Revisiting Differentially Private Algorithms for Decentralized Online Learning

Xiaoyu Wang [1] **Wenhao Yang** [2] **Chang Yao** [1 3] **Mingli Song** [3] **Yuanyu Wan** [1 3]

Abstract

Although the differential privacy (DP) of decentralized online learning has garnered considerable attention recently, existing algorithms are unsatisfactory due to their inability to achieve $(\epsilon, 0)$-DP over all T rounds, recover the optimal regret in the non-private case, and maintain the lightweight computation under complex constraints. To address these issues, we first propose a new decentralized online learning algorithm satisfying $(\epsilon, 0)$-DP over T rounds, and show that it can achieve $\widetilde{O}(n(\rho^{-1/4} + \epsilon^{-1}\rho^{1/4})\sqrt{T})$ and $\widetilde{O}(n(\rho^{-1/2} + \epsilon^{-1}))$ regret bounds for convex and strongly convex functions respectively, where n is the number of local learners and ρ is the spectral gap of the communication matrix. As long as $\epsilon = \Omega(\sqrt{\rho})$, these bounds nearly match existing lower bounds in the non-private case, which implies that $(\epsilon, 0)$-DP of decentralized online learning may be ensured nearly for free. Our key idea is to design a block-decoupled accelerated gossip strategy that can be incorporated with the classical tree-based private aggregation, and also enjoys a faster average consensus among local learners. Furthermore, we develop a projection-free variant of our algorithm to keep the efficiency under complex constraints. As a trade-off, the above regret bounds degrade to $\widetilde{O}(n(T^{3/4} + \epsilon^{-1}T^{1/4}))$ and $\widetilde{O}(n(T^{2/3} + \epsilon^{-1}))$ respectively, which however are even better than the existing private centralized projection-free online algorithm.

1. Introduction

Decentralized online learning (Li et al., 2023) is a popular way to solve distributed applications with streaming data. To be precise, it is commonly characterized as a repeated game between an adversary and n local learners connected by an undirected graph $\mathcal{G} = ([n], E)$, where $E \subseteq [n] \times [n]$ denotes the edge set. At each round t, each local learner $i \in [n]$ first needs to select a local decision $\mathbf{x}_i(t)$ from a feasible set $\mathcal{K} \subseteq \mathbb{R}^d$, and then receives a local loss function $f_{t,i}(\cdot) : \mathcal{K} \mapsto \mathbb{R}$ determined by the adversary. The goal of each local learner i is to choose a sequence of decisions to minimize the regret in terms of the global function $f_t(\mathbf{x}) = \sum_{j=1}^n f_{t,j}(\mathbf{x})$ at each round t, i.e., $R_{T,i} = \sum_{t=1}^T f_t(\mathbf{x}_i(t)) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^T f_t(\mathbf{x})$, where T is the number of total rounds.

To this end, each local learner i needs to estimate global information by communicating with its neighbors. Note that as a common protocol, the communications between these learners occur via only a single gossip step (Xiao & Boyd, 2004) based on a weight matrix $P \in \mathbb{R}^{n\times n}$ at each round. Moreover, to make this problem more tractable, it is common to assume that the decision set $\mathcal{K}$ and all local loss functions $f_{t,i}(\cdot)$ are convex. This setting is known as decentralized online convex optimization (D-OCO), and has been extensively studied (Yan et al., 2013; Hosseini et al., 2013; Zhang et al., 2017; Wan et al., 2020; 2022; 2024a;b), yielding nearly optimal $\widetilde{O}(n\rho^{-1/4}\sqrt{T})$ and $\widetilde{O}(n\rho^{-1/2})$ regret for convex and strongly convex functions, respectively, where $\rho < 1$ is the spectral gap of P.[1]

It is worth noting that one critical appeal of D-OCO is the intrinsic privacy-preserving property, as it prevents each local learner from sharing any sensitive data directly (Yan et al., 2013). However, for traditional D-OCO algorithms, the privacy-preserving ability is limited in the sense that they are still vulnerable to adversarial attacks, such as membership inference attack (Shokri et al., 2017) and model inversion attack (Fredrikson et al., 2015). For this reason, there has been a growing research interest (Li et al., 2018; Hou et al., 2019; Lü et al., 2023; Chen et al., 2023; Cheng et al., 2023; Zhang et al., 2024b) in designing D-OCO algorithms with differential privacy (DP), which prevents any attacker from identifying whether a particular individual is included in a dataset (Dwork et al., 2006).

[1]School of Software Technology, Zhejiang University, Ningbo, China [2]National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing, China [3]State Key Laboratory of Blockchain and Data Security, Zhejiang University, Hangzhou, China. Correspondence to: Yuanyu Wan <wanyy@zju.edu.cn>.

Proceedings of the 42^{nd} International Conference on Machine Learning, Vancouver, Canada. PMLR 267, 2025.

[1]The $\widetilde{O}(\cdot)$ notation hides constant factors as well as the polylogarithmic factors.

However, these private D-OCO algorithms are unsatisfactory for three issues. First, their main idea is to directly add noise to original decisions of some traditional D-OCO algorithms, which cannot trade off the regret and privacy well. Specifically, to achieve the state-of-the-art regret bound of $O(n^{5/4}\rho^{-1/2}\epsilon^{-1}\sqrt{T})$ and $\widetilde{O}(n^{3/2}(\rho\epsilon)^{-1})$ for convex and strongly convex functions (Li et al., 2018), where ϵ is a constant regarding the privacy level, these algorithms can only utilize time-decaying noise to obtain $(\epsilon T, 0)$-DP over all T rounds in the worst case, which degrades linearly according to the increase of T. Second, there exist large gaps in terms of n and ρ between these state-of-the-art regret bounds and nearly optimal regret bounds in the non-private case, which are caused by the standard gossip step (Xiao & Boyd, 2004) utilized in the communication. Third, a projection operation is required by these algorithms to ensure the feasibility of each local decision, which could be time-consuming when facing complex decision sets (Hazan & Kale, 2012).

To address these issues, this paper first proposes a new D-OCO algorithm, namely PD-FTGL, which not only satisfies $(\epsilon, 0)$-DP over all T rounds, but also can achieve improved $\widetilde{O}(n(\rho^{-1/4}+\epsilon^{-1}\rho^{1/4})\sqrt{T})$ and $\widetilde{O}(n(\rho^{-1/2}+\epsilon^{-1}))$ regret bounds for convex and strongly convex functions, respectively. For $\epsilon = \Omega(\sqrt{\rho})$, these regret bounds match the nearly optimal regret bounds in the non-private case (Wan et al., 2024a;b) up to polylogarithmic factors, which implies that our PD-FTGL can ensure $(\epsilon, 0)$-DP nearly for free over a wide range of ϵ. Compared with existing private D-OCO algorithms, the main novelty of our PD-FTGL is to utilize the tree-based private aggregation (Dwork et al., 2010; Jain et al., 2012) to achieve a better trade-off between the regret and privacy, and exploit an accelerated gossip strategy (Liu & Morse, 2011) to reduce the approximation error caused by communication.

Note that although the tree-based technique has been utilized to equip various non-private algorithms for centralized online learning (i.e., $n = 1$) with the $(\epsilon, 0)$-DP guarantee (Smith & Thakurta, 2013; Agarwal & Singh, 2017; Ene et al., 2021; Kairouz et al., 2021), it is non-trivial to apply this technique in general D-OCO as explained below.

- This technique requires that any decision of the non-private algorithm is determined by the partial sum of some variables, e.g., historical gradients in the classical follow-the-regularized-leader (FTRL) algorithm (Shalev-Shwartz & Singer, 2007; Hazan, 2016).
- Unfortunately, existing D-OCO algorithms, including the decentralized variants of FTRL based on the accelerated gossip strategy (Wan et al., 2024a;b), do not satisfy the above requirement, because their decisions rely on coupled communication between adjacent rounds.

To tackle this challenge, our PD-FTGL follows a blocking update mechanism (Wan et al., 2024a;b) to exploit the accelerated gossip strategy, but adopts block-decoupled communication for incorporating the tree-based technique. Furthermore, we notice that PD-FTGL still needs to perform projection operations per block, and thus may also be time-consuming over complex decision sets. To improve the efficiency, we develop a projection-free variant by replacing its projection operations with only linear optimization steps. Our analysis reveals that this variant can achieve $\widetilde{O}(n(T^{3/4}+\epsilon^{-1}T^{1/4}))$ and $\widetilde{O}(n(T^{2/3}+\epsilon^{-1}))$ regret bounds for convex and strongly convex functions, respectively. Although these regret bounds are worse than those of PD-FTGL, such degeneration is a common price for the projection-free property in both centralized and decentralized online learning (Hazan & Kale, 2012; Wan et al., 2024b). Moreover, we want to emphasize that they are even better than the $\widetilde{O}(\epsilon^{-1}T^{3/4})$ regret bound achieved by the existing centralized projection-free online algorithm with $(\epsilon, 0)$-DP (Ene et al., 2021).

2. Related Work

In this section, we first briefly review previous studies on the differential privacy of the special D-OCO with $n = 1$, and then introduce existing algorithms for the general D-OCO. Due to the limitation of space, here we mainly discuss the results that are either directly comparable with ours or provide some technical inspirations. Discussions on additional related work can be found in Appendix A.

2.1. Differential Privacy of Special D-OCO with $n = 1$

In the special case with $n = 1$, D-OCO reduces to the classical online convex optimization (OCO) problem (Zinkevich, 2003). It is well-known that the optimal regret of OCO is $O(\sqrt{T})$ and $O(\log T)$ for convex and strongly convex functions, respectively (Abernethy et al., 2008). Moreover, these optimal regret bounds can be achieved by many algorithms such as online gradient descent (OGD) (Zinkevich, 2003), FTRL (Shalev-Shwartz & Singer, 2007; Hazan, 2016), and the follow-the-leader (FTL) algorithm for strongly convex functions (Hazan et al., 2007).

The differential privacy (DP) of OCO is first studied by Jain et al. (2012), who propose a general framework to convert any given OCO algorithm into a privacy-preserving one. However, their main idea is to simply add some Gaussian noise to each original decision of the given OCO algorithm, which cannot trade off the regret and privacy well. Let ϵ and δ be constants for privacy level. Intuitively, it is easy to ensure (ϵ, δ)-DP for the decision at each round by using noise proportional to the sensitivity, which is generally time-decaying for OCO algorithms. But, according to the classical T-fold composition theorem (Dwork & Roth, 2014), the noise should be further magnified by a factor of $O(T)$ to ensure (ϵ, δ)-DP for all T rounds, which destroys

the sublinearity of the original regret achieved by the given OCO algorithm. To address this issue, Jain et al. (2012) assume that the given OCO algorithm has linearly decaying sensitivity, which only holds for strongly convex functions, and exploit the interdependence between these decisions. Nonetheless, even combining their framework with the optimal OCO algorithm for strongly convex functions, e.g., OGD, they can only achieve a much worse regret bound of $\widetilde{O}(\epsilon^{-1}\sqrt{T})$ while ensuring (ϵ, δ)-DP for all T rounds. To further improve the regret, Jain et al. (2012) also consider a special case of OCO with quadratic functions, and propose a private variant of FTL (Hazan et al., 2007) by combining it with the tree-based private aggregation (Dwork et al., 2010). A critical precondition for this combination is that in this case, the decision of FTL can be determined by the partial sum of some variables related to the quadratic functions. Since the tree-based technique provides a better way to introduce noise, Jain et al. (2012) achieve (ϵ, δ)-DP and $O(\epsilon^{-1}(\log^{3/2} T)\log(1/\delta))$ regret for the special case.

Later, Smith & Thakurta (2013) extend the private variant of FTL (Jain et al., 2012) into OCO with strongly convex functions by replacing FTL with an approximate variant (Hazan et al., 2007) that originally updates as

$$\mathbf{x}(t+1) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \sum_{\tau=1}^{t} \tilde{f}_\tau(\mathbf{x}) \tag{1}$$

where $\tilde{f}_\tau(\mathbf{x}) = \langle \nabla f_\tau(\mathbf{x}(\tau)), \mathbf{x}\rangle + \frac{\alpha}{2}\|\mathbf{x} - \mathbf{x}(\tau)\|^2$ and α is the modulus of the strong convexity. Additionally, they propose to utilize Laplace noise, instead of Gaussian noise used in Jain et al. (2012). Based on these changes, Smith & Thakurta (2013) for the first time establish $(\epsilon, 0)$-DP and $O(\epsilon^{-1}\log^{5/2} T)$ regret for OCO with strongly convex functions. Interestingly, they show that by combining with a strongly convex approximation of general convex functions, their algorithm can be utilized to achieve an $\widetilde{O}(\epsilon^{-1}\sqrt{T})$ regret bound for the general OCO. Recently, Kairouz et al. (2021) develop a specific algorithm with (ϵ, δ)-DP for the general OCO, to achieve a better regret bound of $\widetilde{O}(\epsilon^{-1/2}\sqrt{T})$. Compared with Smith & Thakurta (2013), the critical changes include utilizing Gaussian noise and replacing the approximate FTL in (1) with FTRL (Hazan, 2016) that originally updates as

$$\mathbf{x}(t+1) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \sum_{\tau=1}^{t} \langle \nabla f_\tau(\mathbf{x}(\tau)), \mathbf{x}\rangle + \frac{1}{\eta}\|\mathbf{x}\|^2 \tag{2}$$

where η is a tunable parameter.

Moreover, Ene et al. (2021) consider the case with complex decision sets, in which the above private algorithms may be time-consuming due to the (implicit) projection operation. To tackle the computational bottleneck, they develop the first projection-free OCO algorithm that achieves $(\epsilon, 0)$-DP and an $\widetilde{O}(\epsilon^{-1}T^{3/4})$ regret bound, or (ϵ, δ)-DP and a slightly different regret bound. Besides the tree-based technique and FTRL used before, this projection-free algorithm adopts a blocking update mechanism, and performs each update via multiple iterations of the conditional gradient (CG) method (Frank & Wolfe, 1956; Jaggi, 2013), which only requires more efficient linear optimization steps.

2.2. Algorithms for General D-OCO with $n \geq 2$

The pioneering work of Yan et al. (2013) proposes the first algorithm for D-OCO, which a decentralized variant of OGD (D-OGD). The main idea of D-OGD is to first apply the standard gossip step (Xiao & Boyd, 2004) over local decisions, and then perform a local gradient descent. As proven by Yan et al. (2013), D-OGD can achieve $O(n^{5/4}\rho^{-1/2}\sqrt{T})$ and $\widetilde{O}(n^{3/2}\rho^{-1})$ regret bounds for convex functions and strongly convex functions, respectively.

After that, the standard gossip step has been widely utilized to design algorithms for D-OCO. Specifically, Hosseini et al. (2013) exploit it to develop a decentralized variant of FTRL (D-FTRL), and achieve the same $O(n^{5/4}\rho^{-1/2}\sqrt{T})$ regret bound for convex functions. To efficiently handle complex sets, Zhang et al. (2017) propose a projection-free variant of D-FTRL, and establish an $O(n^{5/4}\rho^{-1/2}T^{3/4})$ regret bound for convex functions. Later, Wan et al. (2020) introduce a blocking update mechanism to reduce the communication complexity of Zhang et al. (2017) from $O(T)$ to $O(\sqrt{T})$ while keeping the same regret bound. The first projection-free D-OCO algorithm for strongly convex functions is proposed by Wan et al. (2021a), and can reduce the regret bound and communication complexity to $\widetilde{O}(n^{3/2}\rho^{-1}T^{2/3})$ and $\widetilde{O}(T^{1/3})$, respectively. Moreover, Wan et al. (2022) unify these two algorithms (Wan et al., 2020; 2021a) into a single one that inherits the theoretical guarantees.

However, Wan et al. (2024a;b) recently have pointed out that even the above projection-based algorithms are suboptimal by establishing $\Omega(n\rho^{-1/4}\sqrt{T})$ and $\Omega(n\rho^{-1/2}\log T)$ lower bounds for D-OCO with convex and strongly convex functions, respectively. To address this issue, Wan et al. (2024a;b) develop a novel D-OCO algorithm, which reduces regret bounds for convex and strongly convex functions to $\widetilde{O}(n\rho^{-1/4}\sqrt{T})$ and $\widetilde{O}(n\rho^{-1/2})$, respectively. Their main idea is to combine the accelerated gossip strategy (Liu & Morse, 2011) with both FTRL (Hazan, 2016) and the approximate variant of FTL (Hazan et al., 2007). Moreover, Wan et al. (2024b) also develop an improved projection-free D-OCO algorithm based on the accelerated gossip strategy. Compared with the projection-free algorithm in Wan et al. (2022), this improved one can reduce the regret bounds for convex and strongly convex functions to $O(nT^{3/4})$ and $\widetilde{O}(nT^{2/3})$ by increasing the communication complexity to $\widetilde{O}(\rho^{-1/2}\sqrt{T})$ and $\widetilde{O}(\rho^{-1/2}T^{1/3})$, respectively.

Table 1. Comparison of our results to previous studies. Abbreviations: convex → cvx, strongly convex → scvx.

