NON-SMOOTH NON-CONVEX STOCHASTIC OPTIMIZATION: SIGN-BASED METHODS AND BEYOND

Anonymous authors
Paper under double-blind review

ABSTRACT

Gradient compression algorithms, such as sign-based methods, are widely used to reduce the computational and communication overhead of large-scale optimization. However, existing studies rely on smoothness or convexity assumptions, which are frequently violated in real-world applications. To address this limitation, we *initiate* the study of non-smooth non-convex stochastic optimization with compressed gradients. Let d denote the dimensionality. First, we develop a sign-based online method and incorporate it into the online-to-non-convex (O2NC) conversion, which yields a query complexity of $O(d\epsilon^{-3}\delta^{-1})$. Second, we consider the zeroth-order optimization setting, where only noisy function values are accessible. In this regime, our method attains an $O(d\epsilon^{-3}\delta^{-1})$ query complexity, which *simultaneously* matches that of the first-order compressed setting and the optimal $O(d\epsilon^{-3}\delta^{-1})$ complexity of zeroth-order optimization without compression. The primary idea is to design a *compression-estimation decoupling strategy*, which constructs the gradient estimator with multiple samples and recursively applies the sign compressor over several rounds, thereby controlling the estimation and compression errors at the same time. Finally, we extend our methods to general compressors and to distributed stochastic optimization with bidirectional compression in both the first-order and zeroth-order settings.

1 INTRODUCTION

In this paper, we investigate the stochastic optimization problem

$$\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}), \tag{1}$$

where $f(\cdot):\mathbb{R}^d\to\mathbb{R}$ is the objective function. We assume access only to a stochastic oracle that, given a query point $\mathbf{x}$, returns a noisy gradient $\nabla f(\mathbf{x};\xi)$ satisfying $\mathbb{E}[\nabla f(\mathbf{x};\xi)]=\nabla f(\mathbf{x})$, where ξ denotes a random sample drawn from a stochastic oracle.

In recent years, gradient compression algorithms have attracted significant research attention (Seide et al., 2014; Bernstein et al., 2018; 2019; Karimireddy et al., 2019; Richtárik et al., 2021; Huang et al., 2022; Yu et al., 2026), among which SignSGD has emerged as a prominent representative. By only employing the sign of the gradient for updates, SignSGD substantially reduces communication overhead and is well suited to large-scale training. Despite this advantage, vanilla SignSGD fails to converge even for convex loss functions. To address this issue, Seide et al. (2014) propose the error feedback (EF) mechanism, which records the past compression error and adds it back to subsequent updates. Building upon EF, Karimireddy et al. (2019) develop EF-SignSGD with a convergence rate by introducing a scaled sign operator, which rescales the sign of the information by its ℓ_1-norm. Although sign-based methods are widely applied across various scenarios, existing studies predominantly rely on smoothness or convexity assumptions on the loss, which are frequently violated in practice. A prominent example is the use of the ReLU activation function in neural networks, which renders the loss non-smooth (Zhang et al., 2020). A parