e., $B_1 = n > T$, we can divide the full gradient computation into $\lceil n/T \rceil$ rounds of communications. We assume these settings of parameters to present the theoretical results of our methods.

4 Main results

In this section, we present the privacy and utility (convergence) guarantees of our methods.

Theorem 4.1 (Privacy of CE-PSGM)**.** If we set local steps $H = K$, batch sizes $B_0 = b_0, B_1 = Kb, B_2 = b$, noise parameter $\sigma^2 = \mathcal{O}\big(\max\big\{\frac{KR\log(1/\delta)}{Mn^2\epsilon^2}, \frac{\log(1/\delta)}{b^2\epsilon}\big\}\big)$, then under conditions that $\epsilon_0 = \mathcal{O}(1)$, $\epsilon = \mathcal{O}(\log(1/\delta))$, $Mn\min\{\gamma_0, \gamma_1\} = \Omega(\log(1/\delta)/\epsilon)$, where $\gamma_0 = b_0/n$, $\gamma_1 = Kb/n$, Algorithm 1 is (ϵ, δ)-DP with $\epsilon = \mathcal{O}\big(\sqrt{(\gamma_0 + R\gamma_1)\log(1/\delta)/(Mn)}\big)$.

Note that the privacy budget ϵ is restricted to be the order of $\sqrt{\gamma_1 R/Mn}$ due to the technical requirements of amplification by shuffling results (Feldman et al., 2023). This is also the case in the previous work (Girgis et al., 2021) of using the shuffle model. This restriction is mild since we can choose a larger R to achieve the privacy guarantee in the low privacy regime.

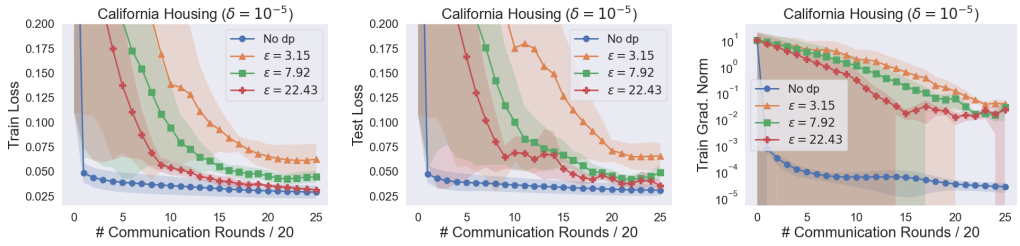
We can also use full gradients to construct the global private gradient estimator $\bar{v}_r$ (line 5-12 in Algorithm 1), as in the previous works (Murata & Suzuki, 2023; Lowy et al., 2023), and we denote this algorithm as CE-PSGD-FG. According to Theorem 4.1, we only need to set $B_0 = B_1 = n, \gamma_0 = \gamma_1 = 1$ to achieve its privacy guarantees. As we mentioned before, if we set $H = 1$, then Algorithm 1 reduces to the mini-batch algorithm and we denote it as MB-PSGM. The privacy guarantees of MB-PSGM can be found in Appendix A.1.

Theorem 4.2 (Utility of CE-PSGM)**.** Suppose f satifies Assumptions 2.1-2.3, and we choose the same parameters $H, B_0, B_1, B_2, \sigma, \epsilon_0$ as in Theorem 4.1. If we set $C_1 = C_2 = L, C_2 = G$, $\beta = \max\Big\{\frac{1}{R}, \min\Big\{\frac{(Mn\epsilon)^{2/3}(D_F L)^{2/3}}{d^{1/3}RG^{4/3}}, \frac{(MKb)^{1/3}(D_F L)^{2/3}}{R^{2/3}G^{4/3}}\Big\}\Big\}$, $\eta = \min\Big\{\frac{\sqrt{\beta Mb}}{\sqrt{K}L}, \frac{\sqrt{\beta}Mn\epsilon}{\sqrt{dR}KL}, \frac{1}{L}, \frac{1}{K\tau}, \frac{\sqrt{b}}{\sqrt{K}L}, \frac{1}{\sqrt{KdL}\sigma}\Big\}$, and assuming $1/(Rb_0) \leq \beta^2/(Kb)$, $\beta \leq 1$, $\epsilon = \mathcal{O}(\log(1/\delta))$, $M\min\{b_0, Kb\} = \Omega(\log(1/\delta)/\epsilon)$, $Kb < n, b_0 < n$, then Algorithm 1 satisfies

$$\mathbb{E}\big\|\nabla F(\widetilde{x})\big\|_2^2 \lesssim \underbrace{\frac{D_F\tau}{R} + \frac{D_F L}{RK} + \frac{D_F L}{R\sqrt{Kb}} + \frac{(D_F LG)^{2/3}}{(MKb)^{2/3}} + \frac{G^2}{MKbR}}_{\text{non-private error}} + \underbrace{\frac{D_F L\sqrt{d}}{R\sqrt{K\epsilon b}} + \frac{D_F L\sqrt{d}}{\sqrt{RMn\epsilon}}}_{\text{private error local}}$$
$$+ \underbrace{\Big(1 + \ell + \ell^{1/3}\Big)\frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}} + \frac{dG^2}{M^2n^2\epsilon^2}}_{\text{private error server}},$$

where $\ell = Mn\epsilon/\sqrt{RdMKb}$.

According to Theorem 4.2, the utility guarantee consists of three parts. The first term, non-private error, represents the optimization error. The second term, private error local, corresponds to the



(a) Train loss (b) Test loss (c) Gradient norm

Figure 1: Numerical results of our proposed MB-PSGM-FG algorithm under different privacy budgets. We can see that our method can achieve reasonable performance even in the high privacy regime.

error introduced by the Gaussian mechanism during the local training. The third term, private error server, is due to the private mechanisms for constructing the global private gradient estimator $\bar{v}_r$ at the server. If $\epsilon \to \infty$, i.e., there is no privacy guarantee, this result reduces to the non-private result (Patel et al., 2022) by replacing the stochastic gradient variance with G^2.

Compared to previous works (Murata & Suzuki, 2023; Lowy et al., 2023), our method enables the use of mini-batch gradients, as opposed to full gradients, for constructing $\bar{v}_r$ at the server. This will cost additional errors $(D_F LG)^{2/3}/(MKb)^{2/3} + G^2/(MKbR)$ in the non-private error term, and introduce additional $\ell + \ell^{1/3}$ factor in the private error server term. The detailed utility guarantees of using full gradient for $\bar{v}_r$ can be found in Appendix A.1, and we refer it as CE-PSGM-FG.

The detailed comparisons of different algorithms are summarized in Table1. Firstly, setting $H = 1$ enables us to derive the utility guarantees (refer to Appendix A.1) for the mini-batch algorithms MB-PSGM and MB-PSGM-FG (utilizing full gradients for $\bar{v}_r$). Compared to the mini-batch algorithm DIFF2-GD (Murata & Suzuki, 2023), the MB-PSGM-FG method attains a faster convergence rate by eliminating the additional $D_F L\sqrt{d}/(Mn\epsilon\sqrt{R})$ term. Additionally, MB-PSGM-FG converges faster than SDP FEDPROX-SPIDER (Lowy et al., 2023), which suffers from an extra $\log(R/\delta)$ dependency due to the advanced composition used in its privacy guarantees (Abadi et al., 2016). Secondly, the convergence rate of our local update algorithm CE-PSGM-FG is on par with the state-of-the-art local update method DIFF2-BVR-LSGD (Murata & Suzuki, 2023), without the need for a trustworthy server. Lastly, the local update algorithm proves more efficient than the mini-batch algorithm when the heterogeneity parameter τ is small and the number of local steps K is large, provided R is sufficiently large.

5 NUMERICAL RESULTS

In this section, we present preliminary numerical results to evaluate our proposed method. Figure 1 illustrates the performance of MB-PSGM-FG method under different privacy guarantees. We can see from Figure 1 that our method can achieve reasonable performances even under high privacy regime. More details about the datasets, model architectures and algorithm parameters can be found in Appendix A.2.

6 CONCLUSION AND FUTURE WORK

In this paper, we develop a differentially private communication efficient algorithm for solving the distributed non-convex optimization problem under privacy constraints. We show that our proposed method can achieve faster convergence rates than the previous methods without relying on the trusted server. We also present preliminary experimental results to evaluate the performance of our method.

As for the future work, we plan to conduct more experiments to thoroughly evaluate the performances of our methods. We also plan to study the optimality of our algorithms.

REFERENCES

Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, pp. 308–318, 2016.

Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. The privacy blanket of the shuffle model. In *Advances in Cryptology–CRYPTO 2019: 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2019, Proceedings, Part II 39*, pp. 638–667. Springer, 2019.