Assumption	Reference	Regret Bound	Privacy	Decentralized?	Projection-free?
$f_{t,i}(\cdot)$: cvx	Li et al. (2018)	$O(n^{5/4}\rho^{-1/2}\epsilon^{-1}\sqrt{T})$	$(\epsilon T, 0)$-DP	✓	×
	Wan et al. (2024a;b)	$\widetilde{O}(n\rho^{-1/4}\sqrt{T})$	–	✓	×
	Corollary 3.7	$\widetilde{O}(n(\rho^{-1/4}+\epsilon^{-1}\rho^{1/4})\sqrt{T})$	$(\epsilon, 0)$-DP	✓	×
	Ene et al. (2021)	$\widetilde{O}(\epsilon^{-1}T^{3/4})$	$(\epsilon, 0)$-DP	×	✓
	Wan et al. (2024b)	$O(nT^{3/4})$	–	✓	✓
	Corollary 3.10	$\widetilde{O}(n(T^{3/4}+\epsilon^{-1}T^{1/4}))$	$(\epsilon, 0)$-DP	✓	✓
$f_{t,i}(\cdot)$: scvx	Li et al. (2018)	$\widetilde{O}(n^{3/2}(\rho\epsilon)^{-1})$	$(\epsilon T, 0)$-DP	✓	×
	Wan et al. (2024a;b)	$\widetilde{O}(n\rho^{-1/2})$	–	✓	×
	Corollary 3.8	$\widetilde{O}(n(\rho^{-1/2}+\epsilon^{-1}))$	$(\epsilon, 0)$-DP	✓	×
	Wan et al. (2024b)	$\widetilde{O}(nT^{2/3})$	–	✓	✓
	Corollary 3.11	$\widetilde{O}(n(T^{2/3}+\epsilon^{-1}))$	$(\epsilon, 0)$-DP	✓	✓

Besides the above progresses, the most relevant works to this paper are about the D-OCO with DP guarantee. Specifically, Li et al. (2018) propose a private variant of D-OGD by simply adding noise to each original decision. Note that this idea is almost the same as the general framework proposed by Jain et al. (2012). Therefore, Li et al. (2018) can only ensure $(\epsilon, 0)$-DP for each round, while achieving sublinear $O(n^{5/4}\rho^{-1/2}\epsilon^{-1}\sqrt{T})$ regret bounds and $\widetilde{O}(n^{3/2}\rho^{-1}\epsilon^{-1})$ for convex and strongly convex functions, respectively. According to the T-fold composition theorem (Dwork & Roth, 2014), such a privacy guarantee will degrade to $(\epsilon T, 0)$-DP over all T rounds in the worst case. Although a series of subsequent algorithms have been proposed (Hou et al., 2019; Lü et al., 2023; Chen et al., 2023; Cheng et al., 2023; Zhang et al., 2024b), both the privacy and regret are not improved. Additionally, these private algorithms are all projection-based. In contrast, this paper proposes a new D-OCO algorithm with $(\epsilon, 0)$-DP over all rounds and improved regret bounds, and also develops a projection-free variant to efficiently handle complex decision sets. A detailed comparison between of previous and our results is summarized in Table 1.

3. Main Results

In this section, we first introduce necessary assumptions and definitions, and then introduce our proposed algorithms as well as the theoretical guarantees. All proofs can be found in the appendix.

3.1. Assumptions and Definitions

Let $\|\mathbf{x}\|$ denote the ℓ_2-norm of any vector $\mathbf{x}$. Following previous studies (Yan et al., 2013; Wan et al., 2024a), we first introduce common assumptions about D-OCO.

Assumption 3.1. The set $\mathcal{K}$ is convex and contains the origin, i.e., $\mathbf{0} \in \mathcal{K}$. Moreover, there exists a constant R such that $\|\mathbf{x}\| \leq R$ for any $\mathbf{x} \in \mathcal{K}$.

Assumption 3.2. For any $t \in [T]$ and $i \in [n]$, the loss function $f_{t,i}(\mathbf{x})$ is G-Lipschitz over $\mathcal{K}$, i.e., $|f_{t,i}(\mathbf{x}) - f_{t,i}(\mathbf{y})| \leq G\|\mathbf{x}-\mathbf{y}\|$ for any $\mathbf{x}, \mathbf{y} \in \mathcal{K}$.

Assumption 3.3. For any $t \in [T]$ and $i \in [n]$, the loss function $f_{t,i}(\mathbf{x})$ is α-strongly convex over $\mathcal{K}$, i.e., $f_{t,i}(\mathbf{y}) \geq f_{t,i}(\mathbf{x}) + \langle \nabla f_{t,i}(\mathbf{x}), \mathbf{y}-\mathbf{x}\rangle + \frac{\alpha}{2}\|\mathbf{x}-\mathbf{y}\|^2$ for any $\mathbf{x}, \mathbf{y} \in \mathcal{K}$.

Assumption 3.4. The communication matrix $P \in \mathbb{R}^{n\times n}$ is supported on the graph $\mathcal{G} = ([n], E)$, symmetric, and doubly stochastic, i.e.,

- $P_{ij} > 0$ only if $(i,j) \in E$ or $i = j$;
- $\sum_{j=1}^n P_{ij} = 1, \forall i \in [n]$ and $\sum_{i=1}^n P_{ij} = 1, \forall j \in [n]$.

Moreover, P is positive semidefinite, and its second largest singular value denoted as $\sigma_2(P)$ is strictly smaller than 1.

Remark. First, when $\alpha = 0$, Assumption 3.3 reduces to the case of general convex functions. Second, the spectral gap of P now can be defined as $\rho = 1 - \sigma_2(P)$.

Then, we provide a formal definition for the differential privacy (DP) of D-OCO (Li et al., 2018; Asi et al., 2023b).

Definition 3.5. Let $\mathcal{F} = (\mathcal{F}(1), \ldots, \mathcal{F}(T))$ with $\mathcal{F}(t) = (f_{t,1}(\cdot), \ldots, f_{t,n}(\cdot))$ be a sequence of loss functions, and $\mathcal{F}'$ be a neighboring sequence generated by replacing one $f_{t,i}(\cdot)$ by $f'_{t,i}(\cdot)$ for some $t \in [T]$ and $i \in [n]$. Let $\mathcal{A}$ denote a randomized D-OCO algorithm that takes $\mathcal{F}$ selected by the adversary Adv during the online process as the input, and outputs a sequence of local decisions denoted as $\mathcal{A} \circ \mathsf{Adv}(\mathcal{F})$ in the space $\mathcal{K}^{n\times T}$. Then, $\mathcal{A}$ is (ϵ, δ)-differentially private if $\Pr[\mathcal{A} \circ \mathsf{Adv}(\mathcal{F}) \in \mathcal{S}] \leq e^{\epsilon}\Pr[\mathcal{A} \circ \mathsf{Adv}(\mathcal{F}') \in \mathcal{S}] + \delta$ holds for any $\mathcal{S} \subseteq \mathcal{K}^{n\times T}$.

3.2. Our PD-FTGL

Inspired by previous studies on the DP of OCO (Jain et al., 2012; Smith & Thakurta, 2013; Kairouz et al., 2021), a natural idea to improve the privacy of D-OCO is to exploit the tree-based private aggregation (Dwork et al., 2010). However, different from FTRL and the approximate FTL in OCO, their decentralized variants (Hosseini et al., 2013; Wan et al., 2024a;b) do not satisfy the precondition of using the tree-based technique.

To be precise, their decentralized variants based on the standard gossip step can be unified to one algorithm called decentralized follow-the-generalized-leader (D-FTGL) (Wan et al., 2024b), which performs the following update

$$\mathbf{x}_i(t+1) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \langle \mathbf{z}_i(t+1), \mathbf{x}\rangle + \frac{t\alpha+2h}{2}\|\mathbf{x}\|^2 \quad (3)$$

for each local learner i, where h is a parameter and $\mathbf{z}_i(t+1)$ is maintained as

$$\mathbf{z}_i(t+1) = \sum_{j=1}^{n} P_{ij}\mathbf{z}_j(t) + (\nabla f_{t,i}(\mathbf{x}_i(t)) - \alpha\mathbf{x}_i(t)). \quad (4)$$

Let $\bar{\mathbf{d}}(t) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{d}_i(t)$, where $\mathbf{d}_i(t) = \nabla f_{t,i}(\mathbf{x}_i(t)) - \alpha\mathbf{x}_i(t)$. We notice that the key role of $\mathbf{z}_i(t+1)$ is to approximate $\sum_{\tau=1}^{t}\bar{\mathbf{d}}(\tau)$, and it thus can be viewed as an extension of $\sum_{\tau=1}^{t}\nabla f_\tau(\mathbf{x}(\tau))$ in (1) and (2). Nonetheless, it cannot be directly rewritten as a partial sum, because the communication in (4) is coupled between adjacent rounds. The same issue also exists in their decentralized variants based on the accelerated gossip strategy (Wan et al., 2024a;b).

To address the above issue, we first introduce a blocking update mechanism (Wan et al., 2020), which divides the total T rounds into L blocks, where L is a parameter and T/L is assumed to be an integer without loss of generality. Then, we only maintain a fixed decision $\mathbf{x}_i(z)$ for each local learner i at all rounds contained by each block z, i.e., $\mathcal{T}_z = \{(z-1)L+1,\ldots,zL\}$. Let $\bar{\mathbf{d}}(z) = \frac{1}{n}\sum_{i=1}^{n}\mathbf{d}_i(z)$, where $\mathbf{d}_i(z) = \sum_{t\in\mathcal{T}_z}(\nabla f_{t,i}(\mathbf{x}_i(z)) - \alpha\mathbf{x}_i(z))$, denote a block-level variant of the previously defined $\bar{\mathbf{d}}(t)$. During each block z, we now can utilize L communications to maintain $\mathbf{d}_i^L(z-1) \approx \bar{\mathbf{d}}(z-1)$ for each local learner i. Intuitively, even using the standard gossip step (Xiao & Boyd, 2004), the partial sum $\sum_{\tau=1}^{z-1}\mathbf{d}_i^L(\tau)$, which is ready to be combined with the tree-based technique, can play a similar role as $\mathbf{z}_i(t+1)$ in (4) for estimating the global information. However, as previously discussed, this approach will result in a suboptimal dependence of the regret on ρ and n.

Thus, inspired by Wan et al. (2024a;b), we adopt the accelerated gossip strategy (Liu & Morse, 2011) to maintain $\mathbf{d}_i^L(z-1)$. Specifically, we perform the following update

$$\mathbf{d}_i^{k+1}(z-1) = (1+\theta)\sum_{j=1}^{n}P_{ij}\mathbf{d}_j^k(z-1) - \theta\mathbf{d}_i^{k-1}(z-1) \quad (5)$$

Algorithm 1 PD-FTGL

1: **Input:** L, θ, h, $\mathcal{D}$, α
2: Initialize $\mathbf{x}_i(1) = \hat{\mathbf{z}}_i(1) = \mathbf{0}, \forall i \in [n]$
3: Create a complete balanced binary tree B_i with $2^{\lceil\log_2(T/L-1)\rceil+1}-1$ nodes for any $i\in[n]$
4: **for** $z = 1, 2, \ldots, T/L$ **do**
5: **for** each local learner $i \in [n]$ **do**
6: Set $\mathbf{d}_i(z) = \mathbf{0}$
7: **for** $k = 0, \ldots, L-1$ **do**
8: Set $t = (z-1)L + k + 1$
9: Play $\mathbf{x}_i(z)$ and query $\nabla f_{t,i}(\mathbf{x}_i(z))$
10: Set $\mathbf{d}_i(z) = \mathbf{d}_i(z) + \nabla f_{t,i}(\mathbf{x}_i(z)) - \alpha\mathbf{x}_i(z)$
11: If $z \geq 2$, update $\mathbf{d}_i^{k+1}(z-1)$ by (5)
12: **end for**
13: Set $\mathbf{d}_i^0(z) = \mathbf{d}_i^{-1}(z) = \mathbf{d}_i(z)$
14: If $z \geq 2$, set $\mathbf{w}_i(z-1) = \mathbf{d}_i^L(z-1)$ and compute $(B_i, \hat{\mathbf{z}}_i(z)) = \text{PrivateSum}(\mathbf{w}_i(z-1), B_i, z-1, \mathcal{D})$ via Algorithm 2
15: Update $\mathbf{x}_i(z+1)$ as in (6)
16: **end for**
17: **end for**

for $k = 0, \ldots, L-1$, where θ is a mixing coefficient, and $\mathbf{d}_i^0(z-1) = \mathbf{d}_i^{-1}(z-1) = \mathbf{d}_i(z-1)$. Then, we can apply the tree-based technique to output a private approximation of $\sum_{\tau=1}^{z-1}\mathbf{d}_i^L(\tau)$, denoted as $\hat{\mathbf{z}}_i(z)$, for each block z. Inspired by (3), we utilize $\hat{\mathbf{z}}_i(z)$ to update the decision as

$$\mathbf{x}_i(z+1) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \langle \hat{\mathbf{z}}_i(z), \mathbf{x}\rangle + \frac{\alpha(z-1)L+2h}{2}\|\mathbf{x}\|^2. \quad (6)$$

Combining with the initialization of $\mathbf{x}_i(1) = \hat{\mathbf{z}}_i(1) = \mathbf{0}$, the detailed procedures of our proposed algorithm are summarized in Algorithm 1, which is named as private decentralized follow-the-generalized-leader (PD-FTGL).

Remark. First, we want to clarify that the *for* loop among these n local learners, i.e., line 5 in Algorithm 1, is mainly used to facilitate presentation, and we actually implement lines 6 to 15 in Algorithm 1 in parallel for these local learners. More specifically, they will synchronously perform the the accelerated gossip strategy via communication to compute $\mathbf{d}_i^{k+1}(z-1)$ in line 11 of Algorithm 1. Second, the implementation of tree-based private aggregation follows Smith & Thakurta (2013). For the sake of completeness, the detailed procedures are outlined in Algorithm 2, where the noise distribution $\mathcal{D}$ will be specified later. As shown in line 14 of Algorithm 1, at the end of each block $z \geq 2$, it is invoked to update the binary tree B_i maintained by each local learner i and compute the private partial sum $\hat{\mathbf{z}}_i(z)$. Finally, it is worth noting that although Wan et al. (2024a;b) have utilized the blocking update mechanism and the accelerated gossip strategy in D-OCO, they propose to maintain $\mathbf{z}_i^L(z)$ to approximate $\sum_{\tau=1}^{z-1}\bar{\mathbf{d}}(\tau)$ directly,

Algorithm 2 PrivateSum (Smith & Thakurta, 2013)

Input: vector $\mathbf{w}_t$, binary tree B, counter t, noise distribution $\mathcal{D}$
Assign $\mathbf{w}_t$ to the t-th leaf node from left to right
Let $L = \{\mathbf{w}_t \to \cdots \to root\}$ be the path from $\mathbf{w}_t$ to root node
Let Λ denote the first left child of B in L
for $\hat{\mathbf{s}} = \mathbf{w}_t \to \cdots \to \Lambda$ **do**
  $\hat{\mathbf{s}} = \hat{\mathbf{s}} + \mathbf{w}_t + \mathbf{b}_s$, where $\mathbf{b}_s \sim \mathcal{D}$
end for
for $\mathbf{s}$ = the next node of Λ in $L \to \cdots \to root$ **do**
  $\mathbf{s} = \mathbf{s} + \mathbf{w}_t$
end for
Let S denote the set contains all top noised nodes, i.e., their parent nodes are not added noise
$\widehat{\mathbf{v}}_t = \sum_{\hat{\mathbf{s}} \in S} \hat{\mathbf{s}}$
Output: updated binary tree B, private partial sum $\widehat{\mathbf{v}}_t$

which needs to set $\mathbf{z}_i^0(z) = \mathbf{z}_i^L(z-1) + \mathbf{d}_i(z-1)$ and $\mathbf{z}_i^{-1}(z) = \mathbf{z}_i^{L-1}(z-1) + \mathbf{d}_i(z-1)$. Therefore, their communication is block-coupled, and $\mathbf{z}_i^L(z)$ cannot be rewritten as a partial sum. In contrast, the communication of our algorithm is block-decoupled, which is critical for exploiting the tree-based technique.

Now, we present the theoretical guarantees of Algorithm 1.

Theorem 3.6. *Under Assumptions 3.1, 3.2, 3.3, and 3.4, for any $i \in [n]$, by setting*

$$\theta = \left(1 + \sqrt{1 - \sigma_2^2(P)}\right)^{-1}, \ L = \left\lceil 4\ln(nT\sqrt{14n})/\sqrt{\rho} \right\rceil \tag{7}$$

and $\mathcal{D} = \mathrm{Lap}^d(0, 6\sqrt{d}\epsilon^{-1}(G + \alpha R)(2 + \log_2 T))$, Algorithm 1 ensures $(\epsilon, 0)$-DP and the following regret bound

$$\begin{aligned}\mathbb{E}[R_{T,i}] \le\ & nhR^2 + nL\sum_{z=1}^{T/L} \frac{12GL(G+2\alpha R)}{\alpha z L + 2h} \\ & + nL\sum_{z=2}^{T/L}\left(\frac{27Gd\,(G+\alpha R)\,(2+\log_2 T)^2}{\epsilon\,(\alpha(z-2)L + 2h)}\right) \\ & + nL\sum_{z=2}^{T/L}\left(\frac{6G(G+\alpha R)}{n\,(\alpha(z-2)L + 2h)}\right).\end{aligned} \tag{8}$$

Remark. From Theorem 3.6, our PD-FTGL can appropriately utilize the Laplace noise at each coordinate, i.e., $\mathrm{Lap}^d(\cdot,\cdot)$, to achieve $(\epsilon, 0)$-DP for D-OCO with both convex and strongly convex functions. Compared with existing algorithms with only $(\epsilon T, 0)$-DP (Li et al., 2018), our algorithm significantly improves the ability of privacy protection. Moreover, combining Theorem 3.6 with the value of α and a suitable h, we can establish specific regret bounds for convex and strongly convex functions, respectively.

