Revisiting Decentralized Online Convex Optimization with Compressed Communication

Hao Zhou[1] **Xiaoyu Wang**[1] **Chang Yao**[1,2] **Mingli Song**[1,2] **Yuanyu Wan**[1,2*]

[1] School of Software Technology, Zhejiang University, Ningbo, China

[2] State Key Laboratory of Blockchain and Data Security, Zhejiang University, Hangzhou, China

{juhao, xiaoyuw, changy, brooksong, wanyy}@zju.edu.cn

Abstract

Decentralized online convex optimization (D-OCO) is a popular framework for distributed applications with streaming data. To tackle the communication bottleneck, previous studies have investigated D-OCO with compressed communication and proposed several algorithms that are variants of online gradient descent (OGD). However, for D-OCO with exact communication, the best existing algorithms are variants of follow-the-regularized-leader (FTRL). In this paper, for the first time, we propose two FTRL-type algorithms for D-OCO with compressed communication. Compared with OGD-type algorithms, our algorithms are more elegant in both algorithmic design and theoretical analysis. The key insight is that the dual update mechanism of FTRL allows us to make a simple application of the technique for average consensus with communication compression. More specifically, our first algorithm considers the full-information setting, and can match the existing regret bounds. Our second algorithm is designed for the bandit setting, and can significantly improve both the regret bounds and communication costs of existing algorithms.

1 Introduction

This paper investigates the decentralized online convex optimization (D-OCO) problem with a particular focus on algorithms that only use compressed communication. In general, D-OCO is formulated as a repeated game between n learners in a network and an adversary. At each round t, each learner $i \in [n]$ selects a decision $\mathbf{x}_i(t)$ from a convex set $\mathcal{K} \subseteq \mathbb{R}^d$, and then receives a convex loss function $f_{t,i}(\cdot) : \mathcal{K} \mapsto \mathbb{R}$ chosen by the adversary. Let $f_t(\cdot) = \sum_{j=1}^{n} f_{t,j}(\cdot)$ denote the global function of round t. For total T rounds, the goal of each learner $i \in [n]$ is to minimize the regret in terms of the global function, i.e., $R_{T,i} = \sum_{t=1}^{T} f_t(\mathbf{x}_i(t)) - \min_{\mathbf{x}\in\mathcal{K}} \sum_{t=1}^{T} f_t(\mathbf{x})$. To this end, these learners are allowed to communicate with their immediate neighbors once per round.

In the case with exact communication, D-OCO has been extensively studied over the past decade [Yan et al., 2013, Hosseini et al., 2013, Zhang et al., 2017, Wan et al., 2020, 2022, Wang et al., 2023, Wan et al., 2024a, 2025, Wan, 2025, Wang et al., 2025, Qiu et al., 2026]. It is well known that most algorithms for the non-distributed setting, including online gradient descent (OGD) [Zinkevich, 2003] and follow-the-regularized-leader (FTRL) [Shalev-Shwartz and Singer, 2007, Xiao, 2009], can be extended into D-OCO by combining with the gossip-based average consensus technique [Xiao and Boyd, 2004, Liu and Morse, 2011]. Notably, by extending FTRL, Wan et al. [2024a, 2025] have achieved nearly optimal regret bounds for D-OCO with exact communication. In contrast, only a few previous studies have considered D-OCO with compressed communication.

*Corresponding author.

Table 1: Comparison of our bandit algorithm and the previous best one. Abbreviations: convex → cvx, strongly convex → scvx.

Assumption	Reference	Regret Bound	Communication Rounds
$f_{t,i}(\cdot)$: cvx	Yang et al. [2026]	$\tilde{O}(\omega^{-1/4}\rho^{-1/2}nT^{3/4})$	$O(T)$
	Corollary 3.13	$O(nT^{3/4})$	$\tilde{O}(\omega^{-1}\rho^{-2}\sqrt{T})$
$f_{t,i}(\cdot)$: scvx	Yang et al. [2026]	$\tilde{O}(\omega^{-1/3}\rho^{-2/3}nT^{2/3}(\log T)^{1/3})$	$O(T)$
	Corollary 3.14	$O(nT^{2/3}(\log T)^{1/3})$	$\tilde{O}(\omega^{-1}\rho^{-2}T^{1/3}(\log T)^{2/3})$

To be precise, two concurrent studies [Tu et al., 2022, Cao and Başar, 2023] propose to combine a decentralized variant of OGD (D-OGD) [Yan et al., 2013] with Choco-Gossip [Koloskova et al., 2019]—a compressed variant of the standard gossip technique [Xiao and Boyd, 2004]. Unfortunately, the proposed algorithm is unsatisfactory, because its regret has a large dependence on the spectral gap $\rho \in (0, 1]$ of the gossip matrix, the compression ratio $\omega \in (0, 1]$, and n. To address this issue, Yang et al. [2026] propose an improved algorithm with $\tilde{O}(\omega^{-1/2}\rho^{-1}n\sqrt{T})$ and $\tilde{O}(\omega^{-1}\rho^{-2}n\log T)$ regret bounds for convex and strongly convex functions, respectively.[2] This algorithm is still a combination of D-OGD and Choco-Gossip, though it introduces a two-level blocking update mechanism and a repeated compressor to further reduce the consensus error.

Thus, it remains unclear whether the FTRL-type algorithm for D-OCO can also be extended into the case with compressed communication, and whether such an extension provides any benefits relative to existing algorithms. In this paper, we provide an affirmative answer to both questions. Specifically, we first propose a compressed variant of the FTRL-type algorithm, and show that it enjoys the same regret bounds as the algorithm of Yang et al. [2026]. Despite no improvement in the bounds, both our algorithm and analysis are much simpler than those of Yang et al. [2026]. The key insight is that for the FTRL-type algorithm, the decision is determined by a local approximation of the cumulative average gradient, i.e., the so-called dual variable. Although the mapping from the dual variable to the decision is equivalent to a projection onto the set $\mathcal{K}$, the consensus error of these dual variables is not affected by the projection. For this reason, it can be controlled by a simple application of Choco-Gossip.

Furthermore, we extend our first algorithm into a more challenging bandit setting, where only the loss value $f_{t,i}(\mathbf{x}_i(t))$ is revealed to each learner i. Note that previous studies [Tu et al., 2022, Yang et al., 2026] have also considered this setting. The bandit algorithm of Yang et al. [2026] achieves the best existing regret bounds, i.e., $\tilde{O}(\omega^{-1/4}\rho^{-1/2}nT^{3/4})$ and $\tilde{O}(\omega^{-1/3}\rho^{-2/3}nT^{2/3}(\log T)^{1/3})$ for convex and strongly convex functions, respectively. We demonstrate that our bandit algorithm is not only simpler than that of Yang et al. [2026], but also improves their two bounds to $O(nT^{3/4})$ and $O(nT^{2/3}(\log T)^{1/3})$, respectively. Moreover, different from the algorithm of Yang et al. [2026] that requires T communication rounds in total, our bandit algorithm only requires $\tilde{O}(\omega^{-1}\rho^{-2}\sqrt{T})$ and $\tilde{O}(\omega^{-1}\rho^{-2}T^{1/3}(\log T)^{2/3})$ communication rounds to achieve these two bounds. A detailed comparison of our bandit algorithm and that of Yang et al. [2026] is presented in Table 1.

2 Related work

Now, we briefly review related work on the standard D-OCO, and compressed communication.

The standard D-OCO. The study of D-OCO dates back to the pioneering work of Yan et al. [2013], in which D-OGD is proposed to achieve $O(n^{5/4}\rho^{-1/2}\sqrt{T})$ and $O(n^{3/2}\rho^{-1}\log T)$ regret bounds for convex and strongly convex functions, respectively. The key idea of D-OGD is to first apply a standard gossip step [Xiao and Boyd, 2004] over these local decisions, and then perform a projected gradient descent step according to only the local function. Note that the essence of the gossip step is to compute a weighted average of these local variables, which needs to be implemented via a

[2] The $\tilde{O}(\cdot)$ notation hides constant factors as well as polylogarithmic factors in n, but not in T.

communication round, and the weights are specified by the gossip matrix. Concurrently, Hosseini et al. [2013] propose a decentralized variant of FTRL (D-FTRL) [Shalev-Shwartz and Singer, 2007, Xiao, 2009], and achieve the same $O(n^{5/4}\rho^{-1/2}\sqrt{T})$ regret bound for convex functions. Different from D-OGD, it uses the standard gossip step to update the dual variables of local decisions.

After that, there has been a growing research interest in developing projection-free algorithms for D-OCO [Zhang et al., 2017, Wan et al., 2020, 2022, Wang et al., 2023]. These algorithms are motivated by the fact that both D-OGD and D-FTRL require a projection to ensure the feasibility of each decision, which can be time-consuming in applications with a complex set $\mathcal{K}$. Although the design of projection-free algorithms is beyond the scope of this paper, Wan et al. [2022] have also proposed a projection-free and generalized variant of D-FTRL for the bandit setting. For convex and strongly convex functions, it achieves $O(n^{5/4}\rho^{-1/2}T^{3/4})$ and $O(n^{3/2}\rho^{-1}T^{2/3}(\log T)^{1/3})$ regret bounds, respectively. More interestingly, it only requires $O(\sqrt{T})$ and $O(T^{1/3}(\log T)^{2/3})$ communication rounds, when handling these two types of functions. Such improvements are due to a blocking update mechanism, i.e., dividing the T rounds into blocks and only updating the decision once per block.

Recently, Wan et al. [2024a, 2025] propose the full-information algorithm called accelerated decentralized follow-the-generalized-leader (AD-FTGL), which enjoys $\tilde{O}(n\rho^{-1/4}\sqrt{T})$ and $\tilde{O}(n\rho^{-1/2}\log T)$ regret bounds for convex and strongly convex functions, respectively. This algorithm is also a variant of D-FTRL, and the critical change is to exploit an accelerated gossip strategy [Liu and Morse, 2011]. They have provided nearly matching lower bounds to demonstrate its optimality. Moreover, Wan et al. [2025] have also proposed a projection-free variant of AD-FTGL with $O(nT^{3/4})$ and $O(nT^{2/3}(\log T)^{1/3})$ regret bounds for convex and strongly convex functions, while communicating $\tilde{O}(\rho^{-1/2}\sqrt{T})$ and $\tilde{O}(\rho^{-1/2}T^{1/3}(\log T)^{2/3})$ rounds, respectively. Although this variant is developed for the full-information setting, it actually can be simply extended into the bandit setting while keeping the same regret bounds and communication costs. Compared with the projection-free bandit algorithm in Wan et al. [2022], the improvements on these regret bounds benefit from the accelerated gossip strategy and some sacrifices of communication costs. In addition, several recent studies have further extended D-OCO with exact communication to handle changing environments [Wan, 2025], differential privacy [Wang et al., 2025], exp-concave losses [Qiu et al., 2026], and continuous submodular maximization [Wan et al., 2026].

Compressed communication. Compressed communication is a popular strategy for reducing the amount of data that has to be sent during each iteration of distributed optimization. For the centralized distributed offline optimization, it is well-known that there are many techniques for compressed communication (also known as compressors), which can be divided into quantization [Alistarh et al., 2017, Wen et al., 2017, Bernstein et al., 2018, Seide et al., 2014], sparsification [Alistarh et al., 2018, Stich et al., 2018, Wangni et al., 2018, Aji and Heafield, 2017, Lin et al., 2018], and their hybrid combination [Basu et al., 2019, Wang et al., 2018]. For the decentralized offline optimization, Tang et al. [2018] is the first work to apply the idea of compressed communication, but only focus on unbiased compressors with high accuracy constraints. To tackle this issue, Koloskova et al. [2019] propose a novel and unified method called Choco-Gossip for average consensus with different compressors in the decentralized setting, as well as a corresponding Choco-SGD algorithm for the decentralized offline optimization.

The study of D-OCO with compressed communication actually can be dated back to Li et al. [2021]. However, they only propose a heuristic approach without any theoretical guarantee. By combining D-OGD [Yan et al., 2013] with Choco-Gossip [Koloskova et al., 2019], Tu et al. [2022] and Cao and Başar [2023] concurrently propose the first algorithm to establish rigorous regret bounds for D-OCO with compressed communication. Specifically, it can respectively achieve $O(C_1\sqrt{T})$ and $O(C_1\log T)$ regret bounds for convex and strongly convex functions, where $C_1 = \max\{\omega^{-2}\rho^{-4}n^{3/2}, \omega^{-4}\rho^{-8}n\}$. Additionally, Tu et al. [2022] have also extended this algorithm into the bandit setting, and the previous two bounds respectively degrade to $O(C_2T^{3/4})$ and $O(C_3T^{2/3}(\log T)^{1/3})$, where $C_2 = \max\{\omega^{-1}\rho^{-2}n^{5/4}, \omega^{-2}\rho^{-4}n\}$ and $C_3 = \max\{\omega^{-2/3}\rho^{-4/3}n^{7/6}, \omega^{-8/3}\rho^{-4}n\}$.

Very recently, Yang et al. [2026] further improve these two algorithms of Tu et al. [2022]. In the full-information setting, their algorithm enjoys $\tilde{O}(\omega^{-1/2}\rho^{-1}n\sqrt{T})$ and $\tilde{O}(\omega^{-1}\rho^{-2}n\log T)$ regret bounds for convex and strongly convex functions, respectively. In the bandit setting, their improved algorithm can achieve $\tilde{O}(\omega^{-1/4}\rho^{-1/2}nT^{3/4})$ and $\tilde{O}(\omega^{-1/3}\rho^{-2/3}nT^{2/3}(\log T)^{1/3})$ regret bounds for convex

and strongly convex functions, respectively. Despite such improvements, their algorithms introduce a complicated scheme that combines a two-level blocking update mechanism with Choco-Gossip and a repeated compressor.