Amos Beimel, Kobbi Nissim, and Eran Omri. Distributed private data analysis: Simultaneously solving how and what. In *Advances in Cryptology–CRYPTO 2008: 28th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 17-21, 2008. Proceedings 28*, pp. 451–468. Springer, 2008.

Franziska Boenisch, Adam Dziedzic, Roei Schuster, Ali Shahin Shamsabadi, Ilia Shumailov, and Nicolas Papernot. When the curious abandon honesty: Federated learning is not private. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)*, pp. 175–199. IEEE, 2023.

Brian Bullins, Kshitij Patel, Ohad Shamir, Nathan Srebro, and Blake E Woodworth. A stochastic newton algorithm for distributed convex optimization. *Advances in Neural Information Processing Systems*, 34, 2021.

Mark Bun, Cynthia Dwork, Guy N Rothblum, and Thomas Steinke. Composable and versatile privacy via truncated cdp. In *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*, pp. 74–86, 2018.

Albert Cheu, Adam Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed differential privacy via shuffling. In *Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19–23, 2019, Proceedings, Part I 38*, pp. 375–403. Springer, 2019.

Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex sgd. *Advances in neural information processing systems*, 32, 2019.

Aymeric Dieuleveut and Kumar Kshitij Patel. Communication trade-offs for local-sgd with large step size. *Advances in Neural Information Processing Systems*, 32, 2019.

John C Duchi, Michael I Jordan, and Martin J Wainwright. Local privacy and statistical minimax rates. In *2013 IEEE 54th Annual Symposium on Foundations of Computer Science*, pp. 429–438. IEEE, 2013.

John C Duchi, Michael I Jordan, and Martin J Wainwright. Minimax optimal procedures for locally private estimation. *Journal of the American Statistical Association*, 113(521):182–201, 2018.

Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of Cryptography: Third Theory of Cryptography Conference, TCC 2006, New York, NY, USA, March 4-7, 2006. Proceedings 3*, pp. 265–284. Springer, 2006.

Cong Fang, Chris Junchi Li, Zhouchen Lin, and Tong Zhang. Spider: Near-optimal non-convex optimization via stochastic path-integrated differential estimator. *Advances in Neural Information Processing Systems*, 31, 2018.

Vitaly Feldman, Audra McMillan, and Kunal Talwar. Stronger privacy amplification by shuffling for rényi and approximate differential privacy. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pp. 4966–4981. SIAM, 2023.

Antonious Girgis, Deepesh Data, Suhas Diggavi, Peter Kairouz, and Ananda Theertha Suresh. Shuffled model of differential privacy in federated learning. In *International Conference on Artificial Intelligence and Statistics*, pp. 2521–2529. PMLR, 2021.

Bargav Jayaraman, Lingxiao Wang, David Evans, and Quanquan Gu. Distributed learning without distress: Privacy-preserving empirical risk minimization. *Advances in Neural Information Processing Systems*, 31, 2018.

Peter Kairouz, Brendan McMahan, Shuang Song, Om Thakkar, Abhradeep Thakurta, and Zheng Xu. Practical and private (deep) learning without sampling or shuffling. In *International Conference on Machine Learning*, pp. 5213–5225. PMLR, 2021a.

Peter Kairouz, H Brendan McMahan, Brendan Avent, Aurélien Bellet, Mehdi Bennis, Arjun Nitin Bhagoji, Kallista Bonawitz, Zachary Charles, Graham Cormode, Rachel Cummings, et al. Advances and open problems in federated learning. *Foundations and Trends® in Machine Learning*, 14(1–2):1–210, 2021b.

Sai Praneeth Karimireddy, Martin Jaggi, Satyen Kale, Mehryar Mohri, Sashank J. Reddi, Sebastian U Stich, and Ananda Theertha Suresh. Mime: Mimicking centralized stochastic algorithms in federated learning, 2021. URL https://openreview.net/forum?id=MJmYbFnJAGa.

Lihua Lei, Cheng Ju, Jianbo Chen, and Michael I Jordan. Non-convex finite-sum optimization via scsg methods. In *NeurIPS*, 2017.

Zhize Li, Haoyu Zhao, Boyue Li, and Yuejie Chi. Soteriafl: A unified framework for private federated learning with communication compression. *Advances in Neural Information Processing Systems*, 35:4285–4300, 2022.

Ruixuan Liu, Yang Cao, Hong Chen, Ruoyang Guo, and Masatoshi Yoshikawa. Flame: Differentially private federated learning in the shuffle model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 8688–8696, 2021.

Andrew Lowy and Meisam Razaviyayn. Private federated learning without a trusted server: Optimal algorithms for convex losses. In *The Eleventh International Conference on Learning Representations*, 2022.

Andrew Lowy, Ali Ghafelebashi, and Meisam Razaviyayn. Private non-convex federated learning without a trusted server. In *International Conference on Artificial Intelligence and Statistics*, pp. 5749–5786. PMLR, 2023.

H Brendan McMahan, Eider Moore, Daniel Ramage, S Hampson, and B Agüera y Arcas. Communication-efficient learning of deep networks from decentralized data (2016). *arXiv preprint arXiv:1602.05629*, 2016.

H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. In *International Conference on Learning Representations*, 2018.

Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pp. 263–275. IEEE, 2017.

Tomoya Murata and Taiji Suzuki. Bias-variance reduced local sgd for less heterogeneous federated learning. In *International Conference on Machine Learning*, pp. 7872–7881. PMLR, 2021.

Tomoya Murata and Taiji Suzuki. Diff2: Differential private optimization via gradient differences for nonconvex distributed learning. *arXiv preprint arXiv:2302.03884*, 2023.

Lam M Nguyen, Jie Liu, Katya Scheinberg, and Martin Takáč. Sarah: A novel method for machine learning problems using stochastic recursive gradient. In *International Conference on Machine Learning*, pp. 2613–2621. PMLR, 2017.

Maxence Noble, Aurélien Bellet, and Aymeric Dieuleveut. Differentially private federated learning on heterogeneous data. In *International Conference on Artificial Intelligence and Statistics*, pp. 10110–10145. PMLR, 2022.

Kumar Kshitij Patel, Lingxiao Wang, Blake E Woodworth, Brian Bullins, and Nati Srebro. Towards optimal communication complexity in distributed non-convex optimization. *Advances in Neural Information Processing Systems*, 35:13316–13328, 2022.

Sebastian U Stich. Local sgd converges fast and communicates little. *arXiv preprint arXiv:1805.09767*, 2018.

Blake E Woodworth, Brian Bullins, Ohad Shamir, and Nathan Srebro. The min-max complexity of distributed stochastic convex optimization with intermittent communication. In *Conference on Learning Theory*, pp. 4386–4437. PMLR, 2021.

A APPENDIX

A.1 ADDITIONAL RESULTS

If we set the local steps $H = 1$, then Algorithm 1 reduces to the mini-batch algorithm, i.e., MB-PSGM. We have the following privacy guarantee of MB-PSGM.

Theorem A.1 (Privacy of MB-PSGM)**.** If we set local steps $H = 1$, $B_0 = b_0, B_1 = Kb$, then Algorithm 1 is (ϵ, δ)-DP for any $\delta > 0$ with

$$\epsilon = \mathcal{O}\left(\sqrt{\frac{(\gamma_0 + R\gamma_1)\log(1/\delta)}{Mn}}\right),$$

where $\gamma_0 = b_0/n$, $\gamma_1 = Kb/n$ and under the conditions that $\epsilon_0 = \mathcal{O}(1)$, $\epsilon = \mathcal{O}(\log(1/\delta))$, $Mn\min\{\gamma_0, \gamma_1\} = \Omega(\log(1/\delta)/\epsilon)$.

Similarly, we can set $\gamma_0 = \gamma_1 = 1$ to obtain the privacy guarantee of our mini-batch method using full gradients, i.e., MB-PSGM-FG.

Note that if we use full gradients to construct $\bar{v}_r$ at the server, we have the following utility guarantee.

Theorem A.2 (Utility of CE-PSGM-FG)**.** Suppose f satifies Assumptions 2.1-2.3, we choose $B_0 = B_1 = n$, and set parameters $H, B_2, \sigma, \epsilon_0$ the same as in Theorem 4.1. If we set clipping paramters $C_1 = C_2 = L, C_2 = G$, $\beta = \max\left\{\frac{1}{R}, \frac{(Mn\epsilon)^{2/3}(D_F L)^{2/3}}{d^{1/3}RG^{4/3}}\right\}$, step size $\eta = \min\left\{\frac{\sqrt{\beta}Mn\epsilon}{\sqrt{dR}KL}, \frac{1}{L}, \frac{1}{K\tau}, \frac{\sqrt{b}}{\sqrt{K}L}, \frac{1}{\sqrt{Kd}L\sigma}\right\}$, and assuming $\beta \leq 1$, $\epsilon = \mathcal{O}(\log(1/\delta))$, $Mn = \Omega(\log(1/\delta)/\epsilon)$, then the output $\widetilde{x}$ of Algorithm 1 satisfies

$$\mathbb{E}\|\nabla F(\widetilde{x})\|_2^2 \lesssim \underbrace{\frac{D_F\tau}{R} + \frac{D_F L}{RK} + \frac{D_F L}{R\sqrt{Kb}}}_{\text{non-private error}} + \underbrace{\frac{D_F L\sqrt{d}}{R\sqrt{K\epsilon}b} + \frac{D_F L\sqrt{d}}{\sqrt{RM}n\epsilon}}_{\text{private error local}} + \underbrace{\frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}} + \frac{dG^2}{M^2n^2\epsilon^2}}_{\text{private error server}}.$$

As we mentioned before, if we choose the local steps $H = 1$, Algorithm 1 reduces to MB-PSGM. We have the following utility guarantee for MB-PSGM.