Corollary 3.7. *Under Assumptions 3.1, 3.2, 3.3 with $\alpha = 0$, and 3.4, for any $i \in [n]$, by using parameters in Theorem 3.6 and setting $h = G\sqrt{14LT(2+\log_2 T)}/R$, Algorithm 1 ensures*

$$\begin{aligned}\mathbb{E}[R_{T,i}] \le & 7nGR\sqrt{LT(2+\log_2 T)} \\ & + 4\epsilon^{-1}ndGR(2+\log_2 T)^{3/2}\sqrt{T/L}.\end{aligned}$$

Corollary 3.8. *Under Assumptions 3.1, 3.2, 3.3 with $\alpha > 0$, and 3.4, for any $i \in [n]$, by using parameters in Theorem 3.6 and setting $h = \alpha L$, Algorithm 1 ensures*

$$\begin{aligned}\mathbb{E}[R_{T,i}] \le & \ 24\alpha^{-1}nGL(G+2\alpha R)\,(1+\ln(T/L)) \\ & + 27(\alpha\epsilon)^{-1}nd(G+2\alpha R)\,(2+\log_2 T)^3 + n\alpha LR^2.\end{aligned}$$

Remark. By recalling the value of L in (7), Corollaries 3.7 and 3.8 show that our PD-FTGL enjoys a regret bound of $\widetilde{O}(n(\rho^{-1/4} + \epsilon^{-1}\rho^{1/4})\sqrt{T})$ for convex functions and an improved one of $\widetilde{O}(n(\rho^{-1/2} + \epsilon^{-1}))$ for strongly convex functions. These two bounds are respectively tighter than the $\widetilde{O}(n^{5/4}\rho^{-1/2}\epsilon^{-1}\sqrt{T})$ and $\widetilde{O}(n^{3/2}\rho^{-1}\epsilon^{-1})$ regret bounds of existing private D-OCO algorithms (Li et al., 2018). Moreover, we want to emphasize that our advantages include two aspects. First, our regret bounds have a better dependence on both n and ρ, which owes to the exploitation of the accelerated gossip strategy (Liu & Morse, 2011). Second, our regret bounds decouple the negative effects of the decentralized setting and the demand for privacy, i.e., ϵ^{-1} is not multiplicative to $\rho^{-1/4}$ and $\rho^{-1/2}$. As a result, we can nearly match the $\Omega(n\rho^{-1/4}\sqrt{T})$ and $\Omega(n\rho^{-1/2}\log T)$ lower bounds for D-OCO in the non-private case, for $\epsilon = \Omega(\sqrt{\rho})$, instead of only $\epsilon = \Omega(1)$. It is also worth noting this improvement requires a non-trivial analysis that shows the sensitivity of the blocked update mechanism is independent of the block size, instead of simply combining the previous analysis of the tree-based technique (Jain et al., 2012; Smith & Thakurta, 2013; Kairouz et al., 2021; Ene et al., 2021).

3.3. Our Projection-free Algorithm

We proceed to consider the case with complex decision sets, in which the implicit projection operation in (6) could become a computational bottleneck of our PD-FTGL. To address this limitation, we propose a projection-free algorithm by combining PD-FTGL with the classical conditional gradient (CG) method (Frank & Wolfe, 1956; Jaggi, 2013). The detailed procedures of this algorithm are summarized in Algorithm 3, and it is named as private decentralized online conditional gradient (PD-OCG).

Specifically, compared with PD-FTGL, there exist two critical differences in our PD-OCG. First, as shown in lines 2 and 16 of Algorithm 3, we further set $\mathbf{x}_i(2) = \mathbf{0}$ and compute $\mathbf{x}_i(z+1)$ for $z \ge 2$ by applying L iterations of

Algorithm 3 PD-OCG

1: **Input:** L, L', $h, \theta, \mathcal{D}, \alpha$
2: Initialize $\mathbf{x}_i(1) = \mathbf{x}_i(2) = \hat{\mathbf{z}}_i(1) = \mathbf{0}, \forall i \in [n]$
3: Create a complete balanced binary tree B_i with $2^{\lceil \log_2(T/L-1)\rceil+1} - 1$ nodes for any $i \in [n]$
4: **for** $z = 1, 2, \ldots, T/L$ **do**
5: **for** each local learner $i \in [n]$ **do**
6: If $z \geq 2$, define $F_{z,i}(\mathbf{x})$ by (9)
7: Set $\mathbf{d}_i(z) = \mathbf{0}$
8: **for** $k = 0, \ldots, L-1$ **do**
9: Set $t = (z-1)L + k + 1$
10: Play $\mathbf{x}_i(z)$ and query $\nabla f_{t,i}(\mathbf{x}_i(z))$
11: $\mathbf{d}_i(z) = \mathbf{d}_i(z) + (\nabla f_{t,i}(\mathbf{x}_i(z)) - \alpha \mathbf{x}_i(z))$
12: If $z \geq 2$ and $k < L'$, update $\mathbf{d}_i^{k+1}(z-1)$ by (5)
13: **end for**
14: Set $\mathbf{d}_i^0(z) = \mathbf{d}_i^{-1}(z) = \mathbf{d}_i(z)$
15: If $z \geq 2$, set $\mathbf{w}_i(z-1) = \mathbf{d}_i^{L'}(z-1)$ and compute $(B_i, \hat{\mathbf{z}}_i(z)) = \text{PrivateSum}(\mathbf{w}_i(z-1), B_i, z-1, \mathcal{D})$ via Algorithm 2
16: If $z \geq 2$, invoke Algorithm 4 to update the decision as $\mathbf{x}_i(z+1) = \text{CG}(\mathcal{K}, L, F_{z,i}(\mathbf{x}), \mathbf{x}_i(z))$
17: **end for**
18: **end for**

Algorithm 4 CG (Frank & Wolfe, 1956; Jaggi, 2013)

1: **Input:** $\mathcal{K}$, L, $F(\mathbf{x})$, $\mathbf{x}(0)$
2: **for** $t = 0, \ldots, L-1$ **do**
3: $\mathbf{v}(t) \in \operatorname{argmin}_{\mathbf{x} \in \mathcal{K}} \langle \nabla F(\mathbf{x}(t)), \mathbf{x} \rangle$
4: $\sigma(t) = \operatorname{argmin}_{\sigma \in [0,1]} F(\mathbf{x}(t) + \sigma(\mathbf{v}(t) - \mathbf{x}(t)))$
5: $\mathbf{x}(t+1) = \mathbf{x}(t) + \sigma(t)(\mathbf{v}(t) - \mathbf{x}(t))$
6: **end for**
7: **Output:** $\mathbf{x}(L)$

CG to approximate the z-th decision of local learner i in PD-FTGL, i.e.,

$$\mathbf{x}_i(z+1) = \text{CG}(\mathcal{K}, L, F_{z,i}(\mathbf{x}), \mathbf{x}_i(z))$$

where the detailed procedures of CG are outlined in Algorithm 4 for the sake of completeness, and the function $F_{z,i}(\mathbf{x})$ is defined as

$$F_{z,i}(\mathbf{x}) = \langle \hat{\mathbf{z}}_i(z-1), \mathbf{x} \rangle + \frac{\alpha(z-2)L + 2h}{2} \|\mathbf{x}\|^2. \quad (9)$$

In this way, these L iterations can be implemented in parallel to the *for* loop from lines 8 to 13 in Algorithm 3, which is a standard trick to avoid unbalanced computational costs at the end of each block (Wan et al., 2022; 2024b). Moreover, let L' denote the value of L defined in (7). We notice that to control the approximation error of CG, the block size L needs to be sublinear to T, which is much larger than L'. However, from the convergence property of the accelerated gossip strategy (Liu & Morse, 2011), L' is sufficient for maintaining a good $\mathbf{d}_i^{L'}(z)$. Therefore, as in line 12 of Algorithm 3, the second difference is to limit the number of accelerated gossip steps in each block by L', instead of the block size L.

Combining our analysis of PD-FTGL with the convergence property of CG, we establish the following guarantees on the privacy and regret of the projection-free algorithm.

Theorem 3.9. *Under Assumptions 3.1, 3.2, 3.3, and 3.4, for any $i \in [n]$, by setting*

$$\theta = \left(1 + \sqrt{1 - \sigma_2^2(P)}\right)^{-1}, \; L' = \left\lceil 4\ln(nT\sqrt{14n})/\sqrt{\rho} \right\rceil \quad (10)$$

and $\mathcal{D} = \text{Lap}^d(0, 6\sqrt{d}\epsilon^{-1}(G + \alpha R)(2 + \log_2 T))$, Algorithm 3 ensures $(\epsilon, 0)$-DP and the following regret bound

$$\begin{aligned}\mathbb{E}[R_{T,i}] \leq\; & nhR^2 + nL \sum_{z=1}^{T/L} \frac{24GL(G + 2\alpha R)}{\alpha z L + 2h} \\ & + nL \sum_{z=3}^{T/L} \left(\frac{27dG\,(G + \alpha R)\,(2 + \log_2 T)^2}{\epsilon\,(\alpha(z-3)L + 2h)} \right) \\ & + nL \sum_{z=3}^{T/L} \left(\frac{12GR}{\sqrt{L}} + \frac{6G(G + \alpha R)}{n\,(\alpha(z-3)L + 2h)} \right).\end{aligned} \quad (11)$$

Remark. From Theorem 3.9, our projection-free algorithm can ensure the same $(\epsilon, 0)$-DP as PD-FTGL, which is because the exploitation of CG does not affect the privacy. Then, combining Theorem 3.9 with the value of α and suitable h and L, we can establish specific regret bounds for convex and strongly convex functions, respectively.

Corollary 3.10. *Under Assumptions 3.1, 3.2, 3.3 with $\alpha = 0$, and 3.4, for any $i \in [n]$, by using parameters in Theorem 3.9 and setting $h = \sqrt{15LTG/R}, L = \sqrt{T}$, Algorithm 3 ensures*

$$\mathbb{E}[R_{T,i}] \leq 21nGRT^{3/4} + 4\epsilon^{-1}ndGRT^{1/4}\,(2 + \log_2 T)^2.$$

Corollary 3.11. *Under Assumptions 3.1, 3.2, 3.3 with $\alpha > 0$, and 3.4, for any $i \in [n]$, by using parameters in Theorem 3.9 and setting $h = \alpha L, L = T^{2/3}\ln^{-2/3} T, C_T = \ln^{-2/3} T + \ln^{1/3} T$, Algorithm 3 ensures*

$$\begin{aligned}\mathbb{E}[R_{T,i}] \leq\; & 36\alpha^{-1}nGT^{2/3}C_T(G + 2\alpha R) \\ & + n\alpha R^2 T^{2/3}\ln^{-2/3} T + 12nGRT^{2/3}\ln^{1/3} T \\ & + 27(\alpha\epsilon)^{-1}ndG(G + \alpha R)\,(2 + \log_2 T)^3.\end{aligned}$$

Remark. First, we note that Corollaries 3.10 and 3.11 implicitly assume that L' in (10) is smaller than $L = \sqrt{T}$ and $L = T^{2/3}\ln^{-2/3} T$, respectively. This assumption is reasonable because T in D-OCO is commonly much larger than other problem constants. Then, from these corollaries, our PD-OCG can achieve $\widetilde{O}(n(T^{3/4} + \epsilon^{-1}T^{1/4}))$ and

$\widetilde{O}(n(T^{2/3}+\epsilon^{-1}))$ regret bounds for convex and strongly convex functions, respectively. Interestingly, if $n=1$ and $\alpha=0$, our PD-OCG can reduce to the existing projection-free algorithm of Ene et al. (2021) with $(\epsilon,0)$-DP for OCO. However, they only establish an $O(\epsilon^{-1}T^{3/4}\log^2 T)$ regret bound for convex functions, which is much worse than our bound in the case of $n=1$, i.e., $\widetilde{O}(T^{3/4}+\epsilon^{-1}T^{1/4})$. This gap is because they simply bound the sensitivity of the blocking update mechanism with the block size, and then need to add a corresponding scale of noise to ensure the privacy. In contrast, as discussed before, our novel analysis on the sensitivity implies that the scale of noise does not need to be proportional to the block size. Moreover, we want to emphasize that such an analysis brings more significant benefits to the projection-free algorithm. Specifically, the two regret bounds of our PD-OCG can nearly match the $O(nT^{3/4})$ and $\widetilde{O}(nT^{2/3})$ regret bounds of the best projection-free D-OCO algorithm in the non-private case (Wan et al., 2024b) for $\epsilon=\Omega(T^{-1/2})$ and $\epsilon=\Omega(T^{-2/3})$, respectively. Finally, similar to Wan et al. (2024b), the communication complexity of our projection-free algorithm is $O(TL'/L)$, which becomes $\widetilde{O}(\rho^{-1/2}\sqrt{T})$ and $\widetilde{O}(\rho^{-1/2}T^{1/3})$ according to the parameters in Corollaries 3.10 and 3.11, respectively.

4. Experiments

In this section, we conduct simulation experiments with convex functions to verify the performance of our algorithms.

4.1. Experimental Setup

Following previous studies (Zhang et al., 2017; Wan et al., 2020), the problem of decentralized online multiclass classification is considered. Specifically, let p and q denote the number of features and classes, respectively. The decision set of is defined as $\mathcal{K}=\{X\in\mathbb{R}^{q\times p}\mid \|X\|_*\leq\tau\}$, where $\|X\|_*$ denotes the trace norm of X and $\tau=10$. At each round $t\in[T]$, the loss function of each local learner i is defined as

$$f_{t,i}(X)=\log\left(1+\sum_{\ell\neq y_i(t)}e^{\left(\mathbf{x}_\ell^\top\mathbf{e}_i(t)-\mathbf{x}_{y_i(t)}^\top\mathbf{e}_i(t)\right)}\right)$$

where $\mathbf{e}_i(t)\in\mathbb{R}^p$ and $y_i(t)\in[q]$ denote the feature vector and class label of the single example at round t, $\mathbf{x}_i^\top$ is the i-th row of the matrix X for any $i\in[q]$, and $\ell=\operatorname{argmax}_{\ell\in[q]}\mathbf{x}_\ell^\top\mathbf{e}_i(t)$ is the predicted class label based on the matrix X. Then, let $X_i(t)\in\mathcal{K}$ denote the decision of each local learner i at each round t. Its average loss at this round equals to $\mathrm{AL}(t,i)=\frac{1}{tn}\sum_{\tau=1}^{t}\sum_{j=1}^{n}f_{\tau,j}(X_i(\tau))$. Moreover, we use two publicly available datasets—letter and poker from the LIBSVM repository (Chang & Lin, 2011), and their details are summarized in Table 2. For letter, we construct a larger one including $10n$ copies of the original data, and evenly distribute them among the n local learners, which implies that $T=150000$. For poker, we construct a larger one including n copies of the original data, and distribute them in the same way, which implies that $T=1000000$. By default, we set $n=9$ and use the complete graph, i.e., all nodes are connected to others.

Table 2. Summary of Datasets.

Dataset	# Features	# Classes	# Examples
letter	16	26	15000
poker	10	10	1000000

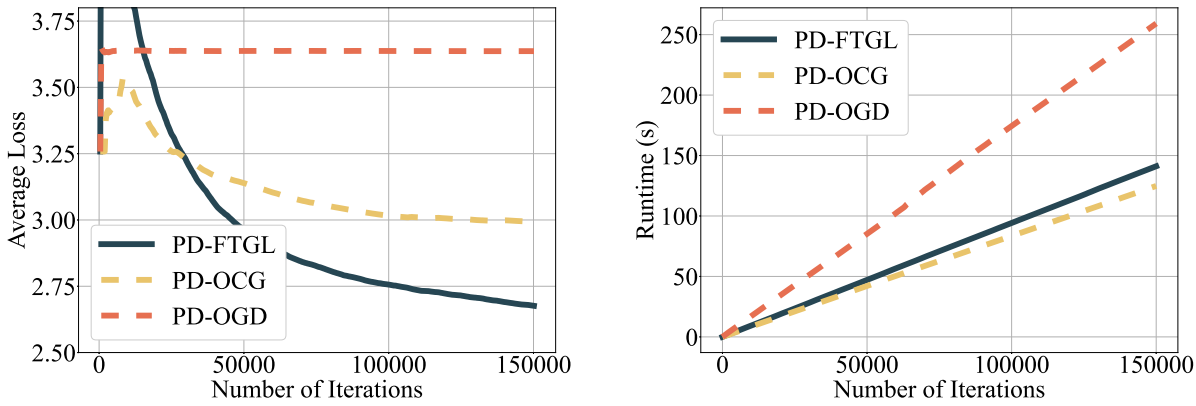
The baseline in our experiments is the existing private variant of D-OGD (PD-OGD) from Li et al. (2018). Note that the original PD-OGD can only achieve $(\epsilon,0)$-DP for each round, instead of all T rounds. To make a fair comparison, we increase the scale of noises added into PD-OGD to achieve $(\epsilon,0)$-DP over all T rounds. Moreover, following previous studies on DP (Abadi et al., 2016; Kairouz et al., 2021), we substitute the original gradient $\nabla f_{t,i}(X_i(t))$ by a clipping gradient defined as

$$\mathbf{g}_{\text{clip}}(t,i)=\nabla f_{t,i}(X_i(t))/\max\left(1,\frac{\|\nabla f_{t,i}(X_i(t))\|_F}{C}\right)$$

where the constant C is tuned from the set $\{0.01,0.1,1,10\}$ to obtain the best performance, and $\|\cdot\|_F$ denotes the Frobenius norm. In this way, the Frobenius norm of the clipping gradient is bounded by C, which simplifies the setting of noise scale when using the tree-based private aggregation. Additionally, for the parameter h of our algorithms, we multiply the theoretical value by a constant tuned from the set $\{0.0001,0.0005,0.001,0.005,\ldots,10,50\}$ for the best performance. Other parameters of all algorithms are set as what their corresponding theories suggest.