3 Main results

In this section, we first introduce necessary assumptions, and then present our two FTRL-type algorithms for D-OCO with compressed communication. The proofs of all theoretical results can be found in the appendix.

3.1 Assumptions

Similar to previous studies on D-OCO with compressed communication [Tu et al., 2022, Yang et al., 2026], we first introduce the following assumptions.

Assumption 3.1. Let $A \in \mathbb{R}^{n\times n}$ denote the gossip matrix. Define the network by a connected undirected graph $\mathcal{G} = ([n], E)$, where $E \subseteq [n] \times [n]$ denotes the edge set. The matrix A is supported on the graph $\mathcal{G}$ and doubly stochastic, i.e.,

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

Moreover, A is symmetric and positive semidefinite.

Assumption 3.2. Let $Q(\cdot) : \mathbb{R}^d \to \mathbb{R}^d$ denote a compressor, whose output can be encoded with fewer bits than the original input. For some $\omega \in (0, 1]$, it satisfies

$$\mathbb{E}_Q[\|Q(\mathbf{x}) - \mathbf{x}\|_2^2] \leq (1 - \omega)\|\mathbf{x}\|_2^2, \quad \forall \mathbf{x} \in \mathbb{R}^d,$$

where $\mathbb{E}_Q$ denotes the expectation over the possible randomness of Q.

Assumption 3.3. There exist two constants $r, R > 0$ such that $r\mathcal{B}^d \subseteq \mathcal{K} \subseteq R\mathcal{B}^d$, where $\mathcal{B}^d$ denotes the unit Euclidean ball centered at the origin in $\mathbb{R}^d$.

Assumption 3.4. At each round $t \in [T]$, the loss function $f_{t,i}(\mathbf{x})$ of each learner $i \in [n]$ is G-Lipschitz over $\mathcal{K}$, i.e., it holds that $|f_{t,i}(\mathbf{x}) - f_{t,i}(\mathbf{y})| \leq G\|\mathbf{x} - \mathbf{y}\|_2, \ \forall \mathbf{x}, \mathbf{y} \in \mathcal{K}$.

Assumption 3.5. At each round $t \in [T]$, the loss function $f_{t,i}(\mathbf{x})$ of each learner $i \in [n]$ is α-strongly convex over $\mathcal{K}$, i.e., it holds that $f_{t,i}(\mathbf{y}) \geq f_{t,i}(\mathbf{x}) + \langle \nabla f_{t,i}(\mathbf{x}), \mathbf{y} - \mathbf{x}\rangle + \frac{\alpha}{2}\|\mathbf{y} - \mathbf{x}\|_2^2, \ \forall \mathbf{x}, \mathbf{y} \in \mathcal{K}$.

Assumption 3.6. At each round t, the loss function $f_{t,i}(\mathbf{x})$ of each learner $i \in [n]$ is bounded over $\mathcal{K}$, i.e., $|f_{t,i}(\mathbf{x})| \leq M$ for any $\mathbf{x} \in \mathcal{K}$. Moreover, all loss functions are chosen beforehand, i.e., the adversary is oblivious.

Remark. First, from Assumption 3.1, the eigenvalues of A can be denoted as $1 = \lambda_1(A) > \lambda_2(A) \geq \cdots \geq \lambda_n(A)$, ensuring a positive spectral gap $\rho = 1 - \lambda_2(A) > 0$. Second, as discussed in Koloskova et al. [2019], Assumption 3.2 captures a broad class of both biased and unbiased compressors (see Appendix I for detailed discussions). Third, Assumption 3.5 with $\alpha = 0$ reduces to the case with general convex functions, which are also considered in this paper. Finally, Assumptions 3.3 and 3.6 are mainly required by the bandit setting [Flaxman et al., 2005, Wan et al., 2022, 2024b]. One can verify that in the full-information setting, the former can be simplified to $\|\mathbf{x}\|_2 \leq R, \forall \mathbf{x} \in \mathcal{K}$, and the latter can be removed.

3.2 Algorithm for full-information setting

Before introducing our algorithms, we first briefly recall the best existing algorithm for D-OCO with exact communication, i.e., AD-FTGL [Wan et al., 2024a, 2025]. Specifically, it adopts a blocking update mechanism with block size L, i.e., dividing the total T rounds into T/L blocks,[3] and only maintaining a decision $\mathbf{x}_i(z)$ for all rounds in each block z. The initial decision is simply set as $\mathbf{x}_i(1) = \mathbf{0}$. To update the decision, it exploits an accelerated gossip strategy [Liu and Morse, 2011] to maintain a dual variable $\mathbf{z}_i(z) \approx \sum_{\tau=1}^{z-1}(1/n)\sum_{i=1}^{n}\mathbf{d}_i(\tau)$, where $\mathbf{d}_i(\tau) = \sum_{t\in\mathcal{T}_\tau}(\nabla f_{t,i}(\mathbf{x}_i(\tau)) -$

[3] Without loss of generality, T/L is assumed to be an integer.

Algorithm 1 Choco-Gossip (CG)

Input: Original vectors $\{\mathbf{z}_i^0\}_{i\in[n]}$, initial compressed vectors $\{\hat{\mathbf{z}}_i^0\}_{i\in[n]}$, initial intermediate vectors $\{\mathbf{s}_i^0\}_{i\in[n]}$, step size γ, number of iterations K
for $k = 0 \dots K-1$ **in parallel** for each $i \in [n]$ **do**
  Compress the difference as $\mathbf{q}_i^k = Q(\mathbf{z}_i^k - \hat{\mathbf{z}}_i^k)$
  Send $\mathbf{q}_i^k$ to neighbors and receive $\mathbf{q}_j^k$
  $\hat{\mathbf{z}}_i^{k+1} = \hat{\mathbf{z}}_i^k + \mathbf{q}_i^k$, $\mathbf{s}_i^{k+1} = \mathbf{s}_i^k + \sum_{j=1}^n A_{ij}\mathbf{q}_j^k$
  $\mathbf{z}_i^{k+1} = \mathbf{z}_i^k + \gamma(\mathbf{s}_i^{k+1} - \hat{\mathbf{z}}_i^{k+1})$
end for

$\alpha\mathbf{x}_i(\tau))$ and $\mathcal{T}_\tau = \{(\tau-1)L+1, \dots, \tau L\}$. Based on $\mathbf{z}_i(z)$, it computes

$$\mathbf{x}_i(z+1) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \langle \mathbf{z}_i(z), \mathbf{x}\rangle + \frac{(z-1)L\alpha + 2h}{2}\|\mathbf{x}\|_2^2, \tag{1}$$

where h is a parameter. Let $\bar{\mathbf{d}}(z) = (1/n)\sum_{i=1}^n \mathbf{d}_i(z)$ and $\bar{\mathbf{z}}(z) = \sum_{\tau=1}^{z-1}\bar{\mathbf{d}}(\tau)$. According to their analysis, the regret of AD-FTGL can be decomposed into two parts: one about the cumulative consensus error of $\|\mathbf{z}_i(z) - \bar{\mathbf{z}}(z)\|_2$ and the other about the regret of a virtual update based on $\bar{\mathbf{z}}(z)$.

Therefore, the critical challenge for extending AD-FTGL into D-OCO with compressed communication becomes how to maintain $\mathbf{z}_i(z)$ with a small consensus error in this case. To this end, we introduce Choco-Gossip [Koloskova et al., 2019], whose detailed procedure is outlined in Algorithm 1, and the following lemma.

Lemma 3.7. *Let* $\beta := \|I - A\|_2$, $\bar{\mathbf{z}}^k = (1/n)\sum_{i=1}^n \mathbf{z}_i^k$ *and* $e^k = \mathbb{E}[\sum_{i=1}^n \|\mathbf{z}_i^k - \bar{\mathbf{z}}^k\|_2^2] + \mathbb{E}[\sum_{i=1}^n \|\mathbf{z}_i^k - \hat{\mathbf{z}}_i^k\|_2^2]$, *where* I *is the identity matrix. Under Assumptions 3.1 and 3.2, Algorithm 1 with* $\mathbf{s}_i^0 = \sum_{j=1}^n A_{ij}\hat{\mathbf{z}}_j^0, \forall i \in [n]$ *and*

$$\gamma = \frac{\rho\omega}{2(1-\omega)(\rho+2)\beta^2 + 16\rho - 8\rho\omega + \rho^2}, \tag{2}$$

ensures $e^k \le (1-\gamma\rho/2)^k e^0$ *for any* $k \in [K]$.

Given n local vectors $\{\mathbf{z}_i^0\}_{i\in[n]}$ in a decentralized network, the role of Choco-Gossip is to approximately compute their average $\bar{\mathbf{z}} = (1/n)\sum_{i=1}^n \mathbf{z}_i^0$ via only compressed communication, which is known as the average consensus problem. Since it is easy to verify that $\bar{\mathbf{z}}^k = \bar{\mathbf{z}}$, Lemma 3.7 is sufficient to provide an upper bound on the consensus error.

Remark. Note that Algorithm 1 follows a memory-efficient implementation of Choco-Gossip, which can be found in the appendix of Koloskova et al. [2019]. Moreover, our Lemma 3.7 actually is a generalized and refined version of the original guarantee on Choco-Gossip. To be precise, Theorem 2 of Koloskova et al. [2019] only shows $e^k \le (1-\omega\rho^2/82)^k e^0$ with the initialization $\hat{\mathbf{z}}_i^0 = \mathbf{0}$ (and $\mathbf{s}_i^0 = \mathbf{0}$ for the memory-efficient implementation), which cannot recover the classical $O((1-\rho)^k)$ consensus error bound of the standard gossip [Xiao and Boyd, 2004] even if $\omega = 1$. In contrast, our lemma holds for more flexible initialization with a worst-case bound of $O((1-\omega\rho^2)^k)$. Specifically, when $\omega = 1$, we have $\gamma = 1/(8+\rho)$ and our bound becomes $e^k \le (1-\rho/(16+2\rho))^k e^0$, which recovers the classical $O((1-\rho)^k)$ consensus error bound of standard gossip, whereas that of Koloskova et al. [2019] remains $O((1-\rho^2)^k)$. Such initialization makes Choco-Gossip plug-and-play in our algorithms, and the improvement may be also of independent interest.

Now, we are ready to combine AD-FTGL with only compressed communication. For brevity, let $\mathrm{CG}(\cdot,\cdot,\cdot,\cdot,\cdot)$ denote Choco-Gossip in Algorithm 1. A natural idea is to apply Choco-Gossip over $\{\mathbf{d}_i(z-1)\}_{i\in[n]}$ during each block z to generate

$$\{\mathbf{d}_i^K(z-1)\}_{i\in[n]} = \mathrm{CG}(\{\mathbf{d}_i(z-1)\}_{i\in[n]}, \{\mathbf{0}\}_{i\in[n]}, \{\mathbf{0}\}_{i\in[n]}, \gamma, K),$$

and then set $\mathbf{z}_i(z) = \sum_{\tau=1}^{z-1}\mathbf{d}_i^K(\tau)$. However, in this way, we need to set K proportional to $O(\log T)$ for making the cumulative consensus error sufficiently small. To address this issue, we exploit Choco-Gossip in a more careful way. Specifically, we simply set $\mathbf{z}_i(1) = \hat{\mathbf{z}}_i(1) = \mathbf{s}_i(1) = \mathbf{0}$ for any

Algorithm 2 CD-FTGL

1: **Input:** strongly convex factor α, regularization parameter h, block size L, consensus step size γ
2: $\mathbf{x}_i(1) = \mathbf{z}_i(1) = \hat{\mathbf{z}}_i(1) = \mathbf{s}_i(1) = \mathbf{0}, \forall i \in [n]$
3: **for** $z = 1, \ldots, T/L$ **in parallel** for each $i \in [n]$ **do**
4: **for** $t = (z-1)L + 1, \ldots, zL$ **do**
5: Play $\mathbf{x}_i(z)$ and observe $\nabla f_{t,i}(\mathbf{x}_i(z))$
6: **end for**
7: **if** $z \geq 2$ **then**
8: $\mathbf{z}_i^0(z) = \mathbf{z}_i(z-1) + \mathbf{d}_i(z-1)$, $\hat{\mathbf{z}}_i^0(z) = \hat{\mathbf{z}}_i(z-1)$, and $\mathbf{s}_i^0(z) = \mathbf{s}_i(z-1)$
9: Set $K = L$ and compute $\mathbf{z}_i^K(z), \hat{\mathbf{z}}_i^K(z), \mathbf{s}_i^K(z)$ by invoking Algorithm 1 as in (3)
10: $\mathbf{z}_i(z) = \mathbf{z}_i^K(z)$, $\hat{\mathbf{z}}_i(z) = \hat{\mathbf{z}}_i^K(z)$, $\mathbf{s}_i(z) = \mathbf{s}_i^K(z)$
11: **end if**
12: $\mathbf{d}_i(z) = \sum_{t \in \mathcal{T}_z} (\nabla f_{t,i}(\mathbf{x}_i(z)) - \alpha \mathbf{x}_i(z))$
13: Compute $\mathbf{x}_i(z+1)$ via (1)
14: **end for**

$i \in [n]$. For each block $z \geq 2$, we first set $\mathbf{z}_i^0(z) = \mathbf{z}_i(z-1) + \mathbf{d}_i(z-1)$, $\hat{\mathbf{z}}_i^0(z) = \hat{\mathbf{z}}_i(z-1)$, and $\mathbf{s}_i^0(z) = \mathbf{s}_i(z-1)$. Then, we apply Choco-Gossip as

$$\begin{aligned} &\left(\{\mathbf{z}_i^K(z)\}_{i\in[n]}, \{\hat{\mathbf{z}}_i^K(z)\}_{i\in[n]}, \{\mathbf{s}_i^K(z)\}_{i\in[n]}\right) \\ &= \mathrm{CG}(\{\mathbf{z}_i^0(z)\}_{i\in[n]}, \{\hat{\mathbf{z}}_i^0(z)\}_{i\in[n]}, \{\mathbf{s}_i^0(z)\}_{i\in[n]}, \gamma, K), \end{aligned} \tag{3}$$

and set $\mathbf{z}_i(z) = \mathbf{z}_i^K(z)$, $\hat{\mathbf{z}}_i(z) = \hat{\mathbf{z}}_i^K(z)$, and $\mathbf{s}_i(z) = \mathbf{s}_i^K(z)$. In this way, once $\mathbf{d}_i(z-1)$ is incorporated into $\mathbf{z}_i(z)$, it will also be affected by the subsequent Choco-Gossip. Thus, K can be independent of T.