Theorem A.3 (Utility of MB-PSGM)**.** Suppose f satifies Assumptions 2.1, 2.2, and we choose the same parameters $H, B_0, B_1, B_2, \epsilon_0$ as in Theorem A.1. If we set clipping paramters as $C_1 = L, C_2 = G$, $\beta = \max\left\{\frac{1}{R}, \min\left\{\frac{(Mn\epsilon)^{2/3}(D_F L)^{2/3}}{d^{1/3}RG^{4/3}}, \frac{(MKb)^{1/3}(D_F L)^{2/3}}{R^{2/3}G^{4/3}}\right\}\right\}$, stepsize $\eta = \min\left\{\frac{\sqrt{\beta M b}}{\sqrt{K}L}, \frac{\sqrt{\beta}Mn\epsilon}{\sqrt{dR}KL}, \frac{1}{L}\right\}$, and assuming $1/(Rb_0) \leq \beta^2/(Kb)$, $\beta \leq 1$, $\epsilon = \mathcal{O}(\log(1/\delta))$, $M\min\{b_0, Kb\} = \Omega(\log(1/\delta)/\epsilon)$, then the output $\widetilde{x}$ of Algorithm 1 satisfies

$$\mathbb{E}\left\|\nabla F(\widetilde{x})\right\|_2^2 \lesssim \underbrace{\frac{D_F L}{R} + \frac{(D_F LG)^{2/3}}{(MKb)^{2/3}} + \frac{G^2}{MKbR}}_{\text{non-private error}} + \underbrace{\left(1 + \left(\frac{Mn\epsilon}{(RdMKb)^{1/2}}\right)^2 + \left(\frac{Mn\epsilon}{(RdMKb)^{1/2}}\right)^{1/3}\right)\frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}} + \frac{dG^2}{M^2n^2\epsilon^2}}_{\text{private error server}}.$$

We can also provide the utility guarantee for the minibatch method using full gradients to construct $\bar{v}_r$ as follows.

Theorem A.4 (Utility of MB-PSGM-FG)**.** Suppose f satifies Assumptions 2.1, 2.2, we choose $B_0 = B_1 = n$, and set parameters H, ϵ_0 the same as in Theorem A.1. If we set clipping paramters $C_1 = L, C_2 = G$, $\beta = \max\left\{\frac{1}{R}, \frac{(Mn\epsilon)^{2/3}(D_F L)^{2/3}}{d^{1/3}RG^{4/3}}\right\}$, stepsize $\eta = \min\left\{\frac{\sqrt{\beta}Mn\epsilon}{\sqrt{dR}KL}, \frac{1}{L}\right\}$, and assuming $\beta \leq 1$, $\epsilon = \mathcal{O}(\log(1/\delta))$, $Mn = \Omega(\log(1/\delta)/\epsilon)$, then the output $\widetilde{x}$ of Algorithm 1 satisfies

$$\mathbb{E}\|\nabla F(\widetilde{x})\|_2^2 \lesssim \underbrace{\frac{D_F L}{R}}_{\text{non-private error}} + \underbrace{\frac{(dD_F LG)^{2/3}}{(Mn\epsilon)^{4/3}} + \frac{dG^2}{M^2n^2\epsilon^2}}_{\text{private error server}}.$$

A.2 EXPERIMENTS

We consider the regression task to evaluate our proposed method MB-PSGM-FG on the California Housing dataset [1]. Following the previous work (Murata & Suzuki, 2023), we randomly divide the dataset into 80% train dataset and 20% test dataset, resulting in a train dataset of size 16512 and a test dataset of size 4128. We then randomly split the train dataset into 10 subsets and assigned each of them to one of 10 clients.

We use the same neural network as in Murata & Suzuki (2023). We set the communication rounds as $R = 500$ with $\epsilon \in \{0.4, 0.8, 1.6\}$, $\delta = 10^{-5}$. We choose the clipping parameter for MB-PSGM-FG from $\{1, 3, 10, 30, 100\}$. We tuned the learning rate by setting it to 0.5 originally. Our algorithm checked the train loss every 20 communication rounds. If all 500 communication rounds finish without the train loss increasing to over 1.05 times its previous value 3 times in a row, the learning rate is finalized. Otherwise, we halve the learning rate and restart training. This process is repeated until the model completes R communication rounds successfully.

We evaluate our method using 3 metrics: train loss, squared train gradient norm, and test loss. We plotted the mean and standard deviation of these metrics over the 5 repeated runs.

A.3 PROOF OF PRIVACY GUARANTEES

First of all, we present several useful lemmas that will be uses to prove our main results.

Given a privacy guarantee in terms of RDP, we can transfer it to (ϵ, δ)-DP using the following lemma.

Lemma A.5. (Mironov, 2017) If a randomized mechanism $\mathcal{M}$ satisfies (α, ρ)-RDP, then $\mathcal{M}$ satisfies $(\rho + \log(1/\delta)/(\alpha - 1), \delta)$-DP for all $\delta \in (0, 1)$.

Recall that, according to Lemma 2.6, we can use Gaussian mechanism to achieve RDP. In addition, we have the following privacy amplification by subsampling result for the Gaussian mechanism.

Lemma A.6. (Bun et al., 2018) Let $\mathcal{M}$ be an Gaussian mechanism that takes a dataset of $n \leq N$ examples as an input and $\gamma = n/N$ is the sampling rate. If $\mathcal{M}$ is $(\alpha, \alpha S_2^2/(2\sigma^2))$-RDP, then when we apply $\mathcal{M}$ on a subsampled dataset, which consists of γN examples sampled without replacement from the input dataset with size N, it is $(\alpha, 6\gamma^2\alpha S_2^2/\sigma^2)$-RDP provided that $\gamma \leq 0.1$, $\sigma/S_2 \geq \sqrt{5}$, and $\alpha \leq \sigma^2 \log(1/\gamma)/(2S_2^2)$.

We have the following composition result for RDP.

Lemma A.7 (Mironov (2017))**.** If k randomized mechanisms $\mathcal{M}_i$ for $i \in [k]$, satisfy (α, ρ_i)-RDP, then their composition $\big(\mathcal{M}_1(S), \ldots, \mathcal{M}_k(S)\big)$ satisfies $(\alpha, \sum_{i=1}^k \rho_i)$-RDP. Moreover, the input of the i-th mechanism can base on the outputs of previous $(i-1)$ mechanisms.

We have the following privacy amplification by shuffling result.

Lemma A.8. (Feldman et al., 2023) For any domain $\mathcal{D}$, let $\mathcal{M}_i : O_1 \times \cdots \times O_{i-1} \times \mathcal{D} \to O_i$ for $i \in [N]$ be a sequence of algorithms such that $\mathcal{M}_i(o_{1:i-1}, \cdot)$ is an ϵ_0-DP local randomizer for auxiliary inputs $o_{1:i-1} \in O_1 \times \cdots O_{i-1}$. Let $\mathcal{A} : \mathcal{D}^N \to O_1 \times \cdots \times O_N$ be the algorithm that given a dataset $z_{1:N} \in \mathcal{D}^N$, samples a uniform random permutation π over $[N]$, then sequentially computes $o_i = \mathcal{M}_i(o_{1:i-1}, z_{\pi(i)})$ for $i \in [N]$ and outputs $o_{1:n}$. Then, for $\epsilon_0 \geq 1$ and any $2 \leq \alpha \leq n/(32\epsilon_0 e^{\epsilon_0})$, $\mathcal{A}$ is $(\alpha, \alpha\rho)$-RDP, where

$$\rho = \frac{ce^{\epsilon_0}}{n},$$

where $c = 1536$.

Given these lemmas, we are ready to prove our privacy amplification by subsampling and shuffling result, which is key to provide the strong privacy and utility guarantees of our method.