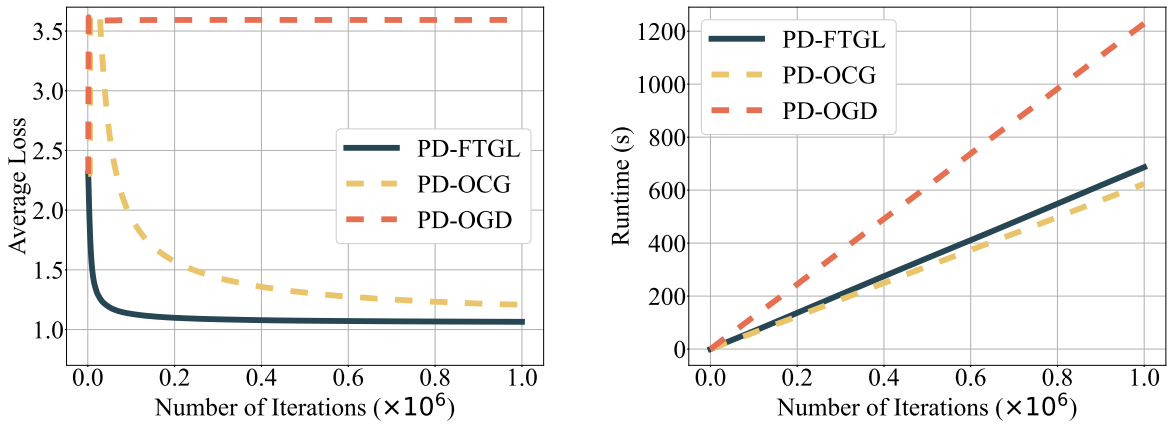
4.2. Experimental Results

Figures 1 and 2 show the comparisons of average loss and runtime of all algorithms with $\epsilon=10$ on letter and poker, respectively. First, in terms of average loss, our PD-FTGL and PD-OCG are much better than PD-OGD. This is because the noise scale required by PD-OGD to achieve the same privacy level is much larger than our algorithms. Second, the average loss of our PD-OCG is worse than that of our PD-FTGL on both datasets, which can be viewed as the price of the projection-free property and is consistent with our theoretical results. Moreover, from the comparison of runtime, PD-OGD is much worse than our PD-FTGL. It is worth noting that this is because PD-OGD performs the projection operation once per round, whereas our PD-FTGL only needs one projection per block. Additionally, we also notice that our PD-OCG can be implemented faster than our



(a) Average Loss (b) Runtime

Figure 1. Comparison of all algorithms with $\epsilon = 10$ on the letter dataset, where we use the complete graph with $n = 9$ nodes.



(a) Average Loss (b) Runtime

Figure 2. Comparison of all algorithms with $\epsilon = 10$ on the poker dataset, where we use the complete graph with $n = 9$ nodes.

PD-FTGL on both datasets, which verifies the benefit of the projection-free property. Besides these experimental results, we have also evaluate the performance of our algorithms under different privacy levels, network sizes, and topologies, which can be found in Appendix B.

5. Conclusion and Future Work

In this paper, we propose the first algorithm to achieve $(\epsilon, 0)$-DP over all T rounds for D-OCO, namely PD-FTGL, and demonstrate that it enjoys $\widetilde{O}(n(\rho^{-1/4} + \epsilon^{-1}\rho^{1/4})\sqrt{T})$ and $\widetilde{O}(n(\rho^{-1/2} + \epsilon^{-1}))$ regret bounds for convex and strongly convex functions, respectively. These regret bounds are tighter than those achieved by existing D-OCO algorithms with only $(\epsilon T, 0)$-DP over all T rounds, and nearly match the lower bounds for D-OCO in the non-private case for $\epsilon = \Omega(\sqrt{\rho})$. Furthermore, to efficiently deal with complex decision sets, we propose a projection-free variant of PD-FTGL, and establish $\widetilde{O}(n(T^{3/4} + \epsilon^{-1}T^{1/4}))$ and $\widetilde{O}(n(T^{2/3} + \epsilon^{-1}))$ regret bounds for convex and strongly convex functions, respectively. These results can nearly match the regret bounds of the best existing projection-free D-OCO algorithm in the non-private case for $\epsilon = \Omega(T^{-1/2})$ and $\epsilon = \Omega(T^{-2/3})$, respectively. However, there still exist some open problems. First, it is appealing to investigate whether our PD-FTGL is nearly optimal for any privacy level ϵ or not. To this end, we need to establish general lower bounds for D-OCO with the DP guarantee, which seems highly non-trivial since there still lack such results even in the special case with $n = 1$. Second, as later discussed in Appendix A, it is possible to improve the regret of OCO with the DP guarantee via a lazy OCO technique if the adversary is oblivious. Thus, it is also natural to extend this improvement into the D-OCO setting.

Acknowledgements

This work was partially supported by National Natural Science Foundation of China (62306275), Zhejiang Provincial Natural Science Foundation of China (LHZSD24F020001, LMS25F030002), Ningbo Yongjiang Talent Introduction Programme (2023A-193-G), and Ningbo Natural Science Foundation (2024J206). The authors would also like to thank the anonymous reviewers for their helpful comments.

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

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.

Abernethy, J., Bartlett, P. L., Rakhlin, A., and Tewari, A. Optimal strategies and minimax lower bounds for online convex games. In *Proceedings of the 21st Annual Conference on Learning Theory*, pp. 415–423, 2008.

Agarwal, N. and Singh, K. The price of differential privacy for online learning. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 32–40, 2017.

Agarwal, N., Kale, S., Singh, K., and Thakurta, A. Differentially private and lazy online convex optimization. In *Proceedings of the 36th Conference on Learning Theory*, pp. 4599–4632, 2023.

Agarwal, N., Kale, S., Singh, K., and Thakurta, A. G. Improved differentially private and lazy online convex optimization. In *Proceedings of the 41st International Conference on Machine Learning*, pp. 343–361, 2024.

Asi, H., Feldman, V., Koren, T., and Talwar, K. Near-optimal algorithms for private online optimization in the realizable regime. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 1107–1120, 2023a.

Asi, H., Feldman, V., Koren, T., and Talwar, K. Private online prediction from experts: Separations and faster rates. In *Proceedings of the 36th Conference on Learning Theory*, pp. 674–699, 2023b.

Asi, H., Koren, T., Liu, D., and Talwar, K. Private online learning via lazy algorithms. In *Advances in Neural Information Processing Systems 37*, pp. 74856–74889, 2024.

Chang, C.-C. and Lin, C.-J. LIBSVM: A library for support vector machines. *ACM transactions on intelligent systems and technology*, 2(3):1–27, 2011.

Chen, L., Zhang, M., and Karbasi, A. Projection-free bandit convex optimization. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, pp. 2047–2056, 2019.

Chen, L., Ding, X., Zhou, P., and Jin, H. Distributed dynamic online learning with differential privacy via path-length measurement. *Information Sciences*, 630(C):135–157, 2023.

Cheng, H., Liao, X., and Li, H. Distributed online private learning of convex nondecomposable objectives. *IEEE Transactions on Network Science and Engineering*, 11(2): 1716–1728, 2023.

Duchi, J. C., Agarwal, A., and Wainwright, M. J. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic control*, 57(3):592–606, 2011.

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. In *Proceedings of the 3rd Conference on Theory of Cryptography*, pp. 265–284, 2006.

Dwork, C., Naor, M., Pitassi, T., and Rothblum, G. N. Differential privacy under continual observation. In *Proceedings of the 42th ACM Symposium on Theory of Computing*, pp. 715–724, 2010.

Ene, A., Nguyen, H. L., and Vladu, A. Projection-free bandit optimization with privacy guarantees. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pp. 7322–7330, 2021.

Frank, M. and Wolfe, P. An algorithm for quadratic programming. *Naval Research Logistics Quarterly*, 3(1–2): 95–110, 1956.

Fredrikson, M., Jha, S., and Ristenpart, T. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pp. 1322–1333, 2015.

Garber, D. and Hazan, E. A linearly convergent variant of the conditional gradient algorithm under strong convexity, with applications to online and stochastic optimization. *SIAM Journal on Optimization*, 26(3):1493–1528, 2016.

Garber, D. and Kretzu, B. Improved regret bounds for projection-free bandit convex optimization. In *Proceedings of the 23rd International Conference on Artificial Intelligence and Statistics*, pp. 2196–2206, 2020.

Garber, D. and Kretzu, B. New projection-free algorithms for online convex optimization with adaptive regret guarantees. In *Proceedings of the 35th Conference on Learning Theory*, pp. 2326–2359, 2022.

Garber, D. and Kretzu, B. Projection-free online exp-concave optimization. In *Proceedings of the 36th Conference on Learning Theory*, pp. 1259–1284, 2023.

Hazan, E. Introduction to online convex optimization. *Foundations and Trends® in Optimization*, 2(3-4):157–325, 2016.

Hazan, E. and Kale, S. Projection-free online learning. In *Proceedings of the 29th International Conference on Machine Learning*, pp. 1843–1850, 2012.

Hazan, E. and Minasyan, E. Faster projection-free online learning. In *Proceedings of the 33rd Conference on Learning Theory*, pp. 1877–1893, 2020.

Hazan, E., Agarwal, A., and Kale, S. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69(2):169–192, 2007.

Hosseini, S., Chapman, A., and Mesbahi, M. Online distributed optimization via dual averaging. In *Proceedings of the 52nd IEEE Conference on Decision and Control*, pp. 1484–1489, 2013.

Hou, M., Li, D., Wu, X., and Shen, X. Differential privacy of online distributed optimization under adversarial nodes. In *Proceedings of the 2019 Chinese Control Conference*, pp. 2172–2177, 2019.

Jaggi, M. Revisiting Frank-Wolfe: Projection-free sparse convex optimization. In *Proceedings of the 30th International Conference on Machine Learning*, pp. 427–435, 2013.

Jain, P., Kothari, P., and Thakurta, A. Differentially private online learning. In *Proceedings of the 25th Conference on Learning Theory*, pp. 24–1, 2012.

Kairouz, P., Mcmahan, B., Song, S., Thakkar, O., Thakurta, A., and Xu, Z. Practical and private (deep) learning without sampling or shuffling. In *Proceedings of the 38th International Conference on Machine Learning*, pp. 5213–5225, 2021.

Kalhan, D. S., Bedi, A. S., Koppel, A., Rajawat, K., Hassani, H., Gupta, A. K., and Banerjee, A. Dynamic online learning via Frank-Wolfe algorithm. *IEEE Transactions on Signal Processing*, 69:932–947, 2021.

Kretzu, B. and Garber, D. Revisiting projection-free online learning: the strongly convex case. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, pp. 3592–3600, 2021.

Levy, K. and Krause, A. Projection free online learning over smooth sets. In *Proceedings of the 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1458–1466, 2019.

Li, C., Zhou, P., Xiong, L., Wang, Q., and Wang, T. Differentially private distributed online learning. *IEEE Transactions on Knowledge and Data Engineering*, 30(8):1440–1453, 2018.

Li, X., Xie, L., and Li, N. A survey of decentralized online learning. *Annual Reviews in Control*, 56(100904), 2023.

Liu, J. and Morse, A. S. Accelerated linear iterations for distributed averaging. *Annual Reviews in Control*, 35(2): 160–165, 2011.

Lü, Q., Zhang, K., Deng, S., Li, Y., Li, H., Gao, S., and Chen, Y. Privacy-preserving decentralized dual averaging for online optimization over directed networks. *IEEE Transactions on Industrial Cyber-Physical Systems*, 1: 79–91, 2023.

Mhammedi, Z. Efficient projection-free online convex optimization with membership oracle. In *Proceedings of the 35th Conference on Learning Theory*, pp. 5314–5390, 2022.

Shalev-Shwartz, S. and Singer, Y. A primal-dual perspective of online learning algorithms. *Machine Learning*, 69:115–142, 2007.

Shokri, R., Stronati, M., Song, C., and Shmatikov, V. Membership inference attacks against machine learning models. In *Proceedings of the 2017 IEEE Symposium on Security and Privacy*, pp. 3–18, 2017.

Smith, A. and Thakurta, A. (Nearly) optimal algorithms for private online learning in full-information and bandit settings. In *Advances in Neural Information Processing Systems 26*, pp. 2733–2741, 2013.

Wan, Y. and Zhang, L. Projection-free online learning over strongly convex sets. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pp. 10076–10084, 2021.

Wan, Y., Tu, W.-W., and Zhang, L. Projection-free Distributed Online Convex Optimization with $O(\sqrt{T})$ Communication Complexity. In *Proceedings of the 37th International Conference on Machine Learning*, pp. 9818–9828, 2020.

Wan, Y., Wang, G., and Zhang, L. Projection-free distributed online learning with strongly convex losses. *arXiv:2103.11102v1*, 2021a.

Wan, Y., Xue, B., and Zhang, L. Projection-free online learning in dynamic environments. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, pp. 10067–10075, 2021b.

Wan, Y., Wang, G., Tu, W.-W., and Zhang, L. Projection-free distributed online learning with sublinear communication complexity. *Journal of Machine Learning Research*, 23(172):7742–7794, 2022.

Wan, Y., Zhang, L., and Song, M. Improved dynamic regret for online frank-wolfe. In *Proceedings of the 36th Annual Conference on Learning Theory*, pp. 3304–3327, 2023.

Wan, Y., Wei, T., Song, M., and Zhang, L. Nearly optimal regret for decentralized online convex optimization. In *Proceedings of the 37th Conference on Learning Theory*, pp. 4862–4888, 2024a.

Wan, Y., Wei, T., Xue, B., Song, M., and Zhang, L. Optimal and efficient algorithms for decentralized online convex optimization. *arXiv:2402.09173*, 2024b.

Wang, Y., Yang, W., Jiang, W., Lu, S., Wang, B., Tang, H., Wan, Y., and Zhang, L. Non-stationary projection-free online learning with dynamic and adaptive regret guarantees. In *Proceedings of the 38th AAAI Conference on Artificial Intelligence*, pp. 15671–15679, 2024.

Watts, D. J. and Strogatz, S. H. Collective dynamics of 'small-world' networks. *Nature*, 393:440–442, 1998.

Xiao, L. and Boyd, S. Fast linear iterations for distributed averaging. *Systems & Control Letters*, 53(1):65–78, 2004.

Yan, F., Sundaram, S., Vishwanathan, S., and Qi, Y. Distributed autonomous online learning: Regrets and intrinsic privacy-preserving properties. *IEEE Transactions on Knowledge and Data Engineering*, 25(11):2483–2493, 2013.

Ye, H., Luo, L., Zhou, Z., and Zhang, T. Multi-consensus decentralized accelerated gradient descent. *Journal of Machine Learning Research*, 24(306):1–50, 2023.

Zhang, C., Wang, Y., Tian, P., Cheng, X., Wan, Y., and Song, M. Projection-free bandit convex optimization over strongly convex sets. In *Proceedings of the 28th Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 118–129, 2024a.

Zhang, K., Lü, Q., Liao, X., and Li, H. Zeroth-order decentralized dual averaging for online optimization with privacy consideration. *IEEE Transactions on Emerging Topics in Computational Intelligence*, 8(6):3754–3766, 2024b.

Zhang, W., Zhao, P., Zhu, W., Hoi, S. C., and Zhang, T. Projection-free distributed online learning in networks. In *Proceedings of the 34th International Conference on Machine Learning*, pp. 4054–4062, 2017.

Zhou, H., Xu, Z., and Tzoumas, V. Efficient online learning with memory via Frank-Wolfe optimization: Algorithms with bounded dynamic regret and applications to control. In *Proceedings of the 62nd IEEE Conference on Decision and Control*, 2023.