From the above discussions, we propose our first algorithm called compressed decentralized follow-the-generalized-leader (CD-FTGL). The detailed procedure is summarized in Algorithm 2.

Remark. First, we want to emphasize that in step 9 of Algorithm 2, the invocation of Algorithm 1 can be implemented in parallel to the *for* loop from steps 4 to 6, which implies that both the computation and communication costs can be allocated to each round in block z. Second, we notice that Yang et al. [2026] also exploit Choco-Gossip in a similar way, but to control the consensus error among decisions. Moreover, according to the OGD-type update of decision, they need to use a more complicated blocking mechanism and a repeated compressor to eliminate the effect of a projection operation on the consensus error. In contrast, in our algorithm, the consensus error can be controlled independent of the projection.

By combining the existing analysis of AD-FTGL [Wan et al., 2024a, 2025] and Lemma 3.7, we first establish a general guarantee on the regret of Algorithm 2.

Theorem 3.8. *Under Assumptions 3.1, 3.2, 3.3, 3.4, and 3.5, for any $i \in [n]$, Algorithm 2 with $L = \lceil 2\ln(n+2)/(\gamma\rho) \rceil$, where γ is defined in (2), ensures*

$$\mathbb{E}[R_{T,i}] \leq nhR^2 + \sum_{z=2}^{T/L} \frac{6nL^2G(G+\alpha R)}{(z-2)L\alpha + 2h} + \sum_{z=1}^{T/L} \frac{12nL^2G(G+2\alpha R)}{zL\alpha + 2h}. \tag{4}$$

By further tuning h, we obtain the following regret bounds for convex and strongly convex functions.

Corollary 3.9. *Under Assumptions 3.1, 3.2, 3.3, 3.4, and 3.5 with $\alpha = 0$, for any $i \in [n]$, Algorithm 2 with $h = 3\sqrt{LT}G/R$ and $L = \lceil 2\ln(n+2)/(\gamma\rho) \rceil$ ensures*

$$\mathbb{E}[R_{T,i}] \leq 6nGR\sqrt{LT}.$$

Corollary 3.10. *Under Assumptions 3.1, 3.2, 3.3, 3.4, and 3.5 with $\alpha > 0$, for any $i \in [n]$, Algorithm 2 with $h = \alpha L$ and $L = \lceil 2\ln(n+2)/(\gamma\rho) \rceil$ ensures*

$$\mathbb{E}[R_{T,i}] \leq n\alpha R^2 L + \frac{6nG(3G+5\alpha R)L}{\alpha} \ln(T/L).$$

Remark. From these two corollaries and the definition of γ in (2), our CD-FTGL achieves $\tilde{O}(\omega^{-1/2}\rho^{-1}n\sqrt{T})$ and $\tilde{O}(\omega^{-1}\rho^{-2}n\log T)$ regret bounds for convex and strongly convex functions,

respectively. Notably, our results match the current best upper bounds achieved by Yang et al. [2026]. Moreover, Yang et al. [2026] also established $\Omega(\omega^{-1/2}\rho^{-1/4}n\sqrt{T})$ and $\Omega(\omega^{-1}\rho^{-1/2}n\log T)$ lower bounds for these two settings. This indicates that our dependencies on ω, n and T are already nearly optimal. Furthermore, based on the discussions about the consensus error in Lemma 3.7, if $\omega = 1$, our two regret bounds improve to $\tilde{O}(\rho^{-1/2}n\sqrt{T})$ and $\tilde{O}(\rho^{-1}n\log T)$, which recovers the best existing results for D-OCO algorithms with standard gossip [Xiao and Boyd, 2004].[4] In contrast, even if $\omega = 1$, the regret bounds of Yang et al. [2026] will only become $\tilde{O}(\rho^{-1}n\sqrt{T})$ and $\tilde{O}(\rho^{-2}n\log T)$.

3.3 Algorithm for bandit setting

Compared with the full-information setting, the critical challenge of the bandit setting is that only the value of each local function at the played decision can be observed. Fortunately, in the literature, there exists a well-known technique called the one-point estimator [Flaxman et al., 2005] that can generate an approximate gradient by using a single loss value. It can be formalized as the following lemma, where $\mathcal{B}^d$ and $\mathcal{S}^d$ denote the unit ball and sphere centered at the origin in $\mathbb{R}^d$, respectively, and $\mathbf{u} \sim \mathcal{B}^d$ (or $\mathbf{u} \sim \mathcal{S}^d$) means that $\mathbf{u}$ is uniformly sampled from $\mathcal{B}^d$ (or $\mathcal{S}^d$).

Lemma 3.11. *(Lemma 1 in Flaxman et al. [2005]) For any function $f(\mathbf{x}) : \mathcal{K} \mapsto \mathbb{R}$ and $\epsilon > 0$, we define its ϵ-smoothed version as $\hat{f}_\epsilon(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{B}^d}[f(\mathbf{x}+\epsilon\mathbf{u})]$. Then, it holds that*

$$\nabla\hat{f}_\epsilon(\mathbf{x}) = \mathbb{E}_{\mathbf{u}\sim\mathcal{S}^d}\left[\frac{d}{\epsilon}f(\mathbf{x}+\epsilon\mathbf{u})\mathbf{u}\right].$$

By combining our CD-FTGL with this technique, our bandit algorithm is outlined in Algorithm 3, and named as compressed decentralized follow-the-bandit-leader (CD-FTBL).

Specifically, compared with CD-FTGL, there exist some critical differences. First, to apply the one-point estimator, in step 5 of Algorithm 3, we play $\mathbf{y}_i(t) = \mathbf{x}_i(z) + \epsilon\mathbf{u}_i(t)$, which consists of $\mathbf{x}_i(z)$ computed by using historical information and a random point $\epsilon\mathbf{u}_i(t)$. Due to $\mathbf{u}_i(t) \sim \mathcal{S}^d$ and Assumption 3.3, $\mathbf{x}_i(z) \in \mathcal{K}_\epsilon = (1-\epsilon/r)\mathcal{K} = \{(1-\epsilon/r)\mathbf{x} \mid \mathbf{x} \in \mathcal{K}\}$ is a sufficient condition for $\mathbf{y}_i(t) \in \mathcal{K}$. Therefore, the second difference is that $\mathbf{x}_i(z+1)$ in step 14 of Algorithm 3 should be computed over the shrunk set $\mathcal{K}_\epsilon$, i.e.,

$$\mathbf{x}_i(z+1) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}_\epsilon}\langle\mathbf{z}_i(z),\mathbf{x}\rangle + \frac{(z-1)L\alpha+2h}{2}\|\mathbf{x}\|_2^2. \tag{5}$$

Third, instead of querying $\nabla f_{t,i}(\mathbf{y}_i(t))$, in step 6 of Algorithm 3, we compute an approximate one as $\mathbf{g}_i(t) = \frac{d}{\epsilon}f_{t,i}(\mathbf{y}_i(t))\mathbf{u}_i(t)$, which is also used to replace the gradient originally required by $\mathbf{d}_i(z)$ in step 13 of Algorithm 3. Finally, inspired by Wan et al. [2022], we notice that a much larger block size may be used to reduce the number of communication rounds. As a result, in step 10 of Algorithm 3, we no longer set $K = L$ when invoking Choco-Gossip.

Moreover, we have the following guarantee regarding the regret of Algorithm 3.

Theorem 3.12. *Under Assumptions 3.1, 3.2, 3.3, 3.4, 3.5, and 3.6, for any $i \in [n]$, Algorithm 3 with $L \geq K = \lceil 2\ln(n+2)/(\gamma\rho)\rceil$ and $\epsilon < r$, ensures*

$$\begin{aligned}\mathbb{E}[R_{T,i}] \leq nhR^2 + 3\epsilon nGT + \frac{\epsilon nGRT}{r} + \sum_{z=1}^{T/L}\frac{2nB_1^2}{zL\alpha+2h} \\ + \sum_{z=1}^{T/L}\frac{6nLGB_1}{(z-1)L\alpha+2h} + \sum_{z=2}^{T/L}\frac{6nLGB_2}{(z-2)L\alpha+2h},\end{aligned} \tag{6}$$

where $B_1 = 2\sqrt{L}dM/\epsilon + 2LG + 3\alpha LR$ and $B_2 = \sqrt{2Ld^2M^2/\epsilon^2 + 2L^2G^2 + 2\alpha^2L^2R^2}$.

By further tuning h and ϵ, we obtain specific regret bounds for convex and strongly convex functions.

Corollary 3.13. *Under Assumptions 3.1, 3.2, 3.3, 3.4, 3.5 with $\alpha = 0$, and 3.6, for any $i \in [n]$, Algorithm 3 with $K = \lceil 2\ln(n+2)/(\gamma\rho)\rceil$, $L = \max\{K, \sqrt{T}\}$, $h = \sqrt{dLT}M/R$, and $\epsilon = c\sqrt{d}\,T^{-1/4}$, where c is a constant such that $\epsilon < r$, ensures*

$$\mathbb{E}[R_{T,i}] = O(nT^{3/4} + n\sqrt{KT}).$$

[4] Actually, the original analysis of both D-OGD and D-FTRL does not provide these results, which are first achieved by Wan et al. [2025] with a refined analysis on the consensus error.

Algorithm 3 CD-FTBL

Input: strongly convex factor α, regularization parameter h, block size L, communication rounds K, consensus step size γ, exploration radius ϵ
$\mathbf{x}_i(1) = \mathbf{z}_i(1) = \hat{\mathbf{z}}_i(1) = \mathbf{s}_i(1) = \mathbf{0}, \forall i \in [n]$
for $z = 1, \ldots, T/L$ **in parallel** for each node i **do**
  for $t = (z-1)L + 1, \ldots, zL$ **do**
    Play $\mathbf{y}_i(t) = \mathbf{x}_i(z) + \epsilon \mathbf{u}_i(t)$, where $\mathbf{u}_i(t) \sim \mathcal{S}^d$
    Compute $\mathbf{g}_i(t) = \frac{d}{\epsilon} f_{t,i}(\mathbf{y}_i(t)) \mathbf{u}_i(t)$
  end for
  if $z \geq 2$ **then**
    $\mathbf{z}_i^0(z) = \mathbf{z}_i(z-1) + \mathbf{d}_i(z-1)$, $\hat{\mathbf{z}}_i^0(z) = \hat{\mathbf{z}}_i(z-1)$, and $\mathbf{s}_i^0(z) = \mathbf{s}_i(z-1)$
    Compute $\mathbf{z}_i^K(z), \hat{\mathbf{z}}_i^K(z), \mathbf{s}_i^K(z)$ by invoking Algorithm 1 as in (3)
    $\mathbf{z}_i(z) = \mathbf{z}_i^K(z)$, $\hat{\mathbf{z}}_i(z) = \hat{\mathbf{z}}_i^K(z)$, $\mathbf{s}_i(z) = \mathbf{s}_i^K(z)$
  end if
  $\mathbf{d}_i(z) = \sum_{t \in \mathcal{T}_z} (\mathbf{g}_i(t) - \alpha \mathbf{x}_i(z))$
  Compute $\mathbf{x}_i(z+1)$ via (5)
end for

Corollary 3.14. *Under Assumptions 3.1, 3.2, 3.3, 3.4, 3.5 with $\alpha > 0$, and 3.6, for any $i \in [n]$, Algorithm 3 with $K = \lceil 2\ln(n+2)/(\gamma\rho) \rceil$, $L = \max\{K, T^{2/3}(\ln T)^{-2/3}\}$, $h = \alpha L$, and $\epsilon = cd^{2/3}T^{-1/3}(\ln T)^{1/3}$, where c is a constant such that $\epsilon < r$, ensures*

$$\mathbb{E}[R_{T,i}] = O\left(nT^{2/3}(\ln T)^{1/3} + nK\log T\right).$$

Remark. We first notice that in the above two regret bounds, only the non-dominant term depends on ρ and ω via the definition of K. This implies that our CD-FTBL enjoys a nice ability to decouple the joint effect of the bandit feedback and the decentralized compressed communication. Moreover, as discussed in previous studies [Wan et al., 2025, Wang et al., 2025], the total number of rounds T is commonly much larger than other problem constants. By combining Corollaries 3.13 and 3.14 with this assumption, the two regret bounds can be simplified to $O(nT^{3/4})$ and $O(nT^{2/3}(\log T)^{1/3})$, which are much tighter than the best existing $\tilde{O}(\omega^{-1/4}\rho^{-1/2}nT^{3/4})$ and $\tilde{O}(\omega^{-1/3}\rho^{-2/3}nT^{2/3}(\log T)^{1/3})$ regret bounds for convex and strongly convex functions [Yang et al., 2026]. While CD-FTGL and previous algorithms [Yang et al., 2026] require $C = T$, our CD-FTBL requires only $C = TK/L$ communication rounds. Under the same assumption, Corollaries 3.13 and 3.14 only require $\tilde{O}(\omega^{-1}\rho^{-2}\sqrt{T})$ and $\tilde{O}(\omega^{-1}\rho^{-2}T^{1/3}(\log T)^{2/3})$ communication rounds, respectively. [5] To the best of our knowledge, this is the first D-OCO algorithm that can reduce the number of communication rounds and the bits of each communication simultaneously.