Before that, following Girgis et al. (2021), let us first formally define the mechanism $\mathcal{M}_r$ for each round r, i.e., the resultant mechanism of both subsampling and shuffling that corresponds to lines 5-12 of Algorithm 1. To be more specific, recall that each client $m \in [M]$ has its local data

[1] https://www.dcc.fc.up.pt/ ltorgo/Regression/cal$_h ousing.html$

$D_m = \{z_1^m, \ldots, z_n^m\}$ and let $\mathcal{D} = \cup_{m=1}^M D_m$ denote the entire dataset. Fix a round r, we let $\mathcal{M}_r^1(x_r, x_{r-1}, \mathcal{D})$ be the private mechanism at round r that takes the entire dataset $\mathcal{D}$, auxiliary inputs (x_r, x_{r-1}) and generates the private term $\frac{1}{MB}\sum_{m=1}^M \sum_{i=1}^B (1-\rho)\bar{g}_{m,r}^{1,i}$ at the server. In particular, according to Algorithm 1, we have

$$\mathcal{M}_r^1(x_r, x_{r-1}, \mathcal{D}) := \mathcal{H}_{MB} \circ (\mathcal{G}_1^1, \ldots, \mathcal{G}_M^1), \tag{A.1}$$

where $\mathcal{G}_m^1 = \mathrm{samp}_{n,B}(\mathcal{M}_L(d_{m,r}^{1,1}), \ldots, \mathcal{M}_L(d_{m,r}^{1,n}))$ and $d_{m,r}^{1,i} := \mathrm{CLIP}(g_{m,r}^{1,i}, C_r^1), \forall m \in [M]$ and $\forall i \in [n]$. Here $\mathcal{H}_{MB}$ denotes the shuffling operation on MB elements and $\mathrm{samp}_{n,B}$ denotes the subsampling operation that chooses a random subset of B elements from a set of n elements. Note that since the same mechanism $\mathcal{M}_L$ is applied, we have the output distribution of first applying subsampling and then private mechanism $\mathcal{M}_L$ is equal to that of first applying private mechanism $\mathcal{M}_L$ and then subsampling. Similarly, we let $\mathcal{M}_r^2(x_{r-1}, \mathcal{D})$ be the private mechanism at round r that takes the entire dataset $\mathcal{D}$, auxiliary input x_{r-1} and generates the private term $\frac{1}{MB}\sum_{m=1}^M \sum_{i=1}^B \rho\bar{g}_{m,r}^{1,i}$ at the server. In particular, according to Algorithm 1, we have

$$\mathcal{M}_r^2(x_{r-1}, \mathcal{D}) := \mathcal{H}_{MB} \circ (\mathcal{G}_1^2, \ldots, \mathcal{G}_M^2),$$

where $\mathcal{G}_m^2 = \mathrm{samp}_{n,B}(\mathcal{M}_L(d_{m,r}^{2,1}), \ldots, \mathcal{M}_L(d_{m,r}^{2,n}))$ and $d_{m,r}^{2,i} := \mathrm{CLIP}(g_{m,r}^{2,i}, C_r^2), \forall m \in [M]$ and $\forall i \in [n]$.

Finally, let the composite mechanism

$$\mathcal{M}_r(x_r, x_{r-1}, \mathcal{D}) = (\mathcal{M}_r^1(x_r, x_{r-1}, \mathcal{D}), \mathcal{M}_r^2(x_{r-1}, \mathcal{D})) \tag{A.2}$$

be the private mechanism that takes the entire dataset $\mathcal{D}$, auxiliary input (x_r, x_{r-1}) and generates the private VR term $\bar{v}_r$ at the server. That is, $\mathcal{M}_r$ can be viewed as a transformation of $\mathcal{M}_L$ via both subsampling and shuffling, both of which can be used to amplify privacy. Thus, one natural question is whether these two amplification effects can be combined to yield an even stronger amplification of privacy. This is answered affirmatively by the following lemma.

Lemma A.9 (Restate of Lemma 2.7)**.** Let $\gamma = B/n$, and M_L is an ϵ_0-LDP mechanism. Then the mechanism $\mathcal{M}_r$ defined in equation A.2 is $(\alpha, 26\gamma^2\alpha\rho)$-RDP with $\rho = \frac{ce^{\epsilon_0}}{MB}$ and $c = 1536$ for all $\alpha \in [2, \frac{MB\log(1/\gamma)}{4ce^{\epsilon_0}}]$ when the following conditions are satisfied: (i) $\gamma \in (0, 0.1], \rho = \frac{ce^{\epsilon_0}}{MB} \leq 0.1, \epsilon_0 \geq 1$; (ii) $c \geq 16\epsilon_0 \log(1/\gamma)$; (iii) $6ce^{\epsilon_0} \leq MB\log(1/\gamma)$.

Proof of Lemma 2.7. For notation simplicity, we denote $\mathcal{M}_r(x_r, x_{r-1}, \mathcal{D})$ by $\mathcal{M}_r(\mathcal{D})$ since our follow-up analysis holds for all x_r, x_{r-1}. We will focus on $\mathcal{M}_r^1$ and the same result holds for $\mathcal{M}_r^2$. The final result for $\mathcal{M}_r$ follows from the composition result for RDP as in Lemma A.7.

As in Girgis et al. (2021), we first define $\mathcal{Z}(\mathcal{D}_r) = \mathcal{H}_{MB}(\mathcal{M}_L(d_r^1), \ldots, \mathcal{M}_L(d_r^{MB}))$, which is a shuffling of MB outputs of a local mechanism $\mathcal{M}_L$, where $\mathcal{D}_r$ is an arbitrary set of MB data points $(d_r^1, \ldots, d_r^{MB})$. Then, by Lemma A.8, the machanism $\mathcal{Z}$ is $(\alpha, \alpha\rho)$-RDP, where $2 \leq \alpha \leq MB/(32\epsilon_0 e^{\epsilon_0})$ and $\rho \leq ce^{\epsilon_0}/(MB)$ for $\epsilon_0 \geq 1$.

Now, we aim to relate our $\mathcal{M}_r^1$ in equation A.1 to the above mechanism $\mathcal{Z}$. To this end, let $\mathcal{T}_{m,r} \subseteq \{1, \ldots, n\}$ denote the random identities of the B data points chosen at client m at round r. Let $\mathcal{D}^{\mathcal{T}_{m,r}} = \{z_j^m : j \in \mathcal{T}_{m,r}\}$, i.e., the subsampled data points at client m in round r. Thus, let $\mathcal{D}^{\overline{\mathcal{T}}_r} = \cup_{m=1}^M D^{\mathcal{T}_{m,r}}$, then we can write $\mathcal{M}_r^1 = \mathcal{Z}(\mathcal{D}^{\overline{\mathcal{T}}_r})$.

Our next observation is the key step: Due to (i) independent subsampling of data points at each client and (ii) no subsampling of clients, the privacy amplification in our case basically reduces to the standard case, i.e., a single agent subsamples B points (without replacement) from a total of n data points. In other words, $\mathcal{M}_r^1$ can be viewed as first applying a subsampling with ratio B/n and then a private mechanism $\mathcal{Z}$ with RDP guarantee stated above. Thus, we can apply Theorem 13 in Bun et al. (2018) to obtain that $\mathcal{M}_r^1$ is $(\alpha, 13\gamma^2\alpha\rho)$-RDP with $\gamma = B/n$ and $\rho = \frac{ce^{\epsilon_0}}{MB}, c = 1536$ for all $\alpha \in [2, \frac{MB\log(1/\gamma)}{4ce^{\epsilon_0}}]$ when the following conditions are satisfied: (i) $\gamma \in (0, 0.1], \rho = \frac{ce^{\epsilon_0}}{MB} \leq 0.1, \epsilon_0 \geq 1$; (ii) $c \geq 16\epsilon_0 \log(1/\gamma)$; (iii) $6ce^{\epsilon_0} \leq MB\log(1/\gamma)$.

Finally, given that the same privacy guarantee holds for $\mathcal{M}_r^2$, by composition of RDP, we have that $\mathcal{M}_r$ is $(\alpha, 26\gamma^2\alpha\rho)$-RDP with $\gamma = B/n$ and $\rho = \frac{ce^{\epsilon_0}}{MB}, c = 1536$ for all $\alpha \in [2, \frac{MB\log(1/\gamma)}{4ce^{\epsilon_0}}]$ when the above conditions (i - iii) are satisfied.

□

Now, we are ready to prove the privacy gaurantees of our method.