Zinkevich, M. Online convex programming and generalized infinitesimal gradient ascent. In *Proceedings of the 20th International Conference on Machine Learning*, pp. 928–936, 2003.

A. Additional Related Work

Here, we first introduce some recent efforts for improving the regret of private OCO algorithms under additional assumptions on functions, decision sets, or the adversary. Then, we briefly review the progress of projection-free OCO algorithms.

A.1. Improved Private Algorithms for OCO

Similar to previous tree-based private algorithms, Agarwal & Singh (2017) propose an $(\epsilon, 0)$-DP algorithm for online linear optimization to achieve an improved regret bound of $\widetilde{O}(\sqrt{T} + \epsilon^{-1})$. Very recently, several DP algorithms have been proposed to achieve more desirable regret bounds for OCO with an oblivious adversary. Specifically, Asi et al. (2023a) consider a oblivious and realizable setting where a zero-loss solution exists, and demonstrate that the regret of the algorithm in Kairouz et al. (2021) can be significantly improved for convex (and smooth) functions. By exploiting recent advances on a lazy OCO setting where the decision can only be switched under a limited budget, Asi et al. (2023b) develop an (ϵ, δ)-DP algorithm to achieve an $\widetilde{O}(\sqrt{T} + \epsilon^{-1}T^{1/3})$ regret bound for convex functions. Following the lazy OCO technique, Agarwal et al. (2023) propose an improved (ϵ, δ)-DP algorithm that can reduce the dependence of the regret bound of Asi et al. (2023b) on the dimensionality. Note that the original algorithm of Agarwal et al. (2023) requires the smoothness of functions, which is removed by Agarwal et al. (2024) with careful modifications. Furthermore, Asi et al. (2024) consider the high privacy regime with $\epsilon \ll 1$, and achieve the (ϵ, δ)-DP and $\widetilde{O}(\sqrt{T} + \epsilon^{-2/3}T^{1/3})$ regret bound via the lazy OCO technique.

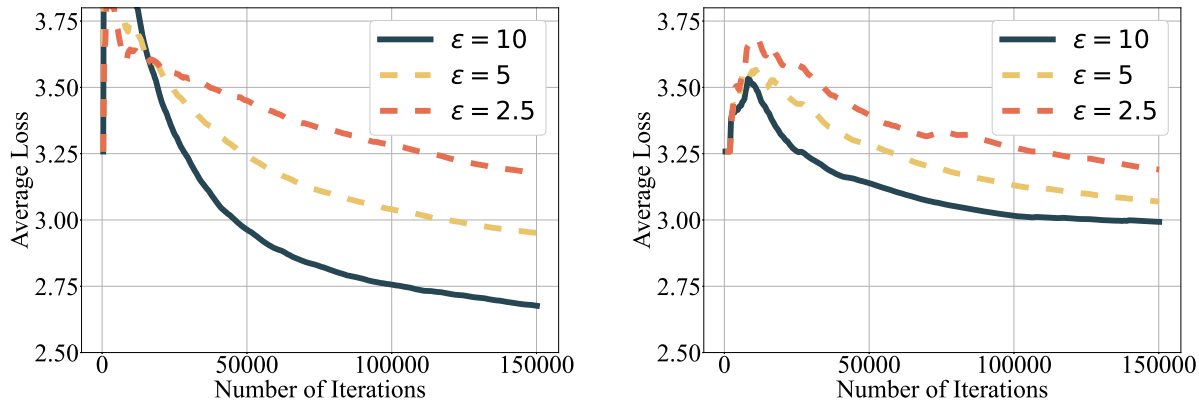
A.2. Projection-free Algorithms for OCO

The first projection-free OCO algorithm is proposed by Hazan & Kale (2012), which is called online conditional gradient (OCG) and can achieve an $O(T^{3/4})$ regret bound. Following this work, there is a growing research interest in improving the regret of projection-free OCO for special types of functions and decision sets (Garber & Hazan, 2016; Kretzu & Garber, 2021; Wan & Zhang, 2021; Levy & Krause, 2019; Hazan & Minasyan, 2020; Garber & Kretzu, 2022; 2023; Mhammedi, 2022). Among these works, the closest one to this paper is Wan & Zhang (2021), which develops a variant of OCG to achieve an $O(T^{2/3})$ regret bound for strongly convex functions. Moreover, several projection-free algorithms have been extended into the bandit setting (Chen et al., 2019; Garber & Kretzu, 2020; Kretzu & Garber, 2021; Zhang et al., 2024a), where only the loss value is revealed to the learner, instead of the full-information of functions. Recently, there has also been a surge of research interest in developing projection-free OCO algorithms for non-stationary environments (Kalhan et al., 2021; Wan et al., 2021b; 2023; Zhou et al., 2023; Wang et al., 2024), in which the regret is replaced with two more suitable metrics called adaptive regret and dynamic regret.

B. Additional Experimental Results

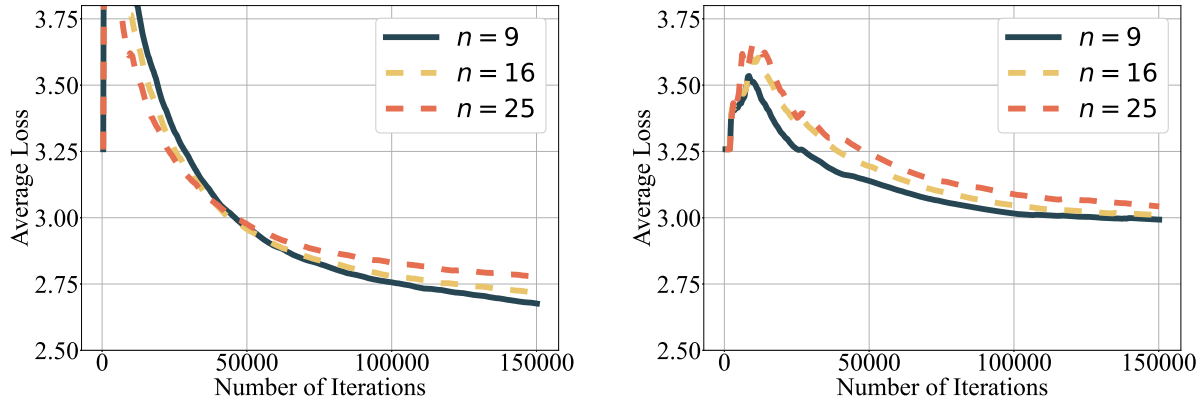
Note that these additional experiments are all conducted on the letter dataset. Moreover, we set $\epsilon = 10$, $n = 9$, and use the complete graph by default. As shown in Figure 3, we first compare the average loss of our PD-FTGL and PD-OCG with different privacy levels, i.e., $\epsilon \in \{2.5, 5, 10\}$. We find that our two algorithms perform better as the value of ϵ increases. Moreover, compared with PD-FTGL, the average loss of PD-OCG varies less among different ϵ, which implies that the privacy level has a weaker impact on PD-OCG. This is consistent with our theoretical results for convex functions, since the privacy-dependent term in the regret bound of PD-FTGL is $\widetilde{O}(n\epsilon^{-1}\rho^{1/4}\sqrt{T})$ and that of PD-OCG is $\widetilde{O}(n\epsilon^{-1}T^{1/4})$. Then, as shown in Figure 4, we compare the average loss of our algorithms with different network sizes, i.e., $n \in \{9, 16, 25\}$. As the network size increases, the average losses of our two algorithms become slightly worse, which is consistent with the dependence of our regret bounds on n.

Finally, we conduct experiments to verify the effect of different network topologies on our algorithms, including the complete graph, cycle graph and Watts-Strogatz (abbr. ws) graph (Watts & Strogatz, 1998). Among these graphs, the complete graph has the largest spectral gap ρ. In contrast, the cycle graph, where each local learner is connected to only two neighbors, has the smallest ρ. The ws graph is generated by a random technique with tunable parameters including the average degree and rewiring probability. We set the average degree of the graph to be 6 and the rewiring probability to be 0.5. As shown in Figures 5(a) and 5(b), for the graph with larger ρ, the average loss of PD-FTGL becomes worse when $\epsilon = 10$, but becomes better when $\epsilon = 1000$. These opposite effects are due to the fact that the $\widetilde{O}(n(\rho^{-1/4} + \epsilon^{-1}\rho^{1/4})\sqrt{T})$ regret of PD-FTGL is dominant by $\widetilde{O}(n\epsilon^{-1}\rho^{1/4}\sqrt{T})$ and $\widetilde{O}(n\rho^{-1/4}\sqrt{T})$ for $\epsilon = 10$ and $\epsilon = 1000$, respectively. From Figures 5(c) and 5(d), the average loss of PD-OCG is almost unaffected by the topology of graph for both values of ϵ, which is reasonable because the regret bound of PD-OCG does not depend on ρ.



(a) PD-FTGL (b) PD-OCG

Figure 3. Comparison of the average loss of our algorithms with different ϵ, where we use the complete graph with $n = 9$ nodes.



(a) PD-FTGL (b) PD-OCG

Figure 4. Comparison of the average loss of our algorithms with $\epsilon = 10$ and different n, where we use the complete graph.

C. Proof of Privacy Guarantee in Theorem 3.6

We define $X = (X(1), \ldots, X(T/L))$, where $X(z) = (\mathbf{x}_1(z), \ldots, \mathbf{x}_n(z))$ for $z \in [T/L]$. For any variable $\mathbf{x}$ which is generated under loss function sequence $\mathcal{F}$, we use $\mathbf{x}'$ to denote the same variable that is generated under $\mathcal{F}'$.

It is not hard to verify that

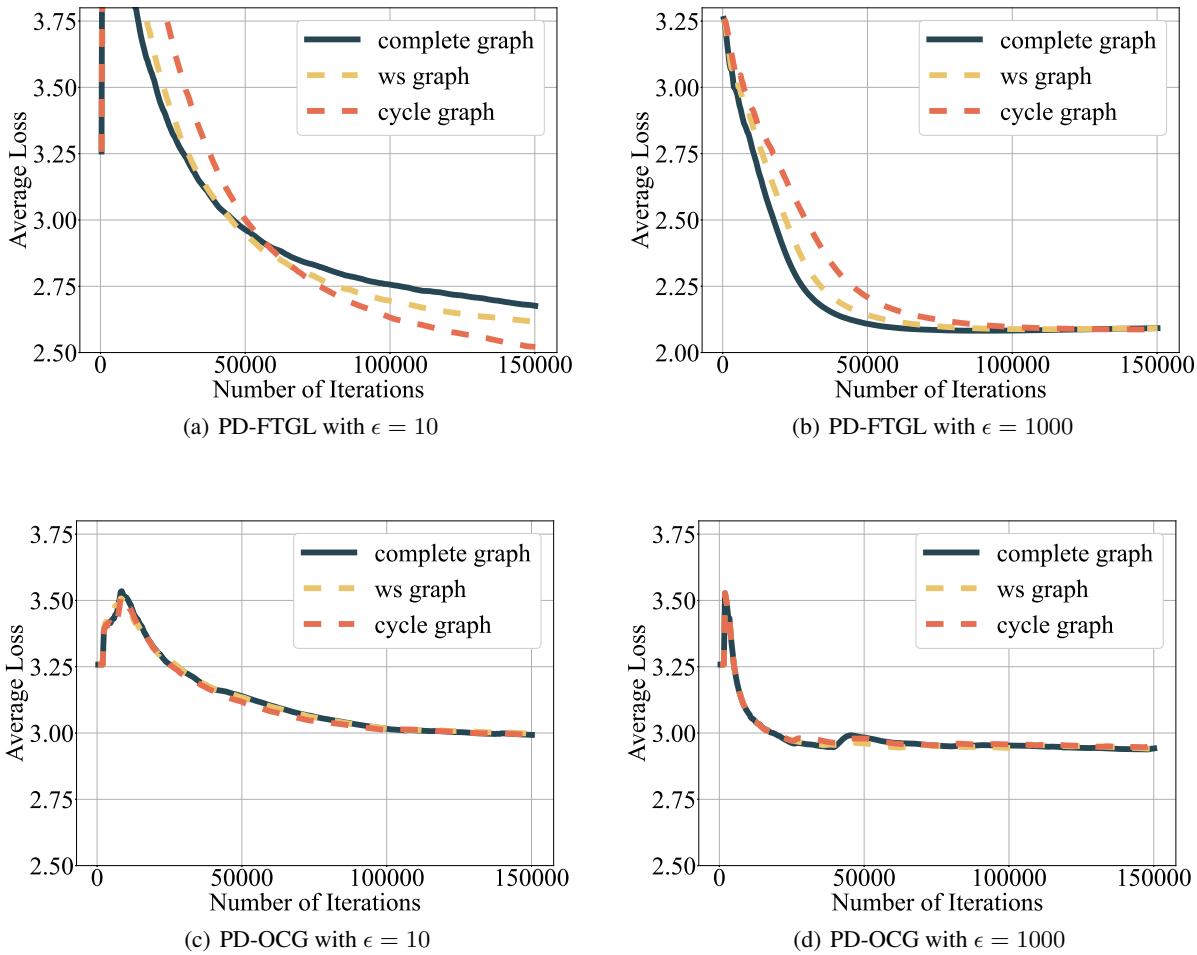
$$\max_{\mathcal{S} \subset \mathcal{K}^{n \times T/L}} \frac{\Pr[X \in \mathcal{S}]}{\Pr[X' \in \mathcal{S}]} = \max_{\xi \in \mathcal{K}^{n \times T/L}} \frac{\Pr[X = \xi]}{\Pr[X' = \xi]}$$

where $\xi = (\xi(1), \ldots, \xi(T/L))$ with $\xi(z)$ denoting any possible value of $X(z)$.

Thus, to prove the $(\epsilon, 0)$-DP guarantee of Algorithm 1, we only need to prove that

$$\frac{\Pr[X = \xi]}{\Pr[X' = \xi]} = \prod_{z=1}^{T/L} \frac{\Pr[X(z) = \xi(z) \mid X(1) = \xi(1), X(2) = \xi(2), X(3) = \xi(3), \ldots, X(z-1) = \xi(z-1)]}{\Pr[X'(z) = \xi(z) \mid X'(1) = \xi(1), X'(2) = \xi(2), X'(3) = \xi(3), \ldots, X'(z-1) = \xi(z-1)]} \leq e^{\epsilon}.$$

Specially, for $z = 1$ the decisions of all local learners are set to be $\mathbf{0}$, which is independent with loss function. As a result, the privacy can not be leaked. For $z \geq 2$, since each $\mathbf{x}_i(z)$ depends on $\hat{\mathbf{z}}_i(z-1)$, and $\hat{\mathbf{z}}_i(z-1)$ is the combination of nodes on the private binary tree in a specific form, according to lines 11 to 12 in Algorithm 2, we can say that all binary trees



(a) PD-FTGL with $\epsilon = 10$

(b) PD-FTGL with $\epsilon = 1000$

(c) PD-OCG with $\epsilon = 10$

(d) PD-OCG with $\epsilon = 1000$

Figure 5. Comparison of the average loss of our algorithms with different graphs, where we set $n = 9$.

maintained by the local learners jointly determine the decision set X. Note that in each block, each local learner operates on the same nodes in its own private tree via Algorithm 2 in parallel. For brevity, we use $m = \lceil \log_2(T/L - 1) \rceil$ and label all the nodes of a binary tree in an post-order traversal order as $1, \ldots, 2^{m+1} - 1$. Let $\hat{\mathbf{s}}_i(r)$ denote the noised value in the r-th node of the binary tree maintained by learner i for any $i \in [n]$ and $r \in [2^{m+1} - 1]$. We use $\hat{S}(r) = (\hat{\mathbf{s}}_1(r), \ldots, \hat{\mathbf{s}}_n(r))$ to denote the set of noised values at node position r on the privacy trees of all local learners $i \in [n]$ and $\hat{S} = (\hat{S}(1), \ldots, \hat{S}(2^{m+1} - 1))$ to denote the set of all nodes from all private trees $B_1, \ldots, B_n$. Moreover, let $\Upsilon = (\Upsilon(1), \ldots, \Upsilon(2^{m+1} - 1))$ denote any possible value of $\hat{S}$, where $\Upsilon(r) = (\omega_1(r), \ldots, \omega_n(r))$ for any $r \in [2^{m+1} - 1]$.