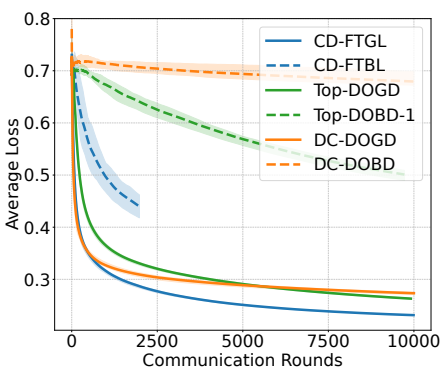
4 Experiments

To verify the performance of our algorithms, we conduct experiments on the decentralized online logistic regression problem with two datasets: ijcnn1 and a9a from the LIBSVM repository [Chang and Lin, 2011]. [6] The loss function for learner i at round t is given by $f_{t,i}(\mathbf{x}) = \ln(1 + \exp(-b_{t,i}\langle \mathbf{a}_{t,i}, \mathbf{x}\rangle))$, where $\mathbf{a}_{t,i} \in \mathbb{R}^d$ is the feature vector and $b_{t,i} \in \{-1, 1\}$ is the class label. The decision set is defined as a bounded Euclidean ball $\mathcal{K} = \{\mathbf{x} \in \mathbb{R}^d \mid \|\mathbf{x}\|_2 \leq \tau\}$, where $\tau = 10$. We adopt the average cumulative loss, defined as $AL(t,i) = \frac{1}{tn}\sum_{\tau=1}^{t}\sum_{j=1}^{n} f_{\tau,j}(\mathbf{x}_i(\tau))$, as the performance metric. Moreover, the original ijcnn1 and a9a datasets consist of 49990 and 32561 examples, respectively. To ensure $T = 10000$, we first make a random permutation of these examples, and then allocate 10000 examples to each local learner cyclically.

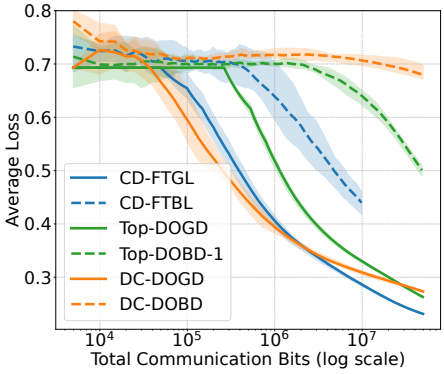
The simulation involves $n = 9$ learners connected via a $\mathcal{G}(n, 2n)$ random graph, which is generated by a tool called NetworkX [Hagberg et al., 2008]. Based on this graph topology, the gossip matrix

[5] With $b_Q(\omega, d)$ bits per message and $M_{\mathcal{G}}$ messages per round, C rounds cost $O(b_Q(\omega, d)M_{\mathcal{G}}C)$ bits. As all algorithms here share the same ω and $M_{\mathcal{G}}$, this cost is proportional to C, so comparing rounds is equivalent to comparing total bits.

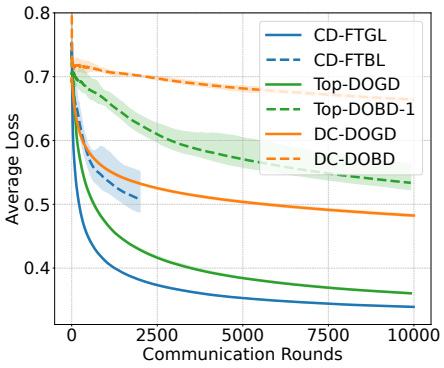
[6] All experiments were conducted on a workstation running Windows 11 with an Intel Core i7-14700 CPU and 16 GB RAM, without GPU acceleration. Our source code is publicly available at https://github.com/SSSS88888/CD-FTGL.



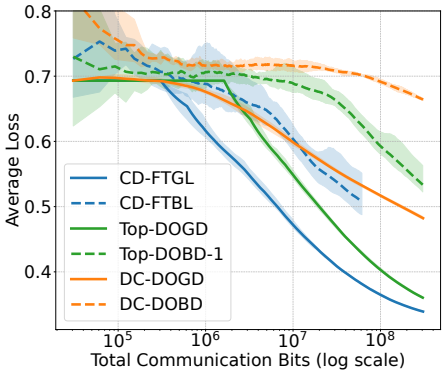
(a) ijcnn1: Loss vs Rounds



(b) ijcnn1: Loss vs Bits



(c) a9a: Loss vs Rounds



(d) a9a: Loss vs Bits

Figure 1: Experimental results on a small random graph (9 nodes and 18 edges) with $\omega = 0.1$.

A satisfying Assumption 3.1 is constructed using the local-degree weights [Xiao and Boyd, 2004], where $A_{ij} = 1/\max\{d_i, d_j\}$ for connected nodes i and j, and $A_{ii} = 1 - \sum_{j \neq i} A_{ij}$. We choose Top-DOGD [Yang et al., 2026] and DC-DOGD [Tu et al., 2022] as the full-information baselines, and Top-DOBD-1 [Yang et al., 2026] and DC-DOBD [Tu et al., 2022] as the bandit baselines. In all experiments, we use the Top-k compressor to reduce communication overhead, and each algorithm is run 5 times, and the mean performance with shaded regions indicating the min-max range is reported.

To ensure a fair comparison, we tune hyper-parameters via grid search. Specifically, the theoretical values of regularization parameter h and learning rate η are multiplied by the scaling coefficients, which are selected from $\{10^{-5}, 10^{-4}, \ldots, 10^{5}\}$. The exploration radius ϵ is chosen from $\{0.1, 1, 10\}$, while the block size L (and L_1 for Top-DOGD and Top-DOBD-1) is tuned over $\{10, 100, 1000\}$. The compression ratio is uniformly set to $\omega = 0.1$ across both datasets. Other parameters remain at their suggested theoretical values.

Figure 1 plots the average loss against the number of communication rounds and transmitted bits for both datasets. In both the full-information and bandit settings, our algorithms achieve a lower average loss compared to their respective baselines. More significantly, our CD-FTBL requires only about 2,000 communication rounds, whereas the bandit baselines require at least 10,000 rounds. To further evaluate the actual communication cost, Figures 1(b) and 1(d) illustrate the average loss with respect to the total transmitted bits. In these plots, the loss curves for our algorithms visibly shift to the left compared to their baselines. This indicates that our algorithms require fewer communication bits to reach the same average loss, and thus verifies their communication efficiency again. Additional experimental results can be found in Appendix A.

5 Conclusion and future work

This paper revisits the problem of D-OCO with compressed communication. Unlike previous studies that focus on OGD-type algorithms, we propose two FTRL-type algorithms, namely CD-FTGL

and CD-FTBL, which simplify the existing analysis and even achieve improved results. Specifically, our CD-FTGL is developed for the full-information setting, and can match the existing $\tilde{O}(\omega^{-1/2}\rho^{-1}n\sqrt{T})$ and $\tilde{O}(\omega^{-1}\rho^{-2}n\log T)$ regret bounds for convex and strongly convex functions, respectively. Our CD-FTBL can handle the bandit setting, and significantly improves the regret for convex and strongly convex functions to $O(nT^{3/4})$ and $O(nT^{2/3}(\log T)^{1/3})$, respectively. Interestingly, it only requires sublinear communication rounds to achieve these results.

Nonetheless, there are still some open problems. First, as previously discussed, CD-FTGL with $\omega = 1$ can recover the best existing results of full-information D-OCO algorithms with the standard gossip. However, there still exist gaps in terms of ρ from the regret bounds of AD-FTGL [Wan et al., 2024a, 2025], which is based on the accelerated gossip strategy. Thus, it is appealing to investigate whether this acceleration technique can also be combined with compressed communication. Second, we notice that in the full-information setting, the blocking update mechanism is not necessary for the application of the standard gossip [Wan et al., 2025]. Thus, it is also interesting to study whether the same results achieved by our CD-FTGL can be obtained with $L = 1$. Finally, our current theoretical guarantees are derived under synchronous communication. It is worth extending our algorithms to the setting with asynchronous communication [Jiang et al., 2021], which may require techniques [Wan et al., 2024b,c, Bao et al., 2025] for handling the delays incurred by asynchrony.

Acknowledgments and Disclosure of Funding

This work was partially supported by Zhejiang Provincial Natural Science Foundation of China (LMS25F030002), Fundamental Research Funds for the Central Universities (No.226-2025-00057), and Ningbo Yongjiang Talent Introduction Programme (2023A-193-G). The authors would also like to thank the anonymous reviewers for their helpful comments.

References

Alham Fikri Aji and Kenneth Heafield. Sparse communication for distributed gradient descent. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 440–445, 2017.

Dan Alistarh, Demjan Grubic, Jerry Li, Ryota Tomioka, and Milan Vojnovic. QSGD: Communication-efficient sgd via gradient quantization and encoding. In *Advances in neural information processing systems 30*, page 1709–1720, 2017.

Dan Alistarh, Torsten Hoefler, Mikael Johansson, Nikola Konstantinov, Sarit Khirirat, and Cédric Renggli. The convergence of sparsified gradient methods. In *Advances in Neural Information Processing Systems 31*, pages 5977–5987, 2018.

Lingchan Bao, Tong Wei, and Yuanyu Wan. Revisiting multi-agent asynchronous online optimization with delays: the strongly convex case. *arXiv preprint arXiv:2503.10013*, 2025.

Debraj Basu, Deepesh Data, Can Karakus, and Suhas Diggavi. Qsparse-local-SGD: Distributed SGD with quantization, sparsification and local computations. In *Advances in Neural Information Processing Systems 32*, pages 14695–14706, 2019.

Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. signSGD: Compressed optimisation for non-convex problems. In *International conference on machine learning*, pages 560–569, 2018.

Aleksandr Beznosikov, Samuel Horvath, Peter Richtárik, and Mher Safaryan. On biased compression for distributed learning. *J. Mach. Learn. Res.*, 24:276:1–276:50, 2020.

Xuanyu Cao and Tamer Başar. Decentralized online convex optimization with compressed communications. *Automatica*, 156:111186, 2023.

Chih-Chung Chang and Chih-Jen Lin. LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(27):1–27, 2011.

John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12:2121–2159, 2011.

Abraham D. Flaxman, Adam Tauman Kalai, and H. Brendan McMahan. Online convex optimization in the bandit setting: Gradient descent without a gradient. In *Proceedings of the 16th Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 385–394, 2005.

Dan Garber and Elad Hazan. A linearly convergent conditional gradient algorithm with applications to online and stochastic optimization. *SIAM Journal on Optimization*, 26(3):1493–1528, 2016.

Aric Hagberg, Pieter J Swart, and Daniel A Schult. Exploring network structure, dynamics, and function using networkx. Technical report, Los Alamos National Laboratory (LANL), Los Alamos, NM (United States), 2008.

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

Saghar Hosseini, Airlie Chapman, and Mehran Mesbahi. Online distributed optimization via dual averaging. In *52nd IEEE Conference on Decision and Control*, pages 1484–1489. IEEE, 2013.

Jiyan Jiang, Wenpeng Zhang, Jinjie Gu, and Wenwu Zhu. Asynchronous decentralized online learning. *Advances in Neural Information Processing Systems*, 34:20185–20196, 2021.

Anastasia Koloskova, Sebastian U. Stich, and Martin Jaggi. Decentralized stochastic optimization and gossip algorithms with compressed communication. In *Proceedings of the 36th International Conference on Machine Learning*, pages 3478–3487, 2019.

Guangxia Li, Jia Liu, Xiao Lu, Peilin Zhao, Yulong Shen, and Dusit Niyato. Decentralized online learning with compressed communication for near-sensor data analytics. *IEEE Communications Letters*, 25(9):2958–2962, 2021.

Yujun Lin, Song Han, Huizi Mao, Yu Wang, and Bill Dally. Deep Gradient Compression: Reducing the communication bandwidth for distributed training. In *International Conference on Learning Representations*, 2018.

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

Hao Qiu, Mengxiao Zhang, and Yuanyu Wan. Decentralized online exp-concave optimization. *arXiv preprint arXiv:2610.25017*, 2026.

Frank Seide, Hao Fu, Jasha Droppo, Gang Li, and Dong Yu. 1-bit stochastic gradient descent and its application to data-parallel distributed training of speech dnns. In *Proceedings of the 15th Annual Conference of the International Speech Communication Association*, pages 1058–1062, 2014.

Shai Shalev-Shwartz and Yoram Singer. A primal-dual perspective of online learning algorithm. *Machine Learning*, 69(2–3):115–142, 2007.

Sebastian U Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified SGD with memory. In *Advances in Neural Information Processing Systems 31*, pages 4452–4463, 2018.

Hanlin Tang, Shaoduo Gan, Ce Zhang, Tong Zhang, and Ji Liu. Communication compression for decentralized training. *Advances in Neural Information Processing Systems 31*, pages 7663–7673, 2018.

Zhipeng Tu, Xi Wang, Yiguang Hong, Lei Wang, Deming Yuan, and Guodong Shi. Distributed online convex optimization with compressed communication. *Advances in Neural Information Processing Systems 35*, pages 34492–34504, 2022.

Yuanyu Wan. Black-box reductions for decentralized online convex optimization in changing environments. In *Proceedings of the 38th Annual Conference on Learning Theory*, pages 5605–5631, 2025.

Yuanyu Wan, Wei-Wei Tu, and Lijun Zhang. Projection-free distributed online convex optimization with $O(\sqrt{T})$ communication complexity. In *Proceedings of the 37th International Conference on Machine Learning*, pages 9818–9828, 2020.

Yuanyu Wan, Guanghui Wang, Wei-Wei Tu, and Lijun Zhang. Projection-free distributed online learning with sublinear communication complexity. *Journal of Machine Learning Research*, 23 (172):1–53, 2022.