Proof of Theorem 4.1. **Privacy guarantee of the variance reduction term at server.** Note that we need to shuffle both $\{\bar{g}_{m,r}^{1,i}\}_{i\in\mathcal{B}_r^m}$ and $\{\bar{g}_{m,r}^{2,i}\}_{i\in\mathcal{B}_r^m}$ at r-th iteration. We denote these two mechanism as $\mathcal{M}_{Ser}^{r,1}$ and $\mathcal{M}_{Ser}^{r,2}$. According to Lemma A.9, $\mathcal{M}_{Ser}^{r,1}$ and $\mathcal{M}_{Ser}^{r,2}$ are $(\alpha, \alpha\bar{\rho})$-RDP, where $\bar{\rho} = 26\gamma_1\widetilde{\rho}$ and $\gamma_1 = B_1/n$ for $r > 0$ and $\gamma_1 = B_0/n$ for $r = 0$, $\widetilde{\rho} = ce^{\epsilon_0}/(Mn)$ for $\epsilon_0 \geq 1$ and under conditions that $\rho \leq 0.1$, $\epsilon_0 \geq 1$, $2 \leq \alpha \leq MB\log(1/\gamma)/(4ce^{\epsilon_0})$, $6ce^{\epsilon_0} \leq MB\log(1/\gamma)$, $c \geq 16\epsilon_0\log(1/\gamma)$, where B, γ denotes B_0, B_1 and γ_1 for simplicity. Note that the conditions $\rho \leq 0.1, c \geq 16\epsilon_0\log(1/\gamma), 6ce^{\epsilon_0} \leq MB\log(1/\gamma)$ can be satisfied when MB is large enough.

Privacy guarantee of the local update for the selected client. According to Algorithm 1 (line 12), we use one client for the local update, then each local data set D_i for $i \in [M]$ will be used at most $\lceil R/M \rceil$ times. Furthermore, at r-th communication round, the selected client will use subsampled Gaussian mechansim (line 17-line 19), denoted by $\mathcal{M}_{loc}^{r,k}$ with sampling rate $\gamma_2 = B_2/n$ at k-th local update. Therefore, according to Lemma A.6, at r-th round, the k-th local update for the selected client is (α, ρ_2)-RDP with respected to the selected local data set, where $\rho_2 = 24\gamma_2^2\alpha/(B_2^2\sigma^2)$. In addition, we need the conditions $\gamma_2 \leq 0.1$, $B_2\sigma \geq 2\sqrt{5}$, and $\alpha \leq \sigma^2 B_2^2\log(1/\gamma_2)/2$.

Privacy guarantee of Algorithm 1. By the composition and post processing results in Lemma A.7, we have that Algorithm 1 is $(\alpha, 2\alpha\bar{\rho}_0 + 2(R-1)\alpha\bar{\rho}_1 + K\lceil R/M\rceil\rho_2)$-RDP, where $\bar{\rho}_0 = 26\gamma_0\widetilde{\rho}$, $\bar{\rho}_1 = 26\gamma_1\widetilde{\rho}$, $\rho_2 = 24\gamma_2^2\alpha/(B_2^2\sigma^2)$, $\gamma_0 = B_0/n$, $\gamma_1 = B_1/n$, $\gamma_2 = B_2/n$, and $\widetilde{\rho} = ce^{\epsilon_0}/(Mn)$. In addition, we require the conditions that $\gamma_2 \leq 0.1$, $B_2\sigma \geq 2\sqrt{5}$, and $\alpha \leq \sigma^2 B_2^2\log(1/\gamma_2)/2$, and $2 \leq \alpha \leq MB\log(1/\gamma)/(4ce^{\epsilon_0})$, $6ce^{\epsilon_0} \leq MB\log(1/\gamma)$.

Finally, by Lemma A.5, we have that Algorithm 1 is $(2\alpha\bar{\rho}_0 + 2(R-1)\alpha\bar{\rho}_1 + K\lceil R/M\rceil\rho_2 + \log(1/\delta)/(\alpha-1), \delta)$-DP. Therefore, we have

$$2\alpha\bar{\rho}_0 + 2(R-1)\alpha\bar{\rho}_1 = \frac{52c(\gamma_0 + R\gamma_1)e^{\epsilon_0}\alpha}{Mn} \leq \frac{C(\gamma_0 + R\gamma_1)\epsilon_0^2\log(1/\delta)}{Mn\epsilon},$$

where C is a constant, and the last inequality is due to the choice of $\alpha = 1 + 2\log(1/\delta)/\epsilon$, $e^{\epsilon_0} = \mathcal{O}(\epsilon_0^2)$ when $\epsilon_0 = \mathcal{O}(1)$, and $\epsilon = \mathcal{O}(\log(1/\delta))$. Therefore, if we choose

$$\epsilon_0 = \epsilon\sqrt{\frac{Mn}{2C(\gamma_0 + R\gamma_1)\log(1/\delta)}}, \tag{A.3}$$

we have

$$2\alpha\bar{\rho}_0 + 2(R-1)\alpha\bar{\rho}_1 = \epsilon/4.$$

Note that we choose $\epsilon_0 = \mathcal{O}(1)$, we have

$$\epsilon = \mathcal{O}\bigg(\sqrt{\frac{(\gamma_0 + R\gamma_1)\log(1/\delta)}{Mn}}\bigg).$$

Furthermore, the conditions $2 \leq \alpha \leq MB\log(1/\gamma)/(4ce^{\epsilon_0})$, $6ce^{\epsilon_0} \leq MB\log(1/\gamma)$ will be satisfied if we have $\epsilon_0 = \mathcal{O}(1)$, $\epsilon = \mathcal{O}(\log(1/\delta))$, $Mn\min\{\gamma_0, \gamma_1\} = \Omega(\log(1/\delta)/\epsilon)$.

In addition, we have

$$K\lceil R/M\rceil\rho_2 = 24\frac{\gamma_2^2\alpha K\lceil R/M\rceil}{B_2^2\sigma^2} = 24\frac{\alpha K\lceil R/M\rceil}{n^2\sigma^2} \leq \epsilon/4,$$

where the last equality comes from the fact that

$$\sigma^2 = \mathcal{O}\bigg(\max\bigg\{\frac{KR\log(1/\delta)}{Mn^2\epsilon^2}, \frac{\log(1/\delta)}{B_2^2\epsilon}\bigg\}\bigg). \tag{A.4}$$

In addition, the conditions $B_2\sigma \geq 2\sqrt{5}$ and $\alpha \leq \sigma^2 B_2^2\log(1/\gamma_2)/2$ can be satisfied due to $\sigma^2 = \mathcal{O}(\log(1/\delta)/(B_2^2\epsilon))$ and $\epsilon = \mathcal{O}(\log(1/\delta))$.

As a result, we have that Algorithm 1 is (ϵ, δ)-DP with

$$\epsilon = \mathcal{O}\bigg(\sqrt{\frac{(\gamma_0 + R\gamma_1)\log(1/\delta)}{Mn}}\bigg),$$

under the conditions that $\epsilon_0 = \mathcal{O}(1)$, $\epsilon = \mathcal{O}(\log(1/\delta))$, $Mn\min\{\gamma_0, \gamma_1\} = \Omega(\log(1/\delta)/\epsilon)$. □

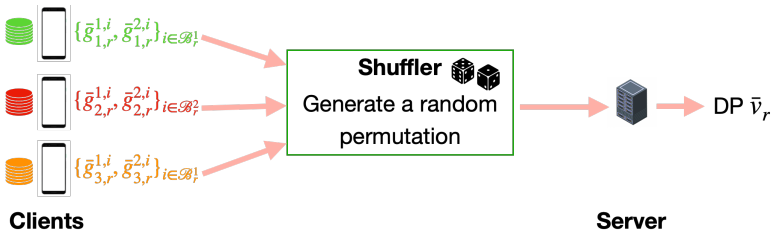


Figure 2: Illustration of the shuffled model.

Proof of Theorem A.1. Following the similar arguement as in the proof of Theorem 4.1 about privacy guarantee of the variance reduction term at server, we can get our result. □

A.4 Proof of Utility Guarantees

Recall that we are using the following ϵ_0-LDP mechanism $\mathcal{M}_L$ (Duchi et al., 2018) in Algorithm 1.

Algorithm 2 LDP Mechanism ($\mathcal{M}_L$: client-side LDP method)

input x with bounded ℓ_2 norm C, local privacy parameter ϵ_0

1: $z = \begin{cases} C\frac{x}{\|x\|_2}, & \text{with probability } \frac{1}{2} + \frac{\|x\|_2}{2C}, \\ -C\frac{x}{\|x\|_2}, & \text{otherwise.} \end{cases}$

2: Sample v uniformaly from the unit sephere S^d

3: $\widetilde{z} = \begin{cases} \text{sign}\left(\langle v, z\rangle\right)v, & \text{with probability } \frac{e^{\epsilon_0}}{1+e^{\epsilon_0}}, \\ -\text{sign}\left(\langle v, z\rangle\right)v, & \text{otherwise.} \end{cases}$

4: $\bar{z} = B\widetilde{z}$, where $B = C\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\frac{\sqrt{\pi}}{2}\frac{d\Gamma\left(\frac{d-1}{2}+1\right)}{\Gamma\left(\frac{d}{2}+1\right)}$

output $\bar{z}$

To provide the utility guarantees, we need the following lemmas.