Due to the post-processing immunity theorem (Dwork et al., 2006), proving the $(\epsilon, 0)$-DP property for Algorithm 1 can be transferred to proving that

$$\frac{\Pr\left[\hat{S} = \Upsilon\right]}{\Pr\left[\hat{S}' = \Upsilon\right]} = \prod_{r=1}^{2^{m+1}-1} \frac{\Pr\left[\hat{S}(r) = \Upsilon(r) \mid \hat{S}(1) = \Upsilon(1), \ldots, \hat{S}(r-1) = \Upsilon(r-1)\right]}{\Pr\left[\hat{S}'(r) = \Upsilon(r) \mid \hat{S}'(1) = \Upsilon(1), \ldots, \hat{S}'(r-1) = \Upsilon(r-1)\right]} \leq e^{\epsilon}. \tag{12}$$

Under the conditions $\hat{S}(1) = \Upsilon(1), \ldots, \hat{S}(r-1) = \Upsilon(r-1)$ and $\hat{S}'(1) = \Upsilon(1), \ldots, \hat{S}'(r-1) = \Upsilon(r-1)$, the local decisions determined by the nodes from 1 to $r - 1$ are also fixed, as well as all leaf nodes among these nodes. Furthermore, since the noise for each node of the binary tree for each local learner is independently sampled from the same Laplace

distribution $\mathcal{D}$, each factor in the above product satisfies

$$\begin{aligned}
&\frac{\Pr\left[\hat{S}(r)=\Upsilon(r)\mid \hat{S}(1)=\Upsilon(1),\dots,\hat{S}(r-1)=\Upsilon(r-1)\right]}{\Pr\left[\hat{S}'(r)=\Upsilon(r)\mid \hat{S}'(1)=\Upsilon(1),\dots,\hat{S}'(r-1)=\Upsilon(r-1)\right]}\\
=&\frac{\Pr\left[\hat{\mathbf{s}}_1(r)=\omega_1(r),\dots,\hat{\mathbf{s}}_n(r)=\omega_n(r)\mid \hat{S}(1)=\Upsilon(1),\dots,\hat{S}(r-1)=\Upsilon(r-1)\right]}{\Pr\left[\hat{\mathbf{s}}'_1(r)=\omega_1(r),\dots,\hat{\mathbf{s}}'_n(r)=\omega_n(r)\mid \hat{S}'(1)=\Upsilon(1),\dots,\hat{S}'(r-1)=\Upsilon(r-1)\right]}\\
=&\frac{\Pr\left[\mathbf{b}_1(r)=\omega_1(r)-\mathbf{s}_1(r),\dots,\mathbf{b}_n(r)=\omega_n(r)-\mathbf{s}_n(r)\mid \hat{S}(1)=\Upsilon(1),\dots,\hat{S}(r-1)=\Upsilon(r-1)\right]}{\Pr\left[\mathbf{b}'_1(r)=\omega_1(r)-\mathbf{s}'_1(r),\dots,\mathbf{b}'_n(r)=\omega_n(r)-\mathbf{s}'_n(r)\mid \hat{S}'(1)=\Upsilon(1),\dots,\hat{S}'(r-1)=\Upsilon(r-1)\right]}\\
=&\frac{\mathrm{pdf}\left[\mathbf{b}_1(r)=\omega_1(r)-\mathbf{s}_1(r),\dots,\mathbf{b}_n(r)=\omega_n(r)-\mathbf{s}_n(r)\mid \hat{S}(1)=\Upsilon(1),\dots,\hat{S}(r-1)=\Upsilon(r-1)\right]}{\mathrm{pdf}\left[\mathbf{b}'_1(r)=\omega_1(r)-\mathbf{s}'_1(r),\dots,\mathbf{b}'_n(r)=\omega_n(r)-\mathbf{s}'_n(r)\mid \hat{S}'(1)=\Upsilon(1),\dots,\hat{S}'(r-1)=\Upsilon(r-1)\right]}
\end{aligned} \tag{13}$$

where $\mathbf{b}_i(r)$ denotes the noise added in the r-th node of local learner i and $\mathbf{s}_i(r)$ denotes the original value of this node. The second equality is due to lines 5 to 7 in Algorithm 2, which implies that $\hat{\mathbf{s}}_i(r)=\mathbf{s}_i(r)+\mathbf{b}_i(r)$. Then, by substituting the probability density function of $\mathcal{D}=\mathrm{Lap}^d(0,6\sqrt{d}\epsilon^{-1}(G+\alpha R)(2+\log_2 T))$, we have

$$\begin{aligned}
&\frac{\mathrm{pdf}\left[\mathbf{b}_1(r)=\omega_1(r)-\mathbf{s}_1(r),\dots,\mathbf{b}_n(r)=\omega_n(r)-\mathbf{s}_n(r)\mid \hat{S}(1)=\Upsilon(1),\dots,\hat{S}(r-1)=\Upsilon(r-1)\right]}{\mathrm{pdf}\left[\mathbf{b}'_1(r)=\omega_1(r)-\mathbf{s}'_1(r),\dots,\mathbf{b}'_n(r)=\omega_n(r)-\mathbf{s}'_n(r)\mid \hat{S}'(1)=\Upsilon(1),\dots,\hat{S}'(r-1)=\Upsilon(r-1)\right]}\\
=&\prod_{i=1}^{n}\frac{\frac{\epsilon}{12\sqrt{d}(G+\alpha R)(2+\log_2 T)}\exp\left\{-\frac{\epsilon\|\omega_i(r)-\mathbf{s}_i(r)\|_1}{6\sqrt{d}(G+\alpha R)(2+\log_2 T)}\right\}}{\frac{\epsilon}{12\sqrt{d}(G+\alpha R)(2+\log_2 T)}\exp\left\{-\frac{\epsilon\|\omega_i(r)-\mathbf{s}'_i(r)\|_1}{6\sqrt{d}(G+\alpha R)(2+\log_2 T)}\right\}}\\
=&\prod_{i=1}^{n}\exp\left\{\frac{\epsilon\left(\|\omega_i(r)-\mathbf{s}'_i(r)\|_1-\|\omega_i(r)-\mathbf{s}_i(r)\|_1\right)}{6\sqrt{d}(G+\alpha R)(2+\log_2 T)}\right\}\le\prod_{i=1}^{n}\exp\left\{\frac{\epsilon\|\mathbf{s}'_i(r)-\mathbf{s}_i(r)\|_1}{6\sqrt{d}(G+\alpha R)(2+\log_2 T)}\right\}.
\end{aligned} \tag{14}$$

According to Algorithm 2, $\mathbf{s}_i(r)$ is the sum of the original value of all its descendant leaf nodes. Due to the definition of DP, $\mathcal{F}$ and $\mathcal{F}'$ only differ in at most one local loss function. We assume they differ at the τ-th iteration for local learner j and denote z_τ as the block that time step τ is in. According to line 14 in Algorithm 1, $\mathbf{w}_i(z-1)=\mathbf{d}_i^L(z-1)$ is the original value of leaf nodes. Let the r_τ-th node denote the leaf node that contains the noised version of $\mathbf{d}_i^L(z_\tau-1)$. Here, we consider two cases.

1. Node r_τ **is not** a descendant of node r : Since we compute the bound under the condition that all nodes 1 to $r-1$ in binary trees of all local learners are fixed, the local decisions of all local learners generated by these nodes are also the same under $\mathcal{F}$ and $\mathcal{F}'$. Since the original values in descendant leaf nodes of r depend on a subset of these decisions and the loss functions on the blocks of these leaf nodes are also the same under $\mathcal{F}$ and $\mathcal{F}'$, these original values are the same as well, which implies that
$$\|\mathbf{s}'_i(r)-\mathbf{s}_i(r)\|_1=0.$$

2. Node r_τ **is** a descendant of node r : The condition that the local decisions determined by nodes 1 to $r-1$ are the same still holds. However, since $\mathcal{F}$ and $\mathcal{F}'$ differ at iteration τ, the original value of r_τ, i.e., $\mathbf{d}_i^L(z_\tau)$, differs between $\mathcal{F}$ and $\mathcal{F}'$. Moreover, since loss functions on other time steps are the same in $\mathcal{F}$ and $\mathcal{F}'$, we have
$$\|\mathbf{s}'_i(r)-\mathbf{s}_i(r)\|=\|\mathbf{d}_i^{L\prime}(z_\tau)-\mathbf{d}_i^L(z_\tau)\|\le\|\mathbf{d}_i^{L\prime}(z_\tau)-\bar{\mathbf{d}}'(z_\tau)\|+\|\mathbf{d}_i^L(z_\tau)-\bar{\mathbf{d}}(z_\tau)\|+\|\bar{\mathbf{d}}'(z_\tau)-\bar{\mathbf{d}}(z_\tau)\|. \tag{15}$$
According to lines 9 to 10 in Algorithm 1, gradients on one block do not interfere with each other, which implies that
$$\begin{aligned}
\left\|\bar{\mathbf{d}}'(z_\tau)-\bar{\mathbf{d}}(z_\tau)\right\|&=\left\|\frac{1}{n}\sum_{i=1}^{n}\sum_{t\in\mathcal{T}_{z_\tau}}\left(\left(\nabla f'_{t,i}\left(\mathbf{x}_i(z_\tau)\right)-\alpha\mathbf{x}'_i(z_\tau)\right)-\left(\nabla f_{t,i}\left(\mathbf{x}_i(z_\tau)\right)-\alpha\mathbf{x}_i(z_\tau)\right)\right)\right\|\\
&\le\frac{1}{n}\left\|\nabla f'_{\tau,j}(\mathbf{x}_j(z_\tau))-\nabla f_{\tau,j}(\mathbf{x}_j(z_\tau))\right\|\le\frac{2G}{n}.
\end{aligned}$$

Then, by using Lemma D.3, we have

$$\left\|\mathbf{d}_i^{L\prime}(z_\tau) - \bar{\mathbf{d}}'(z_\tau)\right\| + \left\|\mathbf{d}_i^{L}(z_\tau) - \bar{\mathbf{d}}(z_\tau)\right\| \leq \frac{4L(G+\alpha R)}{nT}.$$

We naively bound the ℓ_1-norm of a vector by $\|\mathbf{x}\|_1 \leq \sqrt{d}\|\mathbf{x}\|$ where d denotes the dimensionality of $\mathbf{x}$. By substituting the above bound of $\|\bar{\mathbf{d}}'(z_\tau) - \bar{\mathbf{d}}(z_\tau)\|$ and the above inequality into (15), we finally have

$$\|\mathbf{s}_i'(r) - \mathbf{s}_i(r)\|_1 \leq \sqrt{d}\left(\frac{4L(G+\alpha R)}{nT} + \frac{2G}{n}\right) \leq \frac{6\sqrt{d}(G+\alpha R)}{n}.$$

By substituting the above bound of $\|\mathbf{s}_i'(r) - \mathbf{s}_i(r)\|_1$ into (14), we can bound (13) by

$$\frac{\Pr\left[\hat{S}(r) = \Upsilon(r) \mid \hat{S}(1) = \Upsilon(1), \ldots, \hat{S}(r-1) = \Upsilon(r-1)\right]}{\Pr\left[\hat{S}'(r) = \Upsilon(r) \mid \hat{S}'(1) = \Upsilon(1), \ldots, \hat{S}'(r-1) = \Upsilon(r-1)\right]} \leq \exp\left\{\frac{n\epsilon \cdot \frac{6\sqrt{d}(G+\alpha R)}{n}}{6\sqrt{d}(G+\alpha R)(2+\log_2 T)}\right\} = e^{\frac{\epsilon}{2+\log_2 T}}.$$

Now, we can obtain the bound for each factor in the product of (12) by substituting the above inequality into (13). Moreover, since under the condition that when considering the node r for $r \in [2^{m+1}-1]$, noised value in nodes 1 to $r-1$ are fixed, a change in the value of a leaf node affects only its ancestor nodes, which total $m+1$ nodes. Since the same position nodes in the private trees maintained by each local learner are operated in the same block, changing one factor of the product in (12) will only affect $m+1$ other factors, where $m+1$ is the number of layers in the binary tree. By substituting $m = \lceil \log_2(T/L - 1) \rceil$, we can finally prove the $(\epsilon, 0)$-DP guarantee for Algorithm 1 by

$$\frac{\Pr\left[\hat{S} = \Upsilon\right]}{\Pr\left[\hat{S}' = \Upsilon\right]} \leq \prod_{i=1}^{m+1} e^{\frac{\epsilon}{2+\log_2 T}} \leq e^{\epsilon}.$$

D. Proof of the Regret Bound in Theorem 3.6

For any $z \in [T/L]$, we define a virtual global decision as

$$\bar{\mathbf{y}}(z) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \left\langle \sum_{\tau=1}^{z-1} \bar{\mathbf{d}}(\tau), \mathbf{x} \right\rangle + \frac{\alpha(z-1)L + 2h}{2}\|\mathbf{x}\|^2 \tag{16}$$

where $\bar{\mathbf{d}}(\tau) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{d}_i(\tau)$. As in Lemma D.1, we first notice that if the local decision $\mathbf{x}_i(z)$ is close to $\bar{\mathbf{y}}(z+1)$, the regret $R_{T,i}$ will be bounded well.

Lemma D.1. *Suppose Assumptions 3.1, 3.2, 3.3, and 3.4 hold, and* $\mathbb{E}\left[\|\mathbf{x}_i(z) - \bar{\mathbf{y}}(z+1)\|\right] \leq \Xi(z)$ *for any* $i \in [n]$ *and* $z \in [T/L]$*, where* $\Xi(z)$ *is related to* z*. Then, for any* $i \in [n]$*, Algorithm 1 satisfies* $\mathbb{E}[R_{T,i}] \leq 3nGL\sum_{z=1}^{T/L}\Xi(z) + nhR^2$.

Instead of directly bounding the distance between $\mathbf{x}_i(z)$ and $\bar{\mathbf{y}}(z+1)$, we further introduce an intermediate variable, i.e., a non-private version of $\mathbf{x}_i(z)$ defined as $\mathbf{y}_i(1) = \mathbf{0}$ and

$$\mathbf{y}_i(z) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \langle \mathbf{z}_i(z-1), \mathbf{x}\rangle + \frac{\alpha(z-2)L + 2h}{2}\|\mathbf{x}\|^2$$

for $z \geq 2$, where $\mathbf{z}_i(z) = \sum_{\tau=1}^{z-1} \mathbf{d}_i^L(\tau)$ are the non-private partial sum. Then, it is easy to verify that

$$\mathbb{E}\left[\|\mathbf{x}_i(z) - \bar{\mathbf{y}}(z+1)\|\right] \leq \mathbb{E}\left[\|\mathbf{x}_i(z) - \mathbf{y}_i(z)\|\right] + \|\mathbf{y}_i(z) - \bar{\mathbf{y}}(z+1)\|. \tag{17}$$

To bound $\mathbb{E}[\|\mathbf{x}_i(z) - \mathbf{y}_i(z)\|]$, we introduce the following lemma.

Lemma D.2. *(Lemma 5 in Duchi et al. (2011)) Let* $\Pi_{\mathcal{K}}(\mathbf{u}, \eta) = \operatorname{argmin}_{\mathbf{x}\in\mathcal{K}} \langle \mathbf{u}, \mathbf{x}\rangle + \frac{\eta}{2}\|\mathbf{x}\|^2$. *We have*

$$\|\Pi_{\mathcal{K}}(\mathbf{u}, \eta) - \Pi_{\mathcal{K}}(\mathbf{v}, \eta)\| \leq \frac{1}{\eta}\|\mathbf{u} - \mathbf{v}\|.$$

Let $\mathbf{b}_i(r)$ denote the noise added for node r in the private tree of local learner i. Due to the variance of noise distribution $\mathcal{D}$, for any $i \in [n]$ and $r \in \left[2^{\lceil \log_2(T/L-1)\rceil+1} - 1\right]$, we have

$$\mathbb{E}\left[\|\mathbf{b}_i(r)\|^2\right] \leq \sum_{p=1}^{d} 2\left(\frac{6\sqrt{d}(G+\alpha R)(2+\log_2 T)}{\epsilon}\right)^2 = 72d\left(\frac{\sqrt{d}(G+\alpha R)(2+\log_2 T)}{\epsilon}\right)^2.$$

Then, let S_i denote the set of nodes selected to compute the private sum $\hat{\mathbf{z}}_i(z-1)$. According to Algorithm 2, for any $i \in [n]$, we have $|S_i| \leq \lceil \log_2(T/L-1)\rceil + 1$, because there is at most one node for each level of the private tree in the set S_i. Applying Lemma D.2 and combining the above inequality, for any $i \in [n]$ and $z \geq 2$, we have

$$\mathbb{E}\left[\|\mathbf{x}_i(z) - \mathbf{y}_i(z)\|\right] \leq \mathbb{E}\left[\frac{\|\hat{\mathbf{z}}_i(z-1) - \mathbf{z}_i(z-1)\|}{\alpha(z-2)L+2h}\right] \leq \sum_{r\in S_i} \frac{\mathbb{E}[\|\mathbf{b}_i(r)\|]}{\alpha(z-2)L+2h} \leq \frac{9d\,(G+\alpha R)\,(2+\log_2 T)^2}{\epsilon(\alpha(z-2)L+2h)}. \tag{18}$$

Next, we analyze the bound of $\|\mathbf{y}_i(z) - \bar{\mathbf{y}}(z+1)\|$, which requires an additional lemma.