Yuanyu Wan, Tong Wei, Mingli Song, and Lijun Zhang. Nearly optimal regret for decentralized online convex optimization. In *Proceedings of the 37th Annual Conference on Learning Theory*, pages 4862–4888, 2024a.

Yuanyu Wan, Chang Yao, Mingli Song, and Lijun Zhang. Improved regret for bandit convex optimization with delayed feedback. *Advances in Neural Information Processing Systems*, 37: 169–196, 2024b.

Yuanyu Wan, Chang Yao, Mingli Song, and Lijun Zhang. Non-stationary online convex optimization with arbitrary delays. In *Forty-first International Conference on Machine Learning*, pages 49991–50011, 2024c.

Yuanyu Wan, Tong Wei, Bo Xue, Mingli Song, and Lijun Zhang. Optimal and efficient algorithms for decentralized online convex optimization. *Journal of Machine Learning Research*, 26(135): 1–43, 2025.

Yuanyu Wan, Yu Shen, Dingzhi Yu, Bo Xue, and Mingli Song. Improved approximate regret for decentralized online continuous submodular maximization via reductions. *arXiv preprint arXiv:2602.09502*, 2026.

Hongyi Wang, Scott Sievert, Shengchao Liu, Zachary Charles, Dimitris Papailiopoulos, and Stephen Wright. Atomo: Communication-efficient learning via atomic sparsification. *Advances in neural information processing systems 31*, pages 9872–9883, 2018.

Xiaoyu Wang, Wenhao Yang, Chang Yao, Mingli Song, and Yuanyu Wan. Revisiting differentially private algorithms for decentralized online learning. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, pages 65213–65235, 2025.

Yibo Wang, Yuanyu Wan, Shimao Zhang, and Lijun Zhang. Distributed projection-free online learning for smooth and convex losses. In *Proceedings of the 37th AAAI Conference on Artificial Intelligence*, pages 10226–10234, 2023.

Jiawei Wangni, Jun Wang, Ji Liu, and Tong Zhang. Gradient sparsification for communication-efficient distributed optimization. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

Wei Wen, Cong Xu, Feng Yan, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Terngrad: Ternary gradients to reduce communication in distributed deep learning. *Advances in neural information processing systems*, 30, 2017.

Lin Xiao. Dual averaging method for regularized stochastic learning and online optimization. In *Advances in Neural Information Processing Systems 22*, pages 2116–2124, 2009.

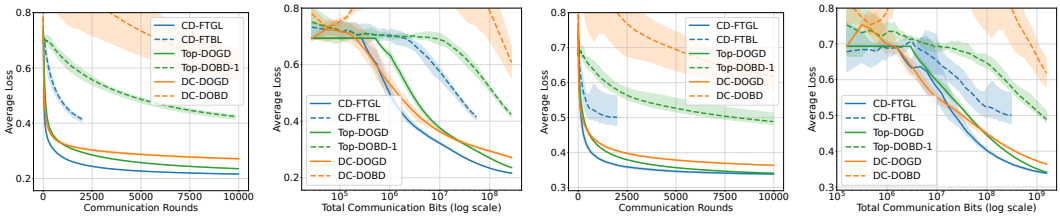
Lin Xiao and Stephen Boyd. Fast linear iterations for distributed averaging. *Systems and Control Letters*, 53(1):65–78, 2004.

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

Sifan Yang, Wenhao Yang, Wei Jiang, and Lijun Zhang. Distributed online convex optimization with efficient communication: Improved algorithm and lower bounds. *arXiv preprint arXiv:2601.04907*, 2026.

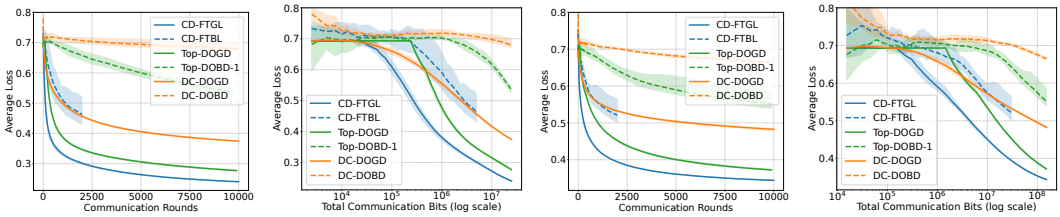
Wenpeng Zhang, Peilin Zhao, Wenwu Zhu, Steven C. H. Hoi, and Tong Zhang. Projection-free distributed online learning in networks. In *Proceedings of the 34th International Conference on Machine Learning*, pages 4054–4062, 2017.

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



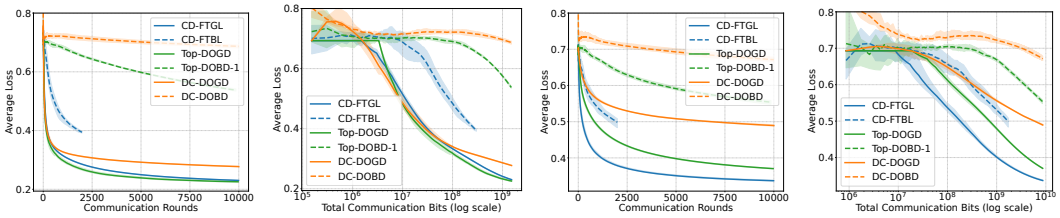
(a) ijcnn1: Loss vs Rounds (b) ijcnn1: Loss vs Bits (c) a9a: Loss vs Rounds (d) a9a: Loss vs Bits

Figure 2: Experimental results on a small random graph (9 nodes and 18 edges) with $\omega = 0.5$.



(a) ijcnn1: Loss vs Rounds (b) ijcnn1: Loss vs Bits (c) a9a: Loss vs Rounds (d) a9a: Loss vs Bits

Figure 3: Experimental results on a small random graph (9 nodes and 18 edges) with $\omega = 0.05$.



(a) ijcnn1: Loss vs Rounds (b) ijcnn1: Loss vs Bits (c) a9a: Loss vs Rounds (d) a9a: Loss vs Bits

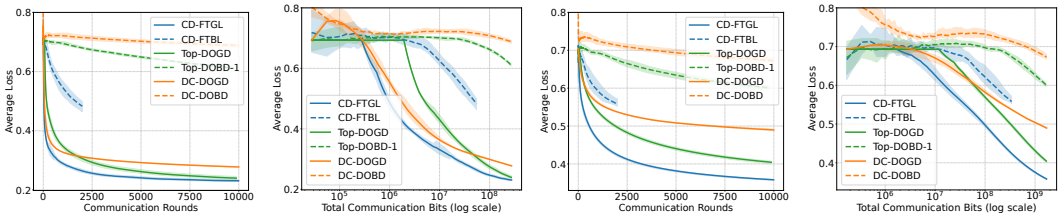
Figure 4: Experimental results on a larger random graph (50 nodes and 100 edges) with $\omega = 0.5$.

A Additional experimental results

In this section, we provide additional experimental results with different network sizes, varying compression ratios, and the sensitivity of the communication frequency parameter.

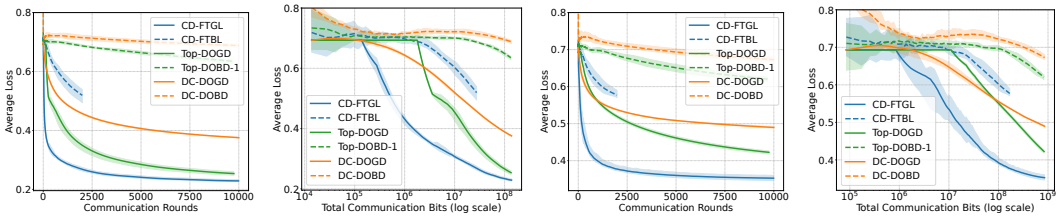
First, Figures 2 and 3 present the convergence behavior on the small random graph (9 nodes and 18 edges) under compression ratios of $\omega = 0.5$ and $\omega = 0.05$, respectively. To further investigate the scalability of our methods in larger networks, we conduct additional experiments on a larger random graph consisting of 50 nodes and 100 edges. The corresponding results for $\omega \in \{0.5, 0.1, 0.05\}$ are illustrated in Figures 4, 5 and 6. Consistent with the main findings, these additional results further validate the superior efficacy and communication efficiency of our proposed algorithms across both full-information and bandit settings, regardless of network scales and compression levels.

Furthermore, to empirically investigate the trade-off between consensus accuracy and communication efficiency, we conduct a sensitivity analysis on the communication frequency K for our CD-FTBL algorithm. Note that in the preceding experiments, the default number of communication rounds per block is set to $K = \lceil L/5 \rceil$. Specifically, we fix the block size at $L = 100$ and vary the number of communication rounds per block K. The results under compression ratios of $\omega \in \{0.5, 0.1, 0.05\}$ are presented in Figures 7, 8, and 9. As depicted in the figures, configurations with a smaller K exhibit a significantly faster decay in average loss with respect to both communication rounds and transmitted bits. However, the final loss achieved by smaller K values is slightly higher. This phenomenon highlights a clear trade-off: decreasing K aggressively reduces communication overhead while slightly compromising the final accuracy. Nevertheless, this observation supports our theoretical analysis that a reduced K is sufficient to control the consensus error within a reasonable tolerance.



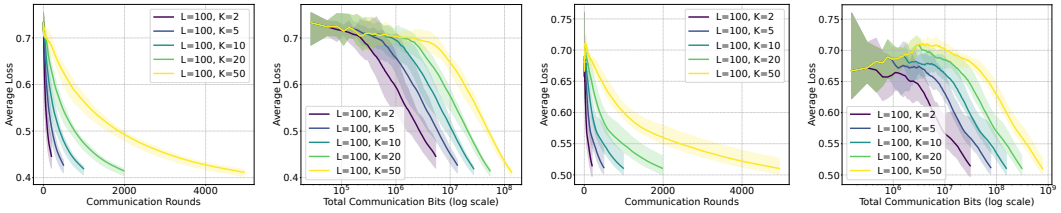
(a) ijcnn1: Loss vs Rounds (b) ijcnn1: Loss vs Bits (c) a9a: Loss vs Rounds (d) a9a: Loss vs Bits

Figure 5: Experimental results on a larger random graph (50 nodes and 100 edges) with $\omega = 0.1$.



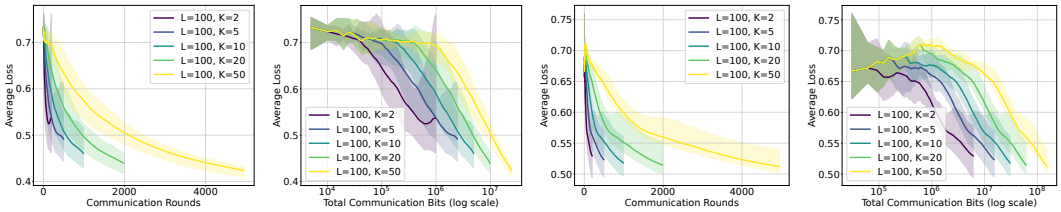
(a) ijcnn1: Loss vs Rounds (b) ijcnn1: Loss vs Bits (c) a9a: Loss vs Rounds (d) a9a: Loss vs Bits

Figure 6: Experimental results on a larger random graph (50 nodes and 100 edges) with $\omega = 0.05$.



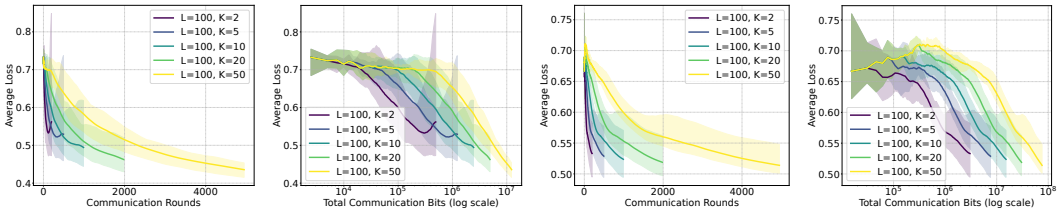
(a) ijcnn1: Loss vs Rounds (b) ijcnn1: Loss vs Bits (c) a9a: Loss vs Rounds (d) a9a: Loss vs Bits

Figure 7: Experimental results of CD-FTBL with varying communication rounds K on a small random graph (9 nodes and 18 edges) with $\omega = 0.5$ and fixed block size $L = 100$.



(a) ijcnn1: Loss vs Rounds (b) ijcnn1: Loss vs Bits (c) a9a: Loss vs Rounds (d) a9a: Loss vs Bits

Figure 8: Experimental results of CD-FTBL with varying communication rounds K on a small random graph (9 nodes and 18 edges) with $\omega = 0.1$ and fixed block size $L = 100$.



(a) ijcnn1: Loss vs Rounds (b) ijcnn1: Loss vs Bits (c) a9a: Loss vs Rounds (d) a9a: Loss vs Bits

Figure 9: Experimental results of CD-FTBL with varying communication rounds K on a small random graph (9 nodes and 18 edges) with $\omega = 0.05$ and fixed block size $L = 100$.