Lemma A.10. (Duchi et al., 2018) The menchanism $\mathcal{M}_L$ in Algorithm 2 is ϵ_0-LDP. In addition, for any $x \in \mathbb{R}^d$ with $\|x\|_2 \leq C$, we have

$$\mathbb{E}[\mathcal{M}_L(x)] = x \text{ and } \|\mathcal{M}_L(x)\|_2 \leq C\frac{e^{\epsilon_0}+1}{e^{\epsilon_0}-1}\frac{3\sqrt{\pi}\sqrt{d}}{4}.$$

Lemma A.11. (Lei et al., 2017) Consider vectors a_i satisfying $\sum_{i=1}^n a_i = 0$. Let $\mathcal{B}$ be a uniform random subset of $\{1, 2, \ldots, n\}$ with size m, we have

$$\mathbb{E}\left\|\frac{1}{m}\sum_{i\in\mathcal{B}} a_i\right\|_2^2 \leq \frac{\mathbb{1}\{|\mathcal{B}| < n\}}{mn}\sum_{i=1}^n \|a_i\|_2^2.$$

Now, we are ready to provide the utility guarantees of our method.

Proof of Theorem 4.2. Note that according to the clipping parameters we choose, we do not execute any clipping procedure in Algorithm 1. Therefore, according to the local update in Algrotihm 1 and

the smoothness, we have

$$
\begin{aligned}
F(w^{\widetilde{m}}_{r+1,k+1}) &\leq F(w^{\widetilde{m}}_{r+1,k}) + \langle \nabla F(w^{\widetilde{m}}_{r+1,k}), w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k} \rangle + \frac{L}{2}\|w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k}\|_2^2 \\
&= F(w^{\widetilde{m}}_{r+1,k}) + \frac{1}{2\eta}\Big(\|w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k} + \eta \nabla F(w^{\widetilde{m}}_{r+1,k})\|_2^2 - \eta^2 \|\nabla F(w^{\widetilde{m}}_{r+1,k})\|_2^2 \\
&\qquad\qquad - \|w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k}\|_2^2 \Big) + \frac{L}{2}\|w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k}\|_2^2 \\
&= F(w^{\widetilde{m}}_{r+1,k}) - \frac{\eta}{2}\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|_2^2 + \frac{\eta}{2}\|\bar{v}^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k})\|_2^2 \\
&\quad - \left(\frac{1}{2\eta} - \frac{L}{2}\right)\|w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k}\|_2^2 \\
&\leq F(w^{\widetilde{m}}_{r+1,k}) - \frac{\eta}{2}\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|_2^2 + \frac{\eta}{2}\|\bar{v}^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k})\|_2^2 - \frac{1}{4\eta}\|w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k}\|_2^2,
\end{aligned} \tag{A.5}
$$

where the last inequality is due to the fact that $\eta \leq 1/(2L)$.
Rearranging terms in equation A.5, we can get

$$
\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2 \leq \frac{2}{\eta}\big(F(w^{\widetilde{m}}_{r+1,k}) - F(w^{\widetilde{m}}_{r+1,k+1})\big) + \|\bar{v}^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k})\|_2^2 - \frac{1}{2\eta^2}\|w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k}\|_2^2.
$$

Since $w^{\widetilde{m}}_{r+1,1} = x_r$ and $w^{\widetilde{m}}_{r+1,K+1} = x_{r+1}$, averaging over K and taking expectation, we can obtain

$$
\begin{aligned}
\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w^{\widetilde{m}}_{r+1,k})\|^2 \leq \frac{2}{K\eta}\big(\mathbb{E}F(x_r) - \mathbb{E}F(x_{r+1})\big) + \frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\bar{v}^{\widetilde{m}}_{r,k} - \nabla F(w^{\widetilde{m}}_{r+1,k})\|_2^2 \\
- \frac{1}{2\eta^2}\frac{1}{K}\sum_{k=1}^{K}\|w^{\widetilde{m}}_{r+1,k+1} - w^{\widetilde{m}}_{r+1,k}\|_2^2.
\end{aligned} \tag{A.6}
$$

In addition, for any j, we have $\bar{v}^j_{r,k} = \bar{v}^j_{r,k-1} + d^j_{k-1} + \nu_k = \bar{v}^j_{r,k-1} + \nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k}) - \nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k-1}) + \nu_k$, where $\nu_k \sim C_3 \cdot N(0, \sigma^2\mathbf{I})$. Therefore, we can obtain

$$
\begin{aligned}
&\mathbb{E}\|\bar{v}^j_{r,k} - \nabla F(w^j_{r+1,k})\|_2^2 \\
&= \mathbb{E}\big\|\big(\bar{v}^j_{r,k-1} - \nabla F(w^j_{r+1,k-1})\big) + \nu_k \\
&\quad + \big(\nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k}) - \nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k-1}) - \nabla F_j(w^j_{r+1,k}) + \nabla F_j(w^j_{r+1,k-1})\big) \\
&\quad + \big(\nabla F_j(w^j_{r+1,k}) - \nabla F_j(w^j_{r+1,k-1}) + \nabla F(w^j_{r+1,k-1}) - \nabla F(w^j_{r+1,k})\big)\big\|_2^2 \\
&= \mathbb{E}\big\|\nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k}) - \nabla F_{j,\mathcal{B}^j_{r,k}}(w^j_{r+1,k-1}) - \nabla F_j(w^j_{r+1,k}) + \nabla F_j(w^j_{r+1,k-1})\big\|_2^2 + \mathbb{E}\|\nu_k\|_2^2 \\
&\quad + \mathbb{E}\big\|\big(\bar{v}^j_{r,k-1} - \nabla F(w^j_{r+1,k-1})\big) \\
&\qquad + \big(\nabla F_j(w^j_{r+1,k}) - \nabla F_j(w^j_{r+1,k-1}) + \nabla F(w^j_{r+1,k-1}) - \nabla F(w^j_{r+1,k})\big)\big\|_2^2 \\
&\leq \frac{L^2}{b}\mathbb{E}\|w^j_{r+1,k} - w^j_{r+1,k-1}\|_2^2 + \left(1 + \frac{1}{K}\right)\mathbb{E}\|\bar{v}^j_{r,k-1} - \nabla F(w^j_{r+1,k-1})\|_2^2 + \mathbb{E}\|\nu_k\|_2^2 \\
&\quad + (1+K)\mathbb{E}\big\|\nabla F_j(w^j_{r+1,k}) - \nabla F_j(w^j_{r+1,k-1}) + \nabla F(w^j_{r+1,k-1}) - \nabla F(w^j_{r+1,k})\big\|_2^2,
\end{aligned}
$$

where the second equality is due to the independence of the random variables, the inequality comes from Lemma A.11 and smoothness. Therefore, we can further obtain

$$\begin{aligned}
&\mathbb{E}\|\bar{v}_{r,k}^{j} - \nabla F(w_{r+1,k}^{j})\|_2^2 \\
&\leq e\mathbb{E}\|v_{r,0}^{j} - \nabla F(w_{r+1,0}^{j})\|_2^2 + \frac{eL^2}{b}\sum_{k=1}^{K}\mathbb{E}\|w_{r+1,k}^{j} - w_{r+1,k-1}^{j}\|_2^2 + e\sum_{k=1}^{K}\mathbb{E}\|\nu_k\|_2^2 \\
&\quad + e(1+K)\sum_{k=1}^{K}\mathbb{E}\big\|\nabla F_j(w_{r+1,k}^{j}) - \nabla F_j(w_{r+1,k-1}^{j}) + \nabla F(w_{r+1,k-1}^{j}) - \nabla F(w_{r+1,k}^{j})\big\|_2^2 \\
&\leq e\mathbb{E}\|v_{r,0}^{j} - \nabla F(w_{r+1,0}^{j})\|_2^2 + \left(\frac{eL^2}{b} + 8eK\tau^2 + edL^2\sigma^2\right)\sum_{k=1}^{K}\mathbb{E}\|w_{r+1,k}^{j} - w_{r+1,k-1}^{j}\|_2^2,
\end{aligned} \tag{A.7}$$

where the second inequality is due to the fact that $\nu \sim C_3 \cdot N(0, \sigma^2\mathbf{I})$ with $C_3 = L\|w_{r+1,k}^{j} - w_{r+1,k-1}^{j}\|_2$, $\mathbb{E}\|\nu_k\|_2^2 = dL^2\sigma^2\mathbb{E}\|w_{r+1,k}^{j} - w_{r+1,k-1}^{j}\|_2^2$, and the second order heterogeneity.