Lemma D.3. *Let $\bar{\mathbf{d}}(z) = \frac{1}{n}\sum_{i=1}^{n} \mathbf{d}_i(z)$. Under Assumptions 3.1, 3.2, 3.3, and 3.4, for any $i \in [n]$ and $z \in \lceil T/L \rceil$, when setting parameters as in (7), Algorithm 1 ensures that*

$$\left\|\mathbf{d}_i^L(z) - \bar{\mathbf{d}}(z)\right\| \leq \frac{2L(G+\alpha R)}{nT}. \tag{19}$$

Applying Lemmas D.2 and D.3, for any $z \geq 2$, we have

$$\|\mathbf{y}_i(z) - \bar{\mathbf{y}}(z-1)\| \leq \frac{1}{\alpha(z-2)L+2h}\sum_{\tau=1}^{z-2}\left\|\mathbf{d}_i^L(\tau) - \bar{\mathbf{d}}(\tau)\right\| \leq \frac{2(z-2)L(G+\alpha R)}{nT(\alpha(z-2)L+2h)} \leq \frac{2(G+\alpha R)}{n(\alpha(z-2)L+2h)}. \tag{20}$$

Now, we still need to analyze $\|\bar{\mathbf{y}}(z-1) - \bar{\mathbf{y}}(z+1)\|$. To this end, we define $\ell_z(\mathbf{x}) = \left\langle \bar{\mathbf{d}}(z), \mathbf{x}\right\rangle + \frac{\alpha L}{2}\|\mathbf{x}\|^2$ and $J_z(\mathbf{x}) = \sum_{\tau=1}^{z} \ell_\tau(\mathbf{x}) + h\|\mathbf{x}\|^2$ for any $z \in [T/L]$. It is easy to verify that the function $J_z(\mathbf{x})$ is $(\alpha z L + 2h)$-strongly convex and $\bar{\mathbf{y}}(z+1) = \operatorname{argmin}_{\mathbf{x}\in\mathcal{K}} J_z(\mathbf{x})$.

Moreover, as proven by Hazan & Kale (2012), for any λ-strongly convex functions $f(\mathbf{x})$ over the set $\mathcal{K}$ with $\mathbf{x}^* = \operatorname{argmin}_{\mathbf{x}\in\mathcal{K}} f(\mathbf{x})$, it holds that

$$\frac{\lambda}{2}\|\mathbf{x} - \mathbf{x}^*\|^2 \leq f(\mathbf{x}) - f(\mathbf{x}^*),\ \forall \mathbf{x} \in \mathcal{K}. \tag{21}$$

Combining (21) with the strong convexity of $J_z(\mathbf{x})$, for any $1 \leq z' \leq z$, we have

$$\begin{aligned}
\|\bar{\mathbf{y}}(z') - \bar{\mathbf{y}}(z+1)\|^2 \leq& \frac{2}{\alpha z L + 2h}\left(J_z(\bar{\mathbf{y}}(z')) - J_z(\bar{\mathbf{y}}(z+1))\right) \\
=& \frac{2}{\alpha z L + 2h}\left(J_{z'-1}(\bar{\mathbf{y}}(z')) - J_{z'-1}(\bar{\mathbf{y}}(z+1))\right) + \frac{2}{\alpha z L + 2h}\sum_{\tau=z'-1}^{z}\left(\ell_\tau(\bar{\mathbf{y}}(z')) - \ell_\tau(\bar{\mathbf{y}}(z+1))\right) \\
\leq& \frac{2}{\alpha z L + 2h}\sum_{\tau=z'-1}^{z}\left(\ell_\tau(\bar{\mathbf{y}}(z')) - \ell_\tau(\bar{\mathbf{y}}(z+1))\right).
\end{aligned}$$

Additionally, for any $z \geq 1$, it is easy to verify that

$$\begin{aligned}
|\ell_z(\mathbf{x}) - \ell_z(\mathbf{y})| \leq& |\left\langle \nabla\ell_z(\mathbf{x}), \mathbf{x} - \mathbf{y}\right\rangle| \leq \|\nabla\ell_z(\mathbf{x})\|\,\|\mathbf{x} - \mathbf{y}\| \leq \left\|\bar{\mathbf{d}}(z-1) + \alpha L\mathbf{x}\right\|\|\mathbf{x} - \mathbf{y}\| \\
\leq& \left(\frac{1}{n}\sum_{i=1}^{n}\|\mathbf{d}_i(z-1)\| + \alpha L\|\mathbf{x}\|\right)\|\mathbf{x} - \mathbf{y}\| \leq L(G + 2\alpha R)\,\|\mathbf{x} - \mathbf{y}\|
\end{aligned}$$

where the last inequality is due to the following inequality derived under Assumptions 3.1 and 3.2

$$\max_{i\in[n]}\|\mathbf{d}_i(z)\| = \max_{i\in[n]}\left\|\sum_{t\in\mathcal{T}_z}\left(\nabla f_{t,i}\left(\mathbf{x}_i(z)\right) - \alpha\mathbf{x}_i(z)\right)\right\| \leq \max_{i\in[n]}\sum_{t\in\mathcal{T}_z}\left(\|\nabla f_{t,i}\left(\mathbf{x}_i(z)\right)\| + \alpha\left\|\mathbf{x}_i(z)\right\|\right) \leq L(G+\alpha R). \tag{22}$$

Then, for any $1 \leq z' \leq z$, we have

$$\|\bar{\mathbf{y}}(z') - \bar{\mathbf{y}}(z+1)\| \leq \frac{2}{\alpha z L + 2h} L(z - z' + 1)(G + 2\alpha R). \tag{23}$$

Combining (18), (20) and the above inequality, for $z \geq 2$, we have

$$\mathbb{E}\left[\|\mathbf{x}_i(z) - \mathbf{y}_i(z)\| + \|\mathbf{y}_i(z) - \bar{\mathbf{y}}(z+1)\|\right] \leq \frac{2(G + \alpha R)}{n(\alpha(z-2)L + 2h)} + \frac{9d\,(G + \alpha R)\,(2 + \log_2 T)^2}{\epsilon\,(\alpha(z-2)L + 2h)} + \frac{4L(G + 2\alpha R)}{\alpha z L + 2h}.$$

For $z = 1$, due to $\mathbf{x}_i(1) = \mathbf{y}_i(1) = \bar{\mathbf{y}}(1) = \mathbf{0}$, we have

$$\mathbb{E}\left[\|\mathbf{x}_i(1) - \mathbf{y}_i(1)\| + \|\mathbf{y}_i(1) - \bar{\mathbf{y}}(2)\|\right] \leq \frac{2L(G + 2\alpha R)}{\alpha L + 2h}.$$

Finally, we can derive the regret bound in Theorem 3.6 by first substituting the above two inequalities into (17) and then using Lemma D.1.

E. Proof of Lemma D.1

According to Assumptions 3.2 and 3.3, for any $\mathbf{x} \in \mathcal{K}$, it is not hard to verify that

$$\begin{aligned}
&f_{t,j}(\mathbf{x}_i(z)) - f_{t,j}(\mathbf{x}) \\
\leq& f_{t,j}(\mathbf{x}_j(z)) - f_{t,j}(\mathbf{x}) + G\|\mathbf{x}_j(z) - \mathbf{x}_i(z)\| \\
\leq& \langle \nabla f_{t,j}(\mathbf{x}_j(z)), \mathbf{x}_j(z) - \mathbf{x}\rangle - \frac{\alpha}{2}\|\mathbf{x}_j(z) - \mathbf{x}\|^2 + G\|\mathbf{x}_j(z) - \mathbf{x}_i(z)\| \\
\leq& \langle \nabla f_{t,j}(\mathbf{x}_j(z)), \bar{\mathbf{y}}(z+1) - \mathbf{x}\rangle - \frac{\alpha}{2}\|\mathbf{x}_j(z) - \mathbf{x}\|^2 + \langle \nabla f_{t,j}(\mathbf{x}_j(z)), \mathbf{x}_j(z) - \bar{\mathbf{y}}(z+1)\rangle \\
&+ G\|\mathbf{x}_j(z) - \bar{\mathbf{y}}(z+1) + \bar{\mathbf{y}}(z+1) - \mathbf{x}_i(z)\| \\
\leq& \langle \nabla f_{t,j}(\mathbf{x}_j(z)), \bar{\mathbf{y}}(z+1) - \mathbf{x}\rangle - \frac{\alpha}{2}\|\mathbf{x}_j(z) - \mathbf{x}\|^2 + 2G\|\mathbf{x}_j(z) - \bar{\mathbf{y}}(z+1)\| + G\|\mathbf{x}_i(z) - \bar{\mathbf{y}}(z+1)\|.
\end{aligned} \tag{24}$$

Recall that we suppose for any $i \in [n]$ and $z \in [T/L]$, there exists a $\Xi(z)$ such that $\mathbb{E}\left[\|\mathbf{x}_i(z) - \bar{\mathbf{y}}(z+1)\|\right] \leq \Xi(z)$. Then, we have

$$\mathbb{E}\left[f_{t,j}(\mathbf{x}_i(z)) - f_{t,j}(\mathbf{x})\right] \leq \langle \nabla f_{t,j}(\mathbf{x}_j(z)), \bar{\mathbf{y}}(z+1) - \mathbf{x}\rangle - \frac{\alpha}{2}\|\mathbf{x}_j(z) - \mathbf{x}\|^2 + 3G\Xi(z).$$

Furthermore, we notice that

$$\begin{aligned}
\|\mathbf{x}_j(z) - \mathbf{x}\|^2 &= \|\mathbf{x}_j(z) - \bar{\mathbf{y}}(z+1)\|^2 + 2\langle \mathbf{x}_j(z) - \bar{\mathbf{y}}(z+1), \bar{\mathbf{y}}(z+1) - \mathbf{x}\rangle + \|\bar{\mathbf{y}}(z+1) - \mathbf{x}\|^2 \\
&= \|\mathbf{x}_j(z) - \bar{\mathbf{y}}(z+1)\|^2 + 2\langle \mathbf{x}_j(z), \bar{\mathbf{y}}(z+1) - \mathbf{x}\rangle + \|\mathbf{x}\|^2 - \|\bar{\mathbf{y}}(z+1)\|^2 \\
&\geq 2\langle \mathbf{x}_j(z), \bar{\mathbf{y}}(z+1) - \mathbf{x}\rangle + \|\mathbf{x}\|^2 - \|\bar{\mathbf{y}}(z+1)\|^2.
\end{aligned} \tag{25}$$

Combining the above two inequalities, for any $z \geq 1$ and $\mathbf{x} \in \mathcal{K}$, we have

$$\mathbb{E}\left[f_{t,j}(\mathbf{x}_i(z)) - f_{t,j}(\mathbf{x})\right] \leq \langle \nabla f_{t,j}(\mathbf{x}_j(z)) - \alpha\mathbf{x}_j(z), \bar{\mathbf{y}}(z+1) - \mathbf{x}\rangle - \frac{\alpha}{2}\left(\|\mathbf{x}\|^2 - \|\bar{\mathbf{y}}(z+1)\|^2\right) + 3G\Xi(z).$$

By summing up the above inequality overall T iterations and n local learners, for any $\mathbf{x} \in \mathcal{K}$, we have

$$\begin{aligned}
&\mathbb{E}[R_{T,i}] = \mathbb{E}\left[\sum_{z=1}^{T/L}\sum_{t\in\mathcal{T}_z}\sum_{j=1}^{n}\left(f_{t,j}(\mathbf{x}_i(z)) - f_{t,j}(\mathbf{x})\right)\right] \\
\leq& \sum_{z=1}^{T/L}\sum_{t\in\mathcal{T}_z}\sum_{j=1}^{n}\left(\langle \nabla f_{t,j}(\mathbf{x}_j(z)) - \alpha\mathbf{x}_j(z), \bar{\mathbf{y}}(z+1) - \mathbf{x}\rangle - \frac{\alpha}{2}\left(\|\mathbf{x}\|^2 - \|\bar{\mathbf{y}}(z+1)\|^2\right)\right) + \sum_{z=1}^{T/L}\sum_{t\in\mathcal{T}_z}\sum_{j=1}^{n} 3G\Xi(z) \\
\leq& n\sum_{z=1}^{T/L}\left(\langle \bar{\mathbf{d}}(z), \bar{\mathbf{y}}(z+1) - \mathbf{x}\rangle + \frac{\alpha L}{2}\left(\|\bar{\mathbf{y}}(z+1)\|^2 - \|\mathbf{x}\|^2\right)\right) + 3nLG\sum_{z=1}^{T/L}\Xi(z).
\end{aligned} \tag{26}$$

To bound the first term in the right side of (26), we introduce the following lemma.

Lemma E.1. *(Lemma 6.6 in Garber & Hazan (2016)) Let $\{f_t(\mathbf{x})\}_{t=1}^T$ be a sequence of functions and $\mathbf{x}_t^* \in \operatorname{argmin}_{\mathbf{x}\in\mathcal{K}} \sum_{\tau=1}^t f_\tau(\mathbf{x})$ for all $t \in [T]$. Then, it holds that*

$$\sum_{t=1}^T f_t(\mathbf{x}_t^*) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^T f_t(\mathbf{x}) \leq 0.$$

Let $\ell_0(\mathbf{x}) = h\|\mathbf{x}\|^2$ and $\ell_z(\mathbf{x}) = \left\langle \bar{\mathbf{d}}(z), \mathbf{x} \right\rangle + \frac{\alpha L}{2}\|\mathbf{x}\|^2$ for any $z \in [T/L]$. Then, for any $\mathbf{x} \in \mathcal{K}$, it is easy to verify that

$$\begin{aligned} &\sum_{z=1}^{T/L} \left(\left\langle \bar{\mathbf{d}}(z), \bar{\mathbf{y}}(z+1) - \mathbf{x} \right\rangle + \frac{\alpha L}{2} \left(\|\bar{\mathbf{y}}(z+1)\|^2 - \|\mathbf{x}\|^2 \right) \right) \\ =& \sum_{z=1}^{T/L} \left(\ell_z(\bar{\mathbf{y}}(z+1) - \ell_z(\mathbf{x}) \right) \leq h\|\mathbf{x}\|^2 - h\|\bar{\mathbf{y}}(z+1)\|^2 \leq hR^2 \end{aligned} \tag{27}$$

where the first inequality is due to Lemma E.1 and $\bar{\mathbf{y}}(z+1) = \operatorname{argmin}_{\mathbf{x}\in\mathcal{K}} \sum_{\tau=0}^z \ell_\tau(\mathbf{x})$. Finally, we complete this proof by substituting (27) into (26).

F. Proof of Lemma D.3

Let $H^k = [\mathbf{d}_1^k(z)^\top; \mathbf{d}_2^k(z)^\top; \ldots; \mathbf{d}_n^k(z)^\top] \in \mathbb{R}^{n\times d}$ for $k = 0, \ldots, L$ and $H^{-1} = H^0$. According to (5), it is easy to verify that

$$H^{k+1} = (1+\theta)PH^k - \theta H^{k-1} \text{ for } k = 0, \ldots, L-1. \tag{28}$$

Moreover, according to the convergence property of the accelerated gossip strategy, we have the following lemma.

Lemma F.1. *(Proposition 1 in Ye et al. (2023)) Under Assumption 3.4, the iterations of (28) with $\theta = (1+\sqrt{1-\sigma_2^2(P)})^{-1}$ ensure that*

$$\left\|H^L - \bar{H}\right\|_F \leq \sqrt{14}\left(1 - \left(1 - \frac{1}{\sqrt{2}}\right)\sqrt{1-\sigma_2(P)}\right)^L \left\|H^0 - \bar{H}\right\|_F$$

where $\bar{H} = \frac{1}{n}\mathbf{1}\mathbf{1}^\top H^0$ and $\|\cdot\|_F$ denotes the Frobenius norm.

Note that $\frac{1}{n}\mathbf{1}\mathbf{1}^\top H^0 = [\bar{\mathbf{d}}(z)^\top; \bar{\mathbf{d}}(z)^\top; \ldots; \bar{\mathbf{d}}(z)^\top]$. From Lemma F.1, we have

$$\begin{aligned} \left\|\mathbf{d}_i^L(z) - \bar{\mathbf{d}}(z)\right\| &\leq \|H^L - \frac{1}{n}\mathbf{1}\mathbf{1}^\top H^0\|_F \leq \sqrt{14}\left(1 - \left(1 - \frac{1}{\sqrt{2}}\right)\sqrt{1-\sigma_2(P)}\right)^L \left\|H^0 - \frac{1}{n}\mathbf{1}\mathbf{1}^\top H^0\right\|_F \\ &\leq 2\sqrt{14}\left(1 - \left(1 - \frac{1}{\sqrt{2}}\right)\sqrt{1-\sigma_2(P)}\right)^L \|H^0\|_F \\ &= 2\sqrt{14}\left(1 - \left(1 - \frac{1}{\sqrt{2}}\right)\sqrt{1-\sigma_2(P)}\right)^L \sqrt{\sum_{i=1}^n \|\mathbf{d}_i^0(z)\|^2}. \end{aligned}$$

Let $c = 1 - \frac{1}{\sqrt{2}}$. Due to $L = \lceil 4\ln(nT\sqrt{14n})/\sqrt{\rho} \rceil$ and $c^{-1} < 4$, it is not hard to verify that

$$\begin{aligned} \left(1 - c\sqrt{1-\sigma_2(P)}\right)^L &\leq \left(1 - c\sqrt{1-\sigma_2(P)}\right)^{\frac{4\ln(nT\sqrt{14n})}{\sqrt{1-\sigma_2(P)}}} \leq \left(1 - c\sqrt{1-\sigma_2(P)}\right)^{\frac{\ln(nT\sqrt{14n})}{c\sqrt{1-\sigma_2(P)}}} \\ &\leq \left(1 - c\sqrt{1-\sigma_2(P)}\right)^{\frac{\ln(nT\sqrt{14n})}{\ln\left(1-c\sqrt{1-\sigma_2(P)}\right)^{-1}}} = \frac{1}{nT\sqrt{14n}} \end{aligned}$$

where the first inequality is due to $1 - c\sqrt{1-\sigma_2(P)} < 1$ and the second inequality is due to $\ln x^{-1} \geq 1 - x$ for any $x > 0$. Combining the above two inequalities, we can finally derive the lemma as

$$\left\|\mathbf{d}_i^L(z) - \bar{\mathbf{d}}(z)\right\| \leq \frac{2\sqrt{14\sum_{i=1}^n \|\mathbf{d}_i(z)\|^2}}{nT\sqrt{14n}} \leq \frac{2\max_{i\in[n]}\|\mathbf{d}_i(z)\|}{nT} \leq \frac{2L(G+\alpha R)}{nT}$$

where the last inequality is due to (22).