B Proof of Theorem 3.8

Recall that $\bar{\mathbf{d}}(z) = (1/n)\sum_{i=1}^n \mathbf{d}_i(z)$. For $z \in [T/L]$, we define the virtual leader sequence as

$$\bar{\mathbf{x}}(z) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}} \langle \mathbf{x}, \bar{\mathbf{z}}(z)\rangle + \frac{(z-1)L\alpha + 2h}{2}\|\mathbf{x}\|_2^2,$$

where $\bar{\mathbf{z}}(z) = \sum_{\tau=1}^{z-1}\bar{\mathbf{d}}(\tau)$. Following the analysis of Wan et al. [2025], the total regret can be decomposed into the regret of the virtual leader and a deviation term. To be precise, for $z \in [T/L]$, let $\ell_z(\mathbf{x}) = \langle \bar{\mathbf{d}}(z), \mathbf{x}\rangle + (L\alpha/2)\|\mathbf{x}\|_2^2$ and $\mathbf{x}^* \in \min_{\mathbf{x}\in\mathcal{K}}\sum_{t=1}^T f_t(\mathbf{x})$. According to (41) and (42) in the proof of Theorem 1 of Wan et al. [2025], we have

$$R_{T,i} \leq n\sum_{z=1}^{T/L}\left(\ell_z(\bar{\mathbf{x}}(z+1)) - \ell_z(\mathbf{x}^*)\right) + nG\sum_{z=1}^{T/L}\left(L\Delta_i(z) + 2\sum_{j=1}^n \Delta_j(z)\right), \tag{7}$$

where $\Delta_i(z) = \|\mathbf{x}_i(z) - \bar{\mathbf{x}}(z+1)\|_2$.

To analyze the regret of the virtual leader, we introduce a classical lemma.

Lemma B.1 (Lemma 6.6 in Garber and Hazan [2016]). *Let $\{\ell_t(\mathbf{x})\}_{t=1}^T$ be a sequence of convex functions and $\mathbf{x}_t^* \in \operatorname{argmin}_{\mathbf{x}\in\mathcal{K}}\sum_{\tau=1}^t \ell_\tau(\mathbf{x})$ for any $t \in [T]$. Then, it holds that $\sum_{t=1}^T \ell_t(\mathbf{x}_t^*) - \min_{\mathbf{x}\in\mathcal{K}}\sum_{t=1}^T \ell_t(\mathbf{x}) \leq 0$.*

By further defining $\ell_0(\mathbf{x}) = h\|\mathbf{x}\|_2^2$ and then applying Lemma B.1, it is easy to verify that the first term in the right side of (7) can be bounded by nhR^2. By combining this result and taking expectation over two sides of (7), we have

$$\mathbb{E}[R_{T,i}] \leq nG\sum_{z=1}^{T/L}\left(L\mathbb{E}[\Delta_i(z)] + 2\sum_{j=1}^n \mathbb{E}[\Delta_j(z)]\right) + nhR^2. \tag{8}$$

Then, we proceed to analyze $\mathbb{E}[\Delta_i(z)]$, which now is affected by compressed communication. However, similar to analysis of Wan et al. [2025], for any $z \geq 2$, we still split $\mathbb{E}[\Delta_i(z)]$ into a consensus error term and a leader stability term, i.e.,

$$\mathbb{E}[\Delta_i(z)] \leq \mathbb{E}\left[\|\mathbf{x}_i(z) - \bar{\mathbf{x}}(z-1)\|_2\right] + \mathbb{E}[\|\bar{\mathbf{x}}(z-1) - \bar{\mathbf{x}}(z+1)\|_2]. \tag{9}$$

To bound the consensus error in (9), we introduce two useful lemmas, where the latter one is derived by using Lemma 3.7.

Lemma B.2 (Lemma 5 in Duchi et al. [2011]). *Let $\Pi_{\mathcal{K}}(\mathbf{u}, \eta) = \operatorname{argmin}_{\mathbf{x}\in\mathcal{K}}\langle \mathbf{u}, \mathbf{x}\rangle + \frac{1}{\eta}\|\mathbf{x}\|_2^2$. For any $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$, we have $\|\Pi_{\mathcal{K}}(\mathbf{u}, \eta) - \Pi_{\mathcal{K}}(\mathbf{v}, \eta)\|_2 \leq \frac{\eta}{2}\|\mathbf{u} - \mathbf{v}\|_2$.*

Lemma B.3. *Suppose Assumption 3.1, 3.2 and 3.4 hold. Consider Algorithm 2 and 3 with the consensus step size γ and the communication rounds K in Theorem 3.8 and 3.12, then for all $z \in [T/L]$, it holds that*

$$\mathbb{E}\left[\|\mathbf{z}_i(z) - \bar{\mathbf{z}}(z)\|_2\right] < \frac{2\mathbb{E}[\|D(z-1)\|_F]}{\sqrt{n}},$$

where $D(z) = [\mathbf{d}_1(z), \ldots, \mathbf{d}_n(z)] \in \mathbb{R}^{d\times n}$ for any $z \geq 1$ and $D(0) = \mathbf{0}^{d\times n}$.

Under Assumptions 3.3 and 3.4, for any $z \in [T/L]$, we have

$$\|D(z)\|_F = \sqrt{\sum_{i=1}^n \|\mathbf{d}_i(z)\|_2^2} = \sqrt{\sum_{i=1}^n \left\|\sum_{t\in\mathcal{T}_z}\left(\nabla f_{t,i}(\mathbf{x}_i(t)) - \alpha\mathbf{x}_i(t)\right)\right\|_2^2} \leq \sqrt{n}L(G + \alpha R). \tag{10}$$

By combining Lemma B.3 with (10), we have

$$\mathbb{E}\left[\|\mathbf{z}_i(z) - \bar{\mathbf{z}}(z)\|_2\right] < 2L\left(G + \alpha R\right)$$

for any $z \in [T/L]$. Then, by applying Lemma B.2 and the above inequality, for any $z \geq 2$, we have

$$\mathbb{E}\left[\|\mathbf{x}_i(z) - \bar{\mathbf{x}}(z-1)\|_2\right] \leq \frac{\mathbb{E}\left[\|\mathbf{z}_i(z-1) - \bar{\mathbf{z}}(z-1)\|_2\right]}{(z-2)L\alpha + 2h} \leq \frac{2L(G + \alpha R)}{(z-2)L\alpha + 2h}. \tag{11}$$

For the leader stability in (9), according to (38) in the proof of Theorem 1 in Wan et al. [2025], for any $z \geq 2$, we have

$$\|\bar{\mathbf{x}}(z-1) - \bar{\mathbf{x}}(z+1)\|_2 \leq \frac{4L(G+2\alpha R)}{zL\alpha + 2h}. \tag{12}$$

Finally, note that $\Delta_i(1) = \|\mathbf{x}_i(1) - \bar{\mathbf{x}}(2)\|_2 \leq (2L(G+2\alpha R))/(L\alpha+2h)$, which is also due to (38) in the proof of Theorem 1 in Wan et al. [2025]. Therefore, this proof can be completed by substituting (11) and (12) into (9), and combining with (8) and the upper bound of $\Delta_i(1)$.

C Proof of Theorem 3.12

For $z \in [T/L]$, we define the virtual leader sequence as

$$\bar{\mathbf{x}}(z+1) = \operatorname*{argmin}_{\mathbf{x}\in\mathcal{K}_\epsilon} \langle \mathbf{x}, \bar{\mathbf{z}}(z)\rangle + \frac{(z-1)L\alpha + 2h}{2}\|\mathbf{x}\|_2^2$$

where $\bar{\mathbf{z}}(z) = \sum_{\tau=1}^{z-1} \bar{\mathbf{d}}(\tau)$, but $\bar{\mathbf{d}}(z)$ is defined as

$$\bar{\mathbf{d}}(z) = \frac{1}{n}\sum_{i=1}^n \mathbf{d}_i(z) = \frac{1}{n}\sum_{i=1}^n \sum_{t\in\mathcal{T}_z} \left(\mathbf{g}_i(t) - \alpha\mathbf{x}_i(z)\right).$$

Moreover, let $\tilde{\mathbf{x}}^* = (1-\epsilon/r)\mathbf{x}^* \in \mathcal{K}_\epsilon$. Following the proof of Theorems 5 and 6 in Wan et al. [2022], the total regret can be bounded as

$$\mathbb{E}[R_{T,i}] \leq \underbrace{\mathbb{E}\left[\sum_{z=1}^{T/L}\sum_{t\in\mathcal{T}_z}\sum_{j=1}^n \left(\hat{f}_{t,j,\epsilon}(\mathbf{x}_i(z)) - \hat{f}_{t,j,\epsilon}(\tilde{\mathbf{x}}^*)\right)\right]}_{:=\hat{R}_{T,i}} + 3\epsilon nGT + \frac{\epsilon nGRT}{r}, \tag{13}$$

where $\hat{f}_{t,j,\epsilon}(\cdot)$ is the ϵ-smoothed version of $f_{t,j}(\cdot)$.

Note that according to Lemma 2.6 in Hazan [2016], over the shrunk set $\mathcal{K}_\epsilon$, the ϵ-smoothed function $\hat{f}_{t,j,\epsilon}(\cdot)$ shares the same strong convexity and Lipschitz property as $f_{t,j}(\cdot)$. Therefore, following (7) and using Lemma 3.11, it is not hard to verify that

$$\hat{R}_{T,i} \leq \underbrace{n\sum_{z=1}^{T/L}\mathbb{E}\left[\ell_z(\bar{\mathbf{x}}(z+1)) - \ell_z(\tilde{\mathbf{x}}^*)\right]}_{:=\bar{R}_z} + nG\sum_{z=1}^{T/L}\left(L\mathbb{E}[\Delta_i(z)] + 2\sum_{j=1}^n \mathbb{E}[\Delta_j(z)]\right), \tag{14}$$

where $\ell_z(\mathbf{x}) = \langle \mathbf{x}, \bar{\mathbf{d}}(z)\rangle + (L\alpha/2)\|\mathbf{x}\|_2^2$ and $\Delta_i(z) = \|\mathbf{x}_i(z) - \bar{\mathbf{x}}(z+1)\|_2$.

Unlike the full-information setting, we need to introduce a new lemma to bound $\mathbb{E}[\|\bar{\mathbf{d}}(z)\|_2^2]$.

Lemma C.1 (Derived from Lemma 6 in Wan et al. [2022]). *Under Assumptions 3.3, 3.4, and 3.6, for any $i \in [n]$ and $z \in [T/L]$, let $\bar{\mathbf{d}}(z) = \frac{1}{n}\sum_{i=1}^n \mathbf{d}_i(z)$, Algorithm 3 ensures*

$$\mathbb{E}[\|\mathbf{d}_i(z)\|_2^2] \leq 2L\frac{d^2M^2}{\epsilon^2} + 2L^2G^2 + 2\alpha^2L^2R^2.$$

For brevity, let

$$B_1 := 2\frac{\sqrt{L}dM}{\epsilon} + 2LG + 3\alpha LR, \qquad B_2 := \sqrt{2L\frac{d^2M^2}{\epsilon^2} + 2L^2G^2 + 2\alpha^2L^2R^2}.$$

Due to Lemma C.1 and the definition of $\bar{\mathbf{d}}(z)$, we have

$$\mathbb{E}\left[\|\bar{\mathbf{d}}(z)\|_2^2\right] \leq \frac{1}{n}\sum_{i=1}^n \mathbb{E}[\|\mathbf{d}_i(z)\|_2^2] \leq B_2^2.$$

By combining the above inequality with Jensen's inequality, we have

$$\mathbb{E}[\|\nabla\ell_z(\mathbf{x})\|_2] = \mathbb{E}[\left\|\bar{\mathbf{d}}(z) + L\alpha\mathbf{x}\right\|_2] \leq \sqrt{\mathbb{E}\left[\|\bar{\mathbf{d}}(z)\|_2^2\right]} + L\alpha R \leq B_1.$$

Then, to bound the first term in the right side of (14), we notice that

$$\begin{aligned}\bar{R}_z \leq & n\sum_{z=1}^{T/L}\mathbb{E}[\ell_z(\bar{\mathbf{x}}(z+2))-\ell_z(\tilde{\mathbf{x}}^*)]+nB_1\sum_{z=1}^{T/L}\mathbb{E}[\|\bar{\mathbf{x}}(z+2)-\bar{\mathbf{x}}(z+1)\|_2] \\ \leq & nhR^2+nB_1\sum_{z=1}^{T/L}\mathbb{E}[\|\bar{\mathbf{x}}(z+2)-\bar{\mathbf{x}}(z+1)\|_2],\end{aligned} \tag{15}$$

where the second inequality is derived by using Lemma B.1.

Following (38) in the proof of Theorem 1 in Wan et al. [2025], for any $z \geq 1$, we have

$$\mathbb{E}\left[\|\bar{\mathbf{x}}(z+1)-\bar{\mathbf{x}}(z+2)\|_2\right] \leq \frac{2B_1}{zL\alpha+2h}. \tag{16}$$

Moreover, similar to the full-information setting, for any $z \geq 2$, we split $\mathbb{E}[\Delta_i(z)]$ into a consensus error term and a leader stability term as follows

$$\mathbb{E}[\Delta_i(z)] \leq \mathbb{E}\left[\|\mathbf{x}_i(z)-\bar{\mathbf{x}}(z)\|_2\right]+\|\bar{\mathbf{x}}(z)-\bar{\mathbf{x}}(z+1)\|_2. \tag{17}$$

By applying Lemmas B.2 and B.3, for any $z \geq 2$, we have

$$\mathbb{E}\left[\|\mathbf{x}_i(z)-\bar{\mathbf{x}}(z)\|_2\right] \leq \frac{2B_2}{(z-2)L\alpha+2h}. \tag{18}$$

By substituting (18) and (16) into (17), for any $z \geq 2$, we have

$$\mathbb{E}[\Delta_i(z)] \leq \frac{2B_2}{(z-2)L\alpha+2h}+\frac{2B_1}{(z-1)L\alpha+2h}. \tag{19}$$

Finally, this proof can be completed by substituting (16) into (15), and then combining with (13), (14), (19) and $\Delta_i(1)=\|\mathbf{x}_i(1)-\bar{\mathbf{x}}(2)\|_2=0$.