Plugging the result in equation A.7 into equation A.6, we can get

$$\begin{aligned}
\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|\nabla F(w_{r+1,k}^{\widetilde{m}})\|^2 &\leq \frac{2}{K\eta}\big(\mathbb{E}F(x_r) - \mathbb{E}F(x_{r+1})\big) + e\mathbb{E}\|\bar{v}_r - \nabla F(x_r)\|_2^2 \\
&\quad + \left(\frac{eKL^2}{b} + 8eK^2\tau^2 + edKL^2\sigma^2\right)\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\|w_{r+1,k}^{\widetilde{m}} - w_{r+1,k-1}^{\widetilde{m}}\|_2^2 \\
&\quad - \frac{1}{2\eta^2}\frac{1}{K}\sum_{k=1}^{K}\|w_{r+1,k+1}^{\widetilde{m}} - w_{r+1,k}^{\widetilde{m}}\|_2^2 \\
&\leq \frac{2}{K\eta}\big(\mathbb{E}F(x_r) - \mathbb{E}F(x_{r+1})\big) + e\mathbb{E}\|\bar{v}_r - \nabla F(x_r)\|_2^2 \\
&\quad - \frac{1}{4\eta^2}\frac{1}{K}\sum_{k=1}^{K}\|w_{r+1,k+1}^{\widetilde{m}} - w_{r+1,k}^{\widetilde{m}}\|_2^2
\end{aligned} \tag{A.8}$$

where the last inequality is due to the fact that $\eta \leq C/(K\tau)$,$\eta \leq C\sqrt{b}/(\sqrt{K}L)$, $\eta \leq C/(\sqrt{Kd}L\sigma)$. Furthermore, averaging equation A.8 from $r = 0, \ldots, R-1$, we can get

$$\begin{aligned}
\mathbb{E}\|\nabla F(\widetilde{x})\|^2 &\leq \frac{2}{RK\eta}\big(\mathbb{E}F(x_0) - \mathbb{E}F(x_R)\big) + \frac{e}{R}\sum_{r=0}^{R-1}\mathbb{E}\|\bar{v}_r - \nabla F(x_r)\|_2^2 \\
&\quad - \frac{1}{4\eta^2}\frac{1}{RK}\sum_{r=0}^{R-1}\sum_{k=1}^{K}\|w_{r+1,k+1}^{\widetilde{m}} - w_{r+1,k}^{\widetilde{m}}\|_2^2.
\end{aligned} \tag{A.9}$$

Additionally, according Algorithm 1, set $\rho = \beta$ and $B = b$, we have

$$\bar{v}_r = (1-\beta)\bar{v}_{r-1} + \frac{1}{Mb}\sum_{m=1}^{M}\sum_{i=1}^{b}\big((1-\beta)\bar{g}_{m,r}^{1,i} + \beta\bar{g}_{m,r}^{2,i}\big).$$

Therefore, we can get

$$\begin{aligned}\bar{v}_r - \nabla F(x_r) = (1-\beta)\big(\bar{v}_{r-1} - \nabla F(x_{r-1})\big) + \beta\bigg(\frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \nabla F(x_r)\bigg) \\ + (1-\beta)\bigg(\frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_{r-1}) + \nabla F(x_{r-1}) - \nabla F(x_r)\bigg) \\ + (1-\beta)\underbrace{\bigg(\frac{1}{Mb}\sum_{m=1}^{M}\sum_{i=1}^{b}\bar{g}_{m,r}^{1,i} - \bigg(\frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_{r-1})\bigg)\bigg)}_{\mathrm{err}_1} \\ + \beta\underbrace{\bigg(\frac{1}{Mb}\sum_{m=1}^{M}\sum_{i=1}^{b}\bar{g}_{m,r}^{2,i} - \frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r)\bigg)}_{\mathrm{err}_2}.\end{aligned}$$

Case 1: If we are using mini-batch gradients for the server variance reduction term, i.e., the batch size $|\mathcal{B}_r^j| = Kb < n$, according to the definition of $\bar{g}_{m,r}^{1,i}$, $\bar{g}_{m,r}^{2,i}$, Lemma A.10, and Lemma A.11, we have the following conditional probability (up to r-th iteration) hold

$$\begin{aligned}\mathbb{E}_r\big\|\bar{v}_r - \nabla F(x_r)\big\|_2^2 &\le (1-\beta)^2\mathbb{E}_r\big\|\bar{v}_{r-1} - \nabla F(x_{r-1})\big\|_2^2 \\ &\quad + 2\beta^2\mathbb{E}_r\bigg\|\frac{1}{M}\sum_{j=1}^{M}\nabla F_{j,\mathcal{B}_r^j}(x_r) - \frac{1}{M}\sum_{j=1}^{M}\nabla F_j(x_r)\bigg\|_2^2 + 2\beta^2 dC_0\frac{G^2}{Mn} \\ &\quad + 2(1-\beta)^2\frac{L^2}{MKb}\mathbb{E}_r\big\|x_r - x_{r-1}\big\|_2^2 + 2(1-\beta)^2 dC_0\frac{L^2}{Mn}\mathbb{E}\big\|x_r - x_{r-1}\big\|_2^2 \\ &\le (1-\beta)^2\mathbb{E}_r\big\|\bar{v}_{r-1} - \nabla F(x_{r-1})\big\|_2^2 + 2\beta^2\frac{G^2}{MKb} + 2\beta^2 dC_0\frac{G^2}{Mn} \\ &\quad + 2(1-\beta)^2\frac{L^2}{MKb}\mathbb{E}_r\big\|x_r - x_{r-1}\big\|_2^2 + 2(1-\beta)^2 dC_0\frac{L^2}{Mn}\mathbb{E}\big\|x_r - x_{r-1}\big\|_2^2,\end{aligned} \tag{A.10}$$

where $C_0 = 4(e^{\epsilon_0}+1)^2/(e^{\epsilon_0}-1)^2$.

Following the similar proofs in Patel et al. (2022), we can get

$$\begin{aligned}\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|\bar{v}_r - \nabla F(x_r)\big\|_2^2 \le \frac{2(1-\beta)^2L^2}{\beta MKbR}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1} - x_r\big\|_2^2 + \frac{2(1-\beta)^2 dC_0L^2}{\beta MnR}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1} - x_r\big\|_2^2 \\ + 2\beta\frac{G^2}{MKb} + \frac{G^2}{\beta RMb_0} + 2\beta dC_0\frac{G^2}{Mn} + dC_0\frac{G^2}{\beta RMn}.\end{aligned} \tag{A.11}$$

Furthremore, we have $C_0 = 4(e^{\epsilon_0}+1)^2/(e^{\epsilon_0}-1)^2 = C/\epsilon_0^2$ when $\epsilon_0 = \mathcal{O}(1)$ for some constant C. Therefore, according to the definition of ϵ_0 in equation A.3, and the condition that $\beta \ge 1/R$, we have (ignoring the constants and $\log(1/\delta)$ for simpilcity)

$$\begin{aligned}\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|\bar{v}_r - \nabla F(x_r)\big\|_2^2 \lesssim \frac{(1-\beta)^2L^2}{\beta MKbR}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1} - x_r\big\|_2^2 + \frac{(1-\beta)^2 dL^2R}{\beta M^2n^2\epsilon^2R}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1} - x_r\big\|_2^2 \\ + \frac{\beta G^2}{MKb} + \frac{G^2}{\beta RMb_0} + \beta d\frac{RG^2}{M^2n^2\epsilon^2} + d\frac{G^2}{\beta RM^2n^2\epsilon^2}.\end{aligned} \tag{A.12}$$

Therefore, under the additional condition that

$$\frac{G^2}{\beta RMb_0} \le \frac{\beta G^2}{MKb}, \tag{A.13}$$

we have

$$\begin{aligned}\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|\bar{v}_r-\nabla F(x_r)\big\|_2^2 &\lesssim \frac{(1-\beta)^2L^2}{\beta MKbR}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1}-x_r\big\|_2^2+\frac{(1-\beta)^2dL^2R}{\beta M^2n^2\epsilon^2R}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1}-x_r\big\|_2^2\\ &\quad+\frac{\beta G^2}{MKb}+\beta d\frac{RG^2}{M^2n^2\epsilon^2}\\ &\lesssim\left(\frac{L^2}{\beta MKb}+\frac{dL^2R}{\beta M^2n^2\epsilon^2}\right)\frac{1}{R}\sum_{r=0}^{R-1}\mathbb{E}\big\|x_{r+1}-x_r\big\|^2+\frac{\beta G^2}{MKb}+\beta d\frac{RG^2}{M^2n^2\epsilon^2}\\ &\lesssim K^2\left(\frac{L^2}{\beta MKb}+\frac{dL^2R}{\beta M^2n^2\epsilon^2}\right)\frac{1}{RK}\sum_{r=0}^{R-1}\sum_{k=1}^{K}\|w_{r+1,k+1}^{\widetilde{m}}-w_{r+1,k}^{\widetilde{m}}\|_2^2\\ &\quad+\frac{\beta G^2}{MKb}+\beta d\frac{RG^2}{M^2n^2\epsilon^2},\end{aligned}\tag{A.14}$$

where the last line is due to the following

$$\frac{1}{R}\sum_{r=0}^{R-1}\|x_{r+1}-x_r\|_2^2\le\frac{K^2}{RK}\sum_{r=0}^{R-1}\sum_{k=1}^{K}\|w_{r+1,k+1}^{\widetilde{m}}-w_{r+1,k}^{\widetilde{m}}\|_2^2.$$