G. Proof of Corollaries 3.7 and 3.8

For Corollary 3.7, by substituting $\alpha > 0$ and $h = G\sqrt{14LT(2+\log_2 T)}/R$ into (8), we have

$$\begin{aligned}\mathbb{E}[R_{T,i}] &\leq 4nGR\sqrt{LT(2+\log_2 T)} + nL\sum_{z=1}^{T/L}\frac{2GR\sqrt{L}}{\sqrt{T(2+\log_2 T)}} \\ &\quad + nL\sum_{z=2}^{T/L}\left(\frac{4dGR(2+\log_2 T)^{3/2}}{\epsilon\sqrt{TL}} + \frac{GR}{n\sqrt{LT(2+\log_2 T)}}\right) \\ &\leq 7nGR\sqrt{LT(2+\log_2 T)} + \frac{4ndGR(2+\log_2 T)^{3/2}\sqrt{T}}{\epsilon\sqrt{L}}.\end{aligned}$$

Similarly, for Corollary 3.8, we substitute $\alpha > 0$ and $h = \alpha L$ into (8) and obtain that

$$\begin{aligned}\mathbb{E}[R_{T,i}] &\leq n\alpha LR^2 + nL\sum_{z=1}^{T/L}\frac{12G(G+2\alpha R)}{\alpha(z+2)} + nL\sum_{z=2}^{T/L}\left(\frac{27dG\,(G+\alpha R)\,(2+\log_2 T)^2}{\alpha\epsilon zL} + \frac{6G(G+\alpha R)}{n\alpha zL}\right) \\ &\leq n\alpha LR^2 + \frac{12nGL(G+2\alpha R)\,(1+\ln(T/L))}{\alpha} \\ &\quad + \frac{27ndG\,(G+\alpha R)\,(2+\log_2 T)^2\,(1+\ln(T/L))}{\alpha\epsilon} + \frac{6nG(G+\alpha R)\,(1+\ln(T/L))}{n\alpha} \\ &\leq \frac{24nGL(G+2\alpha R)\,(1+\ln(T/L))}{\alpha} + \frac{27nd(G+2\alpha R)\,(2+\log_2 T)^3}{\alpha\epsilon} + n\alpha LR^2.\end{aligned}$$

H. Proof of Theorem 3.9

Since the operation of CG does not introduce any additional sensitive data, the $(\epsilon, 0)$-DP guarantee of Theorem 3.9 can be proved in the same way as that of Theorem 3.6 in the Appendix C. Therefore, here we only prove the regret guarantee in Theorem 3.9. For $z \geq 3$, we define

$$\mathbf{x}_i^*(z) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} F_{z-1,i}(\mathbf{x}) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \langle \hat{\mathbf{z}}_i(z-2), \mathbf{x}\rangle + \frac{\alpha(z-3)L+2h}{2}\|\mathbf{x}\|^2.$$

For $z = 1$ and $z = 2$, we set $\mathbf{x}_i^*(z) = \mathbf{0}$. Thus, when $z = 1$ and $z = 2$, we have $\|\mathbf{x}_i(z) - \mathbf{x}_i^*(z)\| = 0$. For $z \geq 3$, since it is easy to verify that $F_{z-1,i}(\mathbf{x})$ is $(\alpha(z-3)L+2h)$-strongly convex, we have

$$\|\mathbf{x}_i(z) - \mathbf{x}_i^*(z)\|^2 \leq \frac{2}{\alpha(z-3)L+2h}\left(F_{z-1}(\mathbf{x}_i(z)) - F_{z-1}(\mathbf{x}_i^*(z))\right).$$

To bound the approximation error of linear optimization steps in CG, we introduce the definition of smooth functions and a lemma about the convergence rate of CG.

Definition H.1. A function $f(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$ is called β-smooth over $\mathcal{K}$, if it holds that $f(\mathbf{y}) \leq f(\mathbf{x}) + \langle \nabla f(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle + \frac{\beta}{2}\|\mathbf{x} - \mathbf{y}\|^2$ for any $\mathbf{x}, \mathbf{y} \in \mathcal{K}$.

Lemma H.2. (Derived from Theorem 1 in Jaggi (2013)) *If function $F(\mathbf{x}) : \mathcal{K} \to \mathbb{R}$ is convex and β-smooth, and $\|\mathbf{x}\| \leq R$ holds for any $\mathbf{x} \in \mathcal{K}$, Algorithm 4 ensures $F(\mathbf{x}_L) - \min_{\mathbf{x}\in\mathcal{K}} F(\mathbf{x}) \leq 8\beta R^2/(L+2)$.*

By utilizing Lemma H.2 and the fact that $F_{z-1}(\mathbf{x})$ is $(\alpha(z-3)L+2h)$-smooth, we have

$$\|\mathbf{x}_i(z) - \mathbf{x}_i^*(z)\| \leq \sqrt{\frac{2\left(F_{z-1}(\mathbf{x}_i(z)) - F_{z-1}(\mathbf{x}_i^*(z))\right)}{\alpha(z-3)L+2h}} \leq \frac{4R}{\sqrt{L}}. \tag{29}$$

Then, similar to the analysis in Appendix D, we set $\mathbf{z}_i(1) = \mathbf{0}$ and $\mathbf{z}_i(z) = \sum_{\tau=1}^{z-1}\mathbf{d}_i^{L'}(\tau)$ for $z \geq 2$. Moreover, we define $\mathbf{y}_i(1) = \mathbf{y}_i(2) = \mathbf{0}$ and for $z \geq 3$,

$$\mathbf{y}_i(z) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \langle \mathbf{z}_i(z-2), \mathbf{x}\rangle + \frac{\alpha(z-3)L+2h}{2}\|\mathbf{x}\|^2.$$

Therefore, for $z = 1$ and $z = 2$, it holds that $\|\mathbf{x}_i^*(z) - \mathbf{y}_i(z)\| = 0$. For $z \geq 3$, similar to (18) in Appendix D, it is easy to verify that

$$\mathbb{E}\left[\|\mathbf{x}_i^*(z) - \mathbf{y}_i(z)\|\right] \leq \frac{9d\left(G + \alpha R\right)\left(2 + \log_2 T\right)^2}{\epsilon\left(\alpha(z-3)L + 2h\right)}. \tag{30}$$

Next, following the analysis in Appendix D, we reuse the definition of $\bar{\mathbf{y}}(z)$ in (16). Note that the value of L' in (10) is the same as the value of L in (7). Therefore, according to the proof of Lemma D.3, it is easy to verify that $\left\|\mathbf{d}_i^{L'}(\tau) - \bar{\mathbf{d}}(\tau)\right\|$ for Algorithm 3 also enjoys the same upper bound as in (19) of Lemma D.3. By further combining with Lemma D.2, for $z \geq 3$, we have

$$\|\mathbf{y}_i(z) - \bar{\mathbf{y}}(z-2)\| \leq \frac{1}{\alpha(z-3)L + 2h} \sum_{\tau=1}^{z-3} \left\|\mathbf{d}_i^{L'}(\tau) - \bar{\mathbf{d}}(\tau)\right\| \leq \frac{2(z-3)L(G + \alpha R)}{nT\left(\alpha(z-3)L + 2h\right)} \leq \frac{2(G + \alpha R)}{n\left(\alpha(z-3)L + 2h\right)}. \tag{31}$$

Note that (23) in Appendix D also holds for Algorithm 3. Combining (31) and (23), for $z \geq 3$, we have

$$\|\mathbf{y}_i(z) - \bar{\mathbf{y}}(z+1)\| \leq \|\mathbf{y}_i(z) - \bar{\mathbf{y}}(z-2)\| + \|\bar{\mathbf{y}}(z-2) - \bar{\mathbf{y}}(z+1)\| \leq \frac{2(G + \alpha R)}{n\left(\alpha(z-3)L + 2h\right)} + \frac{6L(G + 2\alpha R)}{\alpha z L + 2h}. \tag{32}$$

For $z = 1$ and $z = 2$, since $\mathbf{y}_i(1) = \mathbf{y}_i(2) = \bar{\mathbf{y}}(1) = \mathbf{0}$, we have

$$\|\mathbf{y}_i(1) - \bar{\mathbf{y}}(2)\| = \|\bar{\mathbf{y}}(1) - \bar{\mathbf{y}}(2)\| \leq \frac{2L(G + 2\alpha R)}{\alpha L + 2h}, \ \ \|\mathbf{y}_i(2) - \bar{\mathbf{y}}(3)\| = \|\bar{\mathbf{y}}(1) - \bar{\mathbf{y}}(3)\| \leq \frac{4L(G + 2\alpha R)}{\alpha L + 2h} \tag{33}$$

where the two inequalities are due to (23).

Combining (24), (25), (29), (30), and (32), for any $z \geq 3$ and $\mathbf{x} \in \mathcal{K}$, we have

$$\begin{aligned}
&\mathbb{E}\left[f_{t,j}(\mathbf{x}_i(z)) - f_{t,j}(\mathbf{x})\right] \\
\leq& \left\langle \nabla f_{t,j}(\mathbf{x}_j(z)) - \alpha \mathbf{x}_j(z), \bar{\mathbf{y}}(z+1) - \mathbf{x} \right\rangle - \frac{\alpha}{2}\left(\|\mathbf{x}\|^2 - \|\bar{\mathbf{y}}(z+1)\|^2\right) \\
&+ 2G\mathbb{E}\left[\left\|\mathbf{x}_j(z) - \mathbf{x}_j^*(z)\right\|\right] + G\mathbb{E}\left[\|\mathbf{x}_i(z) - \mathbf{x}_i^*(z)\|\right] + 2G\left\|\mathbf{x}_j^*(z) - \mathbf{y}_j(z)\right\| + G\left\|\mathbf{x}_i^*(z) - \mathbf{y}_i(z)\right\| \\
&+ 2G\left\|\mathbf{y}_j(z) - \bar{\mathbf{y}}(z+1)\right\| + G\left\|\mathbf{y}_i(z) - \bar{\mathbf{y}}(z+1)\right\| \\
\leq& \left\langle \nabla f_{t,j}(\mathbf{x}_j(z)) - \alpha \mathbf{x}_j(z), \bar{\mathbf{y}}(z+1) - \mathbf{x} \right\rangle - \frac{\alpha}{2}\left(\|\mathbf{x}\|^2 - \|\bar{\mathbf{y}}(z+1)\|^2\right) \\
&+ \frac{12GR}{\sqrt{L}} + \frac{27dG\left(G + \alpha R\right)\left(\log_2 T + 2\right)\log_2 T}{\epsilon\left(\alpha(z-3)L + 2h\right)} + \frac{6G(G + \alpha R)}{n\left(\alpha(z-3)L + 2h\right)} + \frac{18GL(G + 2\alpha R)}{\alpha z L + 2h}.
\end{aligned}$$

For $z = 1$ and $z = 2$, similar to the above inequality, we can utilize (33) and $\mathbf{x}_i(z) = \mathbf{x}_i^*(z) = \mathbf{y}_i(z) = \mathbf{0}$ to show that

$$f_{t,j}(\mathbf{x}_i(z)) - f_{t,j}(\mathbf{x}) \leq \left\langle \nabla f_{t,j}(\mathbf{x}_j(z)) - \alpha \mathbf{x}_j(z), \bar{\mathbf{y}}(z+1) - \mathbf{x} \right\rangle - \frac{\alpha}{2}\left(\|\mathbf{x}\|^2 - \|\bar{\mathbf{y}}(z+1)\|^2\right) + \frac{12GL(G + 2\alpha R)}{\alpha L + 2h}.$$

Then, by summing up overall T iterations and n local learners, we have

$$\begin{aligned}
&\mathbb{E}\left[\sum_{z=1}^{T/L} \sum_{t \in \mathcal{T}_z} \sum_{j=1}^{n} \left(f_{t,j}(\mathbf{x}_i(z)) - f_{t,j}(\mathbf{x})\right)\right] \\
\leq& \sum_{z=1}^{T/L} \sum_{t \in \mathcal{T}_z} \sum_{j=1}^{n} \left(\left\langle \nabla f_{t,j}(\mathbf{x}_j(z)) - \alpha \mathbf{x}_j(z), \bar{\mathbf{y}}(z+1) - \mathbf{x} \right\rangle - \frac{\alpha}{2}\left(\|\mathbf{x}\|^2 - \|\bar{\mathbf{y}}(z+1)\|^2\right)\right) + \frac{24nGL^2(G + 2\alpha R)}{\alpha L + 2h} \\
&+ nL \sum_{z=3}^{T/L} \left(\frac{12GR}{\sqrt{L}} + \frac{27dG\left(G + \alpha R\right)\left(2 + \log_2 T\right)^2}{\epsilon\left(\alpha(z-3)L + 2h\right)} + \frac{6G(G + \alpha R)}{n\left(\alpha(z-3)L + 2h\right)} + \frac{18GL(G + 2\alpha R)}{\alpha z L + 2h}\right) \\
\leq& n \sum_{z=1}^{T/L} \left(\left\langle \bar{\mathbf{d}}(z), \bar{\mathbf{y}}(z+1) - \mathbf{x} \right\rangle + \frac{\alpha L}{2}\left(\|\bar{\mathbf{y}}(z+1)\|^2 - \|\mathbf{x}\|^2\right)\right) + nL \sum_{z=1}^{T/L} \frac{24GL(G + 2\alpha R)}{\alpha z L + 2h} \\
&+ nL \sum_{z=3}^{T/L} \left(\frac{12GR}{\sqrt{L}} + \frac{27dG\left(G + \alpha R\right)\left(2 + \log_2 T\right)^2}{\epsilon\left(\alpha(z-3)L + 2h\right)} + \frac{6G(G + \alpha R)}{n\left(\alpha(z-3)L + 2h\right)}\right).
\end{aligned} \tag{34}$$

Note that (27) in the proof of Lemma D.1 still holds here. It is easy to complete this proof by substituting (27) into (34).

I. Proof of Corollaries 3.10 and 3.11

For Corollary 3.10, by substituting $\alpha = 0, h = \sqrt{15LT}G/R$ and $L = \sqrt{T}$ into (11), we have

$$\begin{aligned}\mathbb{E}[R_{T,i}] &\leq 4nGRT^{3/4} + \sum_{t=1}^{T} \frac{4nGR}{T^{1/4}} + n\sum_{t=1}^{T}\left(\frac{12GR}{T^{1/4}} + \frac{4dGR(2+\log_2 T)^2}{\epsilon T^{3/4}} + \frac{GR}{nT^{3/4}}\right). \\ &\leq 21nGRT^{3/4} + \frac{4ndGRT^{1/4}\left(2+\log_2 T\right)^2}{\epsilon}.\end{aligned}$$

For Corollary 3.11, we set $\alpha > 0, h = \alpha L$ and $L = T^{2/3}(\ln^{-2/3} T)$ into (11), and obtain that

$$\begin{aligned}\mathbb{E}[R_{T,i}] &\leq n\alpha R^2 T^{2/3}\ln^{-2/3} T + n\sum_{z=1}^{T/L}\frac{24G(G+2\alpha R)T^{2/3}\ln^{-2/3} T}{\alpha(z+1)} + 12nGRT^{2/3}\ln^{2/3} T \\ &\quad + n\sum_{z=3}^{T/L}\left(\frac{27dG\left(G+\alpha R\right)\left(2+\log_2 T\right)\log_2 T}{\alpha\epsilon(z-1)} + \frac{6G(G+\alpha R)}{n\alpha(z-1)}\right) \\ &\leq n\alpha R^2 T^{2/3}\ln^{-2/3} T + \frac{24nGL(G+2\alpha R)T^{2/3}\left(\ln^{-2/3} T + \ln^{1/3} T\right)}{\alpha} + 12nGRT^{2/3}\ln^{2/3} T \\ &\quad + \left(\frac{27dG\left(G+\alpha R\right)\left(2+\log_2 T\right)\left(1+\ln(T/L)\right)\log_2 T}{\alpha\epsilon} + \frac{6G(G+\alpha R)\left(1+\ln(T/L)\right)}{n\alpha}\right) \\ &\leq \frac{36nGT^{2/3}\left(\ln^{-2/3} T + \ln^{1/3} T\right)(G+2\alpha R)}{\alpha} + n\alpha R^2 T^{2/3}\ln^{-2/3} T \\ &\quad + 12nGRT^{2/3}\ln^{2/3} T + \frac{27ndG(G+\alpha R)\left(2+\log_2 T\right)^3}{\alpha\epsilon}.\end{aligned}$$