D Proof of Lemma 3.7

To facilitate the analysis, we first rewrite the error definition in matrix notation. Let $Z^k = [\mathbf{z}_1^k,\ldots,\mathbf{z}_n^k] \in \mathbb{R}^{d\times n}$ and $\hat{Z}^k = [\hat{\mathbf{z}}_1^k,\ldots,\hat{\mathbf{z}}_n^k] \in \mathbb{R}^{d\times n}$ be the concatenated matrices of local variables and their replicas at iteration k. The average is denoted by $\bar{Z}^k = Z^k\mathcal{M}$, where $\mathcal{M} = \frac{1}{n}\mathbf{1}_n\mathbf{1}_n^\top$. Consequently, the total error e^k defined in Lemma 3.7 can be equivalently written using the Frobenius norm, which is

$$e^k = \mathbb{E}[\|Z^k-\bar{Z}^k\|_F^2]+\mathbb{E}[\|Z^k-\hat{Z}^k\|_F^2].$$

The proof relies on establishing a recursive bound for these consensus and compression errors. We first state a lemma that characterizes the one-step error dynamics.

Lemma D.1. *Suppose Assumptions 3.1, 3.2, and 3.4 hold, and let $\gamma \in (0,1]$. Let $E_1(k) := \mathbb{E}_Q[\|Z^k-\bar{Z}^k\|_F^2]$ and $E_2(k) := \mathbb{E}_Q[\|Z^k-\hat{Z}^k\|_F^2]$. For any iteration $k \geq 0$, the errors satisfy:*

$$E_1(k+1) \leq \left(1-\frac{\gamma\rho}{2}\right)^2 E_1(k)+\left(\gamma+\frac{2\gamma}{\rho}\right)(1-\omega)\beta^2 E_2(k), \tag{20}$$

$$E_2(k+1) \leq \left(1+\frac{2}{\omega}\right)\gamma^2\beta^2 E_1(k)+\left(1-\frac{\omega}{2}\right)(1+8\gamma)E_2(k). \tag{21}$$

Recall the total error $e^k = E_1(k)+E_2(k)$. Summing (20) and (21), we obtain the linear recurrence, i.e.,

$$e^{k+1} \leq \eta(\gamma)E_1(k)+\xi(\gamma)E_2(k) \leq \max\{\eta(\gamma),\xi(\gamma)\}\cdot e^k,$$

where the coefficients are defined as

$$\eta(\gamma) := \left(1-\frac{\gamma\rho}{2}\right)^2+\left(1+\frac{2}{\omega}\right)\gamma^2\beta^2,$$

$$\xi(\gamma) := \left(\gamma+\frac{2\gamma}{\rho}\right)(1-\omega)\beta^2+\left(1-\frac{\omega}{2}\right)(1+8\gamma).$$

It is not hard to establish a valid upper bound for $\max\{\eta(\gamma), \xi(\gamma)\}$ by following the proof of Theorem 2 in Koloskova et al. [2019]. For completeness, we provide the detailed proof here.

First, we observe that $\eta(\gamma)$ is a convex quadratic function with respect to γ. Its minimum occurs at $\gamma' = \frac{2\rho\omega}{4(\omega+2)\beta^2+\rho^2\omega}$, with value $\eta(\gamma') = \frac{4(\omega+2)\beta^2}{4(\omega+2)\beta^2+\rho^2\omega} < 1$. Thus, by Jensen's inequality, for any $\lambda \in [0,1]$, we have

$$\eta(\lambda\gamma') \leq (1-\lambda)\eta(0) + \lambda\eta(\gamma') = 1 - \lambda(1-\eta(\gamma')) = 1 - \lambda\frac{\rho^2\omega}{4(\omega+2)\beta^2+\rho^2\omega}. \tag{22}$$

For the specific choice $\lambda' = \frac{4(\omega+2)\beta^2+\rho^2\omega}{4(1-\omega)(\rho+2)\beta^2+32\rho-16\rho\omega+2\rho^2}$, we obtain the step size $\gamma^* = \lambda'\gamma'$ as given in (2)

$$\gamma^* = \frac{\rho\omega}{2(1-\omega)(\rho+2)\beta^2+16\rho-8\rho\omega+\rho^2}.$$

Substituting this λ' back into the inequality (22), we verify that

$$\eta(\gamma^*) \leq 1 - \frac{\rho^2\omega}{4(1-\omega)(\rho+2)\beta^2+32\rho-16\rho\omega+2\rho^2} = 1 - \frac{\gamma^*\rho}{2}.$$

Similarly, substituting γ^* into $\xi(\gamma)$, it holds that

$$\xi(\gamma^*) = \left(\gamma^* + \frac{2\gamma^*}{\rho}\right)(1-\omega)\beta^2 + \left(1-\frac{\omega}{2}\right)(1+8\gamma^*) = 1 - \frac{\gamma^*\rho}{2}.$$

Thus, we conclude that $\max\{\eta(\gamma^*), \xi(\gamma^*)\} \leq 1 - \frac{\gamma^*\rho}{2}$ and the contraction factor is bounded by $\theta := 1 - \frac{\gamma^*\rho}{2}$. Applying the recursion iteratively yields

$$e^k \leq \theta e^{k-1} \leq \cdots \leq \theta^k e^0.$$

This completes the proof of Lemma 3.7.

E Proof of Lemma D.1

We analyze the evolution of the consensus and compression errors using the matrix-form updates. Since A is doubly stochastic, we have $\bar{Z}^k = Z^k\mathcal{M}$ and $\bar{Z}^k A = \bar{Z}^k$. The update rules in Algorithm 1 imply the following dynamics, i.e.,

$$\hat{Z}^{k+1} = \hat{Z}^k + Q\left(Z^k - \hat{Z}^k\right), \tag{23}$$

$$Z^{k+1} = Z^k + \gamma\hat{Z}^{k+1}(A-I) = Z^k - \gamma\hat{Z}^{k+1}\mathcal{L}. \tag{24}$$

Assumption 3.2 combined with (23) yields the contraction property of

$$\mathbb{E}_Q\left[\left\|Z^k - \hat{Z}^{k+1}\right\|_F^2\right] \leq (1-\omega)\left\|Z^k - \hat{Z}^k\right\|_F^2. \tag{25}$$

Part 1: Consensus error analysis. Recall that $\bar{Z}^k\mathcal{L} = 0$ and $\mathcal{M}\mathcal{L} = 0$. Multiplying (24) by $(I-\mathcal{M})$, we isolate the consensus deviation by

$$\begin{aligned} Z^{k+1} - \bar{Z}^{k+1} &= \left(Z^k - \gamma\hat{Z}^{k+1}\mathcal{L}\right)(I-\mathcal{M}) \\ &= (Z^k - \bar{Z}^k)(I-\gamma\mathcal{L}) - \gamma(\hat{Z}^{k+1} - Z^k)\mathcal{L}. \end{aligned}$$

Using Young's inequality $\|X+Y\|_F^2 \leq (1+\alpha)\|X\|_F^2 + (1+\alpha^{-1})\|Y\|_F^2$ with $\alpha = \frac{\gamma\rho}{2}$, we have

$$\|Z^{k+1} - \bar{Z}^{k+1}\|_F^2 \leq \left(1+\frac{\gamma\rho}{2}\right)\underbrace{\|(Z^k - \bar{Z}^k)(I-\gamma\mathcal{L})\|_F^2}_{T_1} + \left(1+\frac{2}{\gamma\rho}\right)\underbrace{\|\gamma(\hat{Z}^{k+1} - Z^k)\mathcal{L}\|_F^2}_{T_2}. \tag{26}$$

For T_1, utilizing the property $(Z^k - \bar{Z}^k)\mathcal{M} = 0$ and the spectral bound $\|I - \gamma\mathcal{L} - \mathcal{M}\|_2 \leq 1 - \gamma\rho$ for $\gamma\rho \leq 1$, we obtain that

$$T_1 = \|(Z^k - \bar{Z}^k)(I - \gamma\mathcal{L} - \mathcal{M})\|_F^2 \leq (1-\gamma\rho)^2\|Z^k - \bar{Z}^k\|_F^2.$$

For T_2, using $\|\mathcal{L}\|_2 = \beta$, we have $\mathbb{E}_Q[T_2] \leq \gamma^2\beta^2(1-\omega)\mathbb{E}_Q[\|Z^k - \hat{Z}^k\|_F^2]$. Let $E_1(k) := \mathbb{E}_Q[\|Z^k - \bar{Z}^k\|_F^2]$ and $E_2(k) := \mathbb{E}_Q[\|Z^k - \hat{Z}^k\|_F^2]$. Substituting these bounds into (26) and observing that $(1+\frac{\gamma\rho}{2})(1-\gamma\rho)^2 \leq (1-\frac{\gamma\rho}{2})^2$ for $\gamma\rho \leq 1$, we derive that

$$E_1(k+1) \leq \left(1 - \frac{\gamma\rho}{2}\right)^2 E_1(k) + \left(\gamma + \frac{2\gamma}{\rho}\right)(1-\omega)\beta^2 E_2(k).$$

Part 2: Compression error analysis. Rearranging (24) allows us to express the compression update as

$$Z^{k+1} - \hat{Z}^{k+1} = Z^k - \gamma\hat{Z}^{k+1}\mathcal{L} - \hat{Z}^{k+1} = (Z^k - \hat{Z}^{k+1})(I + \gamma\mathcal{L}) - \gamma(Z^k - \bar{Z}^k)\mathcal{L}.$$

By using Young's inequality again with $\alpha = \frac{\omega}{2}$, we have

$$\|Z^{k+1} - \hat{Z}^{k+1}\|_F^2 \leq \left(1 + \frac{\omega}{2}\right)\|(Z^k - \hat{Z}^{k+1})(I+\gamma\mathcal{L})\|_F^2 + \left(1 + \frac{2}{\omega}\right)\|\gamma(Z^k - \bar{Z}^k)\mathcal{L}\|_F^2.$$

By applying (25) and the above inequality, we obtain

$$E_2(k+1) \leq \left(1 + \frac{\omega}{2}\right)\|I + \gamma\mathcal{L}\|_2^2(1-\omega)E_2(k) + \left(1 + \frac{2}{\omega}\right)\gamma^2\beta^2 E_1(k).$$

Using $\|I + \gamma\mathcal{L}\|_2 \leq 1 + \gamma\beta$ and the inequality $(1+\frac{\omega}{2})(1-\omega)(1+\gamma\beta)^2 \leq (1-\frac{\omega}{2})(1+8\gamma)$ for $\beta \leq 2, \gamma \leq 1$, we conclude

$$E_2(k+1) \leq \left(1 + \frac{2}{\omega}\right)\gamma^2\beta^2 E_1(k) + \left(1 - \frac{\omega}{2}\right)(1+8\gamma)E_2(k).$$

This completes the proof of Lemma D.1.

F Proof of Lemma B.3

The proof relies on the recursive error bound established in Lemma 3.7. We proceed the proof in three steps: analyzing the error propagation, determining the number of iterations K, and deriving the final bound.

Step 1: Error propagation analysis. At the beginning of block z (i.e., $k = 0$), the states are initialized based on the output of the previous block and the gradient update

$$\mathbf{z}_i^0(z) = \mathbf{z}_i^K(z-1) + \mathbf{d}_i(z-1), \quad \hat{\mathbf{z}}_i^0(z) = \hat{\mathbf{z}}_i^K(z-1).$$

Consequently, the initial error e_z^0 is bounded by the error at the end of the previous block, e_{z-1}^K. Applying the inequality $\|a+b\|_2^2 \leq 2\|a\|_2^2 + 2\|b\|_2^2$ and the variance decomposition, we derive

$$\begin{aligned} e_z^0 &= \mathbb{E}\sum_{i=1}^n \left(\|\mathbf{z}_i^0(z) - \bar{\mathbf{z}}^0(z)\|_2^2 + \|\mathbf{z}_i^0(z) - \hat{\mathbf{z}}_i^0(z)\|_2^2\right) \\ &\leq 2e_{z-1}^K + 2\sum_{i=1}^n \left(\|\mathbf{d}_i(z-1) - \bar{\mathbf{d}}(z-1)\|_2^2 + \|\mathbf{d}_i(z-1)\|_2^2\right) \\ &\leq 2e_{z-1}^K + 4\|D(z-1)\|_F^2. \end{aligned} \tag{27}$$

Now, we invoke Lemma 3.7. By setting the step size γ as specified, the algorithm ensures the contraction property $e_z^K \leq \theta e_z^0$, where the contraction factor is defined as $\theta := (1 - \frac{\gamma\rho}{2})^K$. Let $C_z := 4\|D(z-1)\|_F^2$. Combining this with (27), we obtain the recursion

$$e_z^K \leq \theta e_z^0 \leq 2\theta e_{z-1}^K + \theta C_z.$$

Denoting $e_z := e_z^K$ and noting the initialization $e_1 = 0$ (since $\mathbf{z}_i(1) = \mathbf{0}$), we solve this recursion for $z \geq 2$. Assuming $2\theta < 1$, summing the geometric series yields

$$e_z \leq \theta C_z \sum_{j=0}^{z-2}(2\theta)^j < \frac{\theta C_z}{1-2\theta}. \tag{28}$$

Step 2: Choosing number of iterations K. Next, we determine the condition on K to ensure the error coefficient scales with $1/\sqrt{n}$. Specifically, we require

$$\sqrt{\frac{\theta}{1-2\theta}} \leq \frac{1}{\sqrt{n}} \iff \theta \leq \frac{1}{n+2}.$$

Using the inequality $\ln(1-x) \leq -x$ for $x \in (0,1)$, it suffices to choose K such that

$$K \geq \frac{\ln(n+2)}{-\ln(1-\frac{\gamma\rho}{2})} \geq \frac{2\ln(n+2)}{\gamma\rho}.$$

Under this condition, we have $\theta \leq 1/(n+2) < 1/2$, which validates the convergence of the geometric series in (28).