Plugging results in equation A.14 into equation A.9, we have

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2\lesssim\frac{1}{RK\eta}\big(\mathbb{E}F(x_0)-\mathbb{E}F(x_R)\big)+\frac{\beta G^2}{MKb}+\beta d\frac{RG^2}{M^2n^2\epsilon^2}\\ +\left(K^2\left(\frac{L^2}{\beta MKb}+\frac{dL^2R}{\beta M^2n^2\epsilon^2}\right)-\frac{1}{4\eta^2}\right)\frac{1}{KR}\sum_{r=0}^{R-1}\sum_{k=1}^{K}\mathbb{E}\|w_{r+1,k}^{\widetilde{m}}-w_{r+1,k-1}^{\widetilde{m}}\|_2^2.\end{aligned}\tag{A.15}$$

Therefore, if we choose $\eta\le C_1\sqrt{\beta Mb}/(\sqrt{K}L)$ and $\eta\le C_1\sqrt{\beta}Mn\epsilon/(\sqrt{dR}KL)$ for some constant C_1, we have

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2\lesssim\frac{1}{RK\eta}\big(\mathbb{E}F(x_0)-\mathbb{E}F(x_R)\big)+\frac{\beta G^2}{MKb}+\beta d\frac{RG^2}{M^2n^2\epsilon^2}.\tag{A.16}$$

Recall that we also have $\eta\le C_1/L$, $\eta\le C_1/(K\tau)$, $\eta\le C_1\sqrt{b}/(\sqrt{K}L)$, $\eta\le C_1/(\sqrt{Kd}L\sigma)$, where σ is defined in equation A.4.

Plugging the choice of η, we get

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2\lesssim\frac{D_F\tau}{R}+\frac{D_FL}{RK}+\frac{D_FL}{R\sqrt{Kb}}+\frac{D_FL}{R\sqrt{\beta MKb}}+\frac{D_FL\sqrt{d}}{\sqrt{R\beta}Mn\epsilon}+\frac{D_FL\sqrt{d}}{\sqrt{RM}n\epsilon}+\frac{D_FL\sqrt{d}}{R\sqrt{K\epsilon}b}\\ +\frac{\beta G^2}{MKb}+\beta d\frac{RG^2}{M^2n^2\epsilon^2},\end{aligned}$$

where $D_F=F(x_0)-F(x^*)$. Therefore, if we choose β as

$$\beta=\max\left\{\frac{1}{R},\min\left\{\frac{(Mn\epsilon)^{2/3}(D_FL)^{2/3}}{d^{1/3}RG^{4/3}},\frac{(MKb)^{1/3}(DL)^{2/3}}{R^{2/3}G^{4/3}}\right\}\right\}=\max\{\beta_1,\beta_2\}$$

we have

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \lesssim & \frac{D_F\tau}{R}+\frac{D_FL}{RK}+\frac{D_FL}{R\sqrt{Kb}}+\frac{D_FL\sqrt{d}}{R\sqrt{K\epsilon b}}+\frac{D_FL\sqrt{d}}{\sqrt{RMn\epsilon}}+\frac{(D_FLG)^{2/3}d^{1/6}}{\sqrt{RMKb}(Mn\epsilon)^{1/3}}+\frac{(D_FLG)^{2/3}\sqrt{d}}{R^{1/6}(MKb)^{1/6}Mn\epsilon}\\ &+\frac{(D_FLG)^{2/3}}{(MKb)^{2/3}}+\frac{(dD_FLG)^{2/3}}{(Mn\epsilon)^{4/3}}+\frac{G^2}{MKbR}+\frac{dG^2}{M^2n^2\epsilon^2}\\ \lesssim & \underbrace{\frac{D_F\tau}{R}+\frac{D_FL}{RK}+\frac{D_FL}{R\sqrt{Kb}}+\frac{(D_FLG)^{2/3}}{(MKb)^{2/3}}+\frac{G^2}{MKbR}}_{\text{non-private error}}+\underbrace{\frac{D_FL\sqrt{d}}{R\sqrt{K\epsilon b}}+\frac{D_FL\sqrt{d}}{\sqrt{RMn\epsilon}}}_{\text{private error local}}\\ &+\underbrace{\left(1+\left(\frac{Mn\epsilon}{(RdMKb)^{1/2}}\right)+\left(\frac{Mn\epsilon}{(RdMKb)^{1/2}}\right)^{1/3}\right)\frac{(dD_FLG)^{2/3}}{(Mn\epsilon)^{4/3}}+\frac{dG^2}{M^2n^2\epsilon^2}}_{\text{private error server}}.\end{aligned} \tag{A.17}$$

The condition we have for this result is $1/(Rb_0) \le \beta^2/(Kb)$ and $\beta \le 1$.

Case 2: if we are using full gradients for the server variance reduction term, i.e., the batch size $|\mathcal{B}_r^j| = Kb = b_0 = n$, the result in equation A.15 reduces to (without condition in equation A.13)

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \lesssim & \frac{1}{RK\eta}\big(\mathbb{E}F(x_0)-\mathbb{E}F(x_R)\big)+\beta d\frac{RG^2}{M^2n^2\epsilon^2}\\ &+\left(K^2\frac{2dL^2R}{\beta M^2n^2\epsilon^2}-\frac{1}{4\eta^2}\right)\frac{1}{KR}\sum_{r=0}^{R-1}\sum_{k=1}^{K}\mathbb{E}\|w_{r+1,k}^{\widetilde{m}}-w_{r+1,k-1}^{\widetilde{m}}\|_2^2.\end{aligned} \tag{A.18}$$

Therefore, if we choose $\eta \le C_1\sqrt{\beta}Mn\epsilon/(\sqrt{dR}KL)$, we have

$$\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \lesssim \frac{1}{RK\eta}\big(\mathbb{E}F(x_0)-\mathbb{E}F(x_R)\big)+\beta d\frac{RG^2}{M^2n^2\epsilon^2}. \tag{A.19}$$

Recall that we also have $\eta \le C_1/L$, $\eta \le C_1/(K\tau)$, $\eta \le C_1\sqrt{b}/(\sqrt{K}L)$, $\eta \le C_1/(\sqrt{Kd}L\sigma)$. Plugging the choice of η, we get

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \lesssim & \frac{D_F\tau}{R}+\frac{D_FL}{RK}+\frac{D_FL}{R\sqrt{Kb}}+\frac{D_FL\sqrt{d}}{\sqrt{R\beta}Mn\epsilon}+\frac{D_FL\sqrt{d}}{\sqrt{RM}n\epsilon}+\frac{D_FL\sqrt{d}}{R\sqrt{K\epsilon}b}\\ &+\beta d\frac{RG^2}{M^2n^2\epsilon^2}.\end{aligned}$$

Therefore, if we choose β as

$$\beta = \max\left\{\frac{1}{R}, \frac{(Mn\epsilon)^{2/3}(DL)^{2/3}}{d^{1/3}RG^{4/3}}\right\} = \max\{\beta_1, \beta_2\},$$

we have

$$\begin{aligned}\mathbb{E}\|\nabla F(\widetilde{x})\|^2 \lesssim & \underbrace{\frac{D_F\tau}{R}+\frac{D_FL}{RK}+\frac{D_FL}{R\sqrt{Kb}}}_{\text{non-private error}}+\underbrace{\frac{D_FL\sqrt{d}}{R\sqrt{K\epsilon}b}+\frac{D_FL\sqrt{d}}{\sqrt{RM}n\epsilon}}_{\text{private error local}}\\ &+\underbrace{\frac{(dD_FLG)^{2/3}}{(Mn\epsilon)^{4/3}}+\frac{dG^2}{M^2n^2\epsilon^2}}_{\text{private error server}}.\end{aligned} \tag{A.20}$$

We only need $\beta \le 1$ for this result. □

Proof of Theorem A.2. This has been proved in the Case 2 of the Proof of Theorem 4.2. □

Proof of Theorem A.3. The proof of this result directly follows the proof of Theorem 4.2. We can just set $K = 1, \tau = L$, and ignore the private error local and the term $D_FL/(R\sqrt{Kb})$. □

Proof of Theorem A.4. The proof of this result directly follows the proof of Case 2 in Theorem 4.2 by setting $K = 1, \tau = L$, and ignore the private error local and the term $D_FL/(R\sqrt{Kb})$. □