Step 3: Final bound. Combining the results from Step 1 and Step 2, we obtain the final bound on the consensus error

$$\mathbb{E}\left[\|\mathbf{z}_i(z) - \bar{\mathbf{z}}(z)\|_2\right] \leq \sqrt{e_z} < \sqrt{\frac{\theta}{1-2\theta}}\sqrt{C_z} \leq \frac{1}{\sqrt{n}} \cdot 2\|D(z-1)\|_F.$$

This concludes the proof of Lemma B.3.

G Proof of Corollaries 3.9 and 3.10

Convex case. In the following, we first consider the case with convex losses, by substituting $\alpha = 0$, $h = 3\sqrt{LT}G/R$ and $L = \lceil 2\ln(n+2)/\gamma\rho \rceil$ into (4), we have

$$\begin{aligned}
\mathbb{E}[R_{T,i}] &\leq 3nLG\left(\sum_{z=2}^{T/L}\frac{LG}{h} + \sum_{z=1}^{T/L}\frac{2LG}{h}\right) + nhR^2 \leq \frac{9nLG^2T}{h} + nhR^2 \\
&\leq 6nGR\sqrt{LT} = O\left(n\sqrt{\log n}\sqrt{T}\cdot(\gamma\rho)^{-1/2}\right) = O\left(n\sqrt{\log n}\cdot\rho^{-1}\omega^{-1/2}\cdot\sqrt{T}\right).
\end{aligned}$$

When $\omega = 1$, we have $(\gamma\rho)^{-1/2} = \sqrt{(8+\rho)/\rho} = O(\rho^{-1/2})$. Thus, the regret bound can be improved to

$$\mathbb{E}[R_{T,i}] = O\left(n\sqrt{\log n}\cdot\rho^{-1/2}\cdot\sqrt{T}\right).$$

Strongly convex case. We continue to consider the case with the strongly convex losses, by substituting $h = \alpha L$, $L = \lceil 2\ln(n+2)/\gamma\rho \rceil$ into (4), we have

$$\begin{aligned}
\mathbb{E}[R_{T,i}] &\leq n\alpha LR^2 + \frac{6nLG}{\alpha}\left((G+\alpha R)\ln(T/L) + 2(G+2\alpha R)\ln(T/L)\right) \\
&= n\alpha LR^2 + \frac{6nLG\,(3G+5\alpha R)\ln(T/L)}{\alpha} \\
&= O\left(n\log n\cdot\gamma^{-1}\rho^{-1}\cdot\log T\right) = O\left(n\log n\cdot\rho^{-2}\omega^{-1}\cdot\log T\right).
\end{aligned}$$

When $\omega = 1$, we have $(\gamma\rho)^{-1/2} = \sqrt{(8+\rho)/\rho} = O(\rho^{-1/2})$. Thus, the regret bound can be improved to

$$\mathbb{E}[R_{T,i}] = O\left(n\log n\cdot\rho^{-1}\cdot\log T\right).$$

H Proof of Corollaries 3.13 and 3.14

Convex case. In the following, we first consider the case with convex losses, in which the parameters of our Algorithm 3 are set to $\alpha = 0$, $L = \max\{K, \sqrt{T}\}$, $K = \lceil 2\ln(n+2)/\gamma\rho \rceil$, $h = \sqrt{dLT}M/R$ and $\epsilon = c\,d^{1/2}T^{-1/4}$. Because of $\alpha = 0$ we have

$$\begin{aligned}
\mathbb{E}[R_{T,i}] \leq\ & 6nLG\sum_{z=2}^{T/L}\frac{\sqrt{2L\frac{d^2M^2}{\epsilon^2} + 2L^2G^2}}{2h} + 6nLG\sum_{z=1}^{T/L}\frac{2\frac{\sqrt{L}dM}{\epsilon} + 2LG}{2h} \\
&+ 2n\sum_{z=1}^{T/L}\frac{\left(2\frac{\sqrt{L}dM}{\epsilon} + 2LG\right)^2}{2h} + nhR^2 + 3\epsilon nGT + \frac{\epsilon nGRT}{r}.
\end{aligned} \tag{29}$$

Because $\epsilon = c\, d^{1/2} T^{-1/4}$ and $h = \sqrt{dLT}M/R$, we can bound the first term on the R.H.S of (29) by

$$6nLG \sum_{z=2}^{T/L} \frac{\sqrt{2L\frac{d^2M^2}{\epsilon^2} + 2L^2G^2}}{2h} \leq 6nLG \cdot \frac{T}{L}\left(\frac{\sqrt{L}dM}{h\epsilon} + \frac{LG}{h}\right) = O(nT^{3/4}) + O(n\sqrt{LT}). \tag{30}$$

Similarly, the second term can be bounded by

$$6nLG \sum_{z=1}^{T/L} \frac{2\frac{\sqrt{L}dM}{\epsilon} + 2LG}{2h} = 6nLG \cdot \frac{T}{L}\left(\frac{\sqrt{L}dM}{h\epsilon} + \frac{LG}{h}\right) = O(nT^{3/4}) + O(n\sqrt{LT}). \tag{31}$$

And the third term can be bounded by

$$2n \sum_{z=1}^{T/L} \frac{\left(2\frac{\sqrt{L}dM}{\epsilon} + 2LG\right)^2}{2h} \leq 4n \cdot \frac{T}{L}\left(\frac{Ld^2M^2}{h\epsilon^2} + \frac{L^2G^2}{h}\right) = O(nT^{3/4}) + O(n\sqrt{LT}). \tag{32}$$

Moreover, the remaining terms on the R.H.S of (29) have the bound of

$$\begin{aligned} &nhR^2 + \left(3 + \frac{R}{r}\right)\epsilon nGT = nM\sqrt{dLT}R + \left(3 + \frac{R}{r}\right) c d^{1/2} nGT^{3/4} \\ &= O(n\sqrt{LT}) + O(nT^{3/4}). \end{aligned} \tag{33}$$

Finally, by combining (29), (30), (31), (32) and (33), we can obtain

$$\mathbb{E}[R_{T,i}] = O(nT^{3/4}) + O(n\sqrt{LT}) \leq O(nT^{3/4} + n\sqrt{KT}).$$

Strongly convex case. We continue to consider the case with the strongly convex losses, in which the parameters of our Algorithm 3 are set to $\alpha > 0$, $L = \max\{K, T^{2/3}(\ln T)^{-2/3}\}$, $K = \lceil 2\ln(n+2)/\gamma\rho \rceil$, $\epsilon = c\, d^{2/3} T^{-1/3} (\ln T)^{1/3}$ and $h = \alpha L$.
Because $\alpha > 0$, $\epsilon = c\, d^{2/3} T^{-1/3} (\ln T)^{1/3}$ and $h = \alpha L$, we can bound the first term on the R.H.S of (29) by

$$\begin{aligned} &6nLG \sum_{z=2}^{T/L} \frac{\sqrt{2L\frac{d^2M^2}{\epsilon^2} + 2L^2G^2 + 2\alpha^2L^2R^2}}{(z-2)L\alpha + 2h} = 6nLG \sum_{z=2}^{T/L} \frac{\sqrt{2L\frac{d^2M^2}{\epsilon^2} + 2L^2G^2 + 2\alpha^2L^2R^2}}{\alpha L z} \\ &\leq 6nG(1 + \ln(T/L))\left(\frac{2\sqrt{L}dM}{\alpha\epsilon} + \frac{2L\sqrt{G^2 + \alpha^2R^2}}{\alpha}\right) \\ &= O\left(n\sqrt{L}\, T^{1/3} (\log T)^{2/3}\right) + O\left(nL \log T\right). \end{aligned} \tag{34}$$

Then, we can bound the second term by

$$\begin{aligned} &6nLG \sum_{z=1}^{T/L} \frac{2\frac{\sqrt{L}dM}{\epsilon} + 2LG + 3\alpha LR}{(z-1)L\alpha + 2h} = 6nLG \sum_{z=1}^{T/L} \frac{2\frac{\sqrt{L}dM}{\epsilon} + 2LG + 3\alpha LR}{\alpha L(z+1)} \\ &\leq 6nG(1 + \ln T/L)\left(\frac{2\sqrt{L}dM}{\alpha\epsilon} + \frac{L(2G + 3\alpha R)}{\alpha}\right) \\ &= O\left(n\sqrt{L}T^{1/3} (\log T)^{2/3}\right) + O\left(nL \log T\right). \end{aligned} \tag{35}$$

Similarly, the third term can be bounded by

$$\begin{aligned} &2n \sum_{z=1}^{T/L} \frac{\left(2\frac{\sqrt{L}dM}{\epsilon} + 2LG + 3\alpha LR\right)^2}{zL\alpha + 2h} = 2n \sum_{z=1}^{T/L} \frac{\left(2\frac{\sqrt{L}dM}{\epsilon} + 2LG + 3\alpha LR\right)^2}{\alpha L(z+2)} \\ &\leq 4n(1 + \ln(T/L))\left(\frac{4d^2M^2}{\alpha\epsilon^2} + \frac{L(2G + 3\alpha R)^2}{\alpha}\right) = O\left(nT^{2/3} (\log T)^{1/3}\right) + O\left(nL \log T\right). \end{aligned} \tag{36}$$

The remaining terms on the R.H.S of (29) can be bounded by

$$\begin{aligned} &nhR^2 + 3\epsilon nGT + \frac{\epsilon nGRT}{r} = n\alpha LR^2 + \left(3 + \frac{R}{r}\right) cnGd^{2/3}T^{2/3}(\ln T)^{1/3} \\ &= O(nT^{2/3}(\log T)^{1/3}). \end{aligned} \tag{37}$$

Finally, by combining (6), (34), (35), (36) and (37), we can obtain

$$\mathbb{E}[R_{T,i}] = O\left(nT^{2/3}(\log T)^{1/3}\right) + O\left(nL\log T\right) \leq O\left(nT^{2/3}(\log T)^{1/3} + nK\log T\right).$$

I Compatibility with both biased and unbiased compressors

At first glance, Assumption 3.2 might seem designed specifically for biased compressors. However, it is actually a general framework that can easily capture both biased and unbiased compressors. We explain this below.

Biased compressors. Compressors like Top-k sparsification are inherently biased ($\mathbb{E}[\mathcal{Q}(x)] \neq x$). They directly fit Assumption 3.2 because they are defined by bounding the compression error. For example, the Top-k operator preserves the k components with the largest magnitudes. Since the discarded $d-k$ components are the smallest, their sum of squares is at most $\frac{d-k}{d}\|x\|_2^2$. This guarantees $\|\mathcal{Q}(x) - x\|_2^2 \leq (1 - \frac{k}{d})\|x\|_2^2$. Therefore, it naturally satisfies Assumption 3.2 with $\omega = k/d$.

Unbiased compressors. Unbiased compressors, such as QSGD or Rand-k, are usually defined by a bounded variance as follow

$$\begin{aligned} \mathbb{E}[\mathcal{C}(x)] &= x, \\ \mathbb{E}[\|\mathcal{C}(x)\|_2^2] &\leq \zeta\|x\|_2^2, \end{aligned}$$

where $\zeta \geq 1$. Directly calculating the compression error gives $\mathbb{E}[\|\mathcal{C}(x) - x\|_2^2] \leq (\zeta - 1)\|x\|_2^2$. Since $\zeta - 1$ can be greater than 1, this does not immediately match the $(1-\omega)\|x\|_2^2$ format in Assumption 3.2.

However, we can make unbiased compressor fit our framework perfectly by applying a simple scaling factor [Beznosikov et al., 2020]. We define a scaled compressor $\mathcal{Q}'(x) = \frac{1}{\zeta}\mathcal{C}(x)$. By calculating its expected squared error, we have

$$\begin{aligned} \mathbb{E}[\|\mathcal{Q}'(x) - x\|_2^2] &= \mathbb{E}\left[\left\|\frac{1}{\zeta}\mathcal{C}(x) - x\right\|_2^2\right] \\ &= \frac{1}{\zeta^2}\mathbb{E}[\|\mathcal{C}(x)\|_2^2] - \frac{2}{\zeta}\|x\|_2^2 + \|x\|_2^2 \\ &\leq \left(\frac{1}{\zeta} + 1 - \frac{2}{\zeta}\right)\|x\|_2^2 = \left(1 - \frac{1}{\zeta}\right)\|x\|_2^2. \end{aligned}$$

This shows that the scaled compressor $\mathcal{Q}'(x)$ perfectly satisfies Assumption 3.2 with a compression ratio of $\omega = 1/\zeta$. Therefore, all the theoretical results in this paper apply to unbiased compressors simply by replacing ω with $1/\zeta$.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The claims match the theoretical bounds in Section 3 and empirical results in Section 4.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discussed the open problems and limitations in Section 5.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: All assumptions are explicitly stated in Section 3.1, and the complete proofs are provided in the Appendix.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide all necessary dataset details, hyperparameters, and network topologies in Section 4.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Our source code is publicly available at https://github.com/SSSS88888/CD-FTGL.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://neurips.cc/public/guides/CodeSubmissionPolicy) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

Answer: [Yes]

Justification: We provide all necessary dataset details, hyperparameters, and network topologies in Section 4.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: As stated in Section 4, all experiments are run 5 times, and figures show the mean performance with shaded regions indicating the min-max range.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All experiments were conducted on a workstation running Windows 11 with an Intel Core i7-14700 CPU and 16 GB RAM, without GPU acceleration.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research fully conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: This work focuses on foundational theoretical optimization algorithms. We do not foresee any direct or immediate negative societal impacts arising from our theoretical results.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.

- If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper does not release any models or scraped datasets that have a high risk for misuse.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We cited the original sources for the LIBSVM datasets and NetworkX library used in Section 4.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.

- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: This paper relies on existing open-source datasets (LIBSVM) and does not introduce any new datasets or similar assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: This research does not involve crowdsourcing or human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: Not applicable, as this research does not involve human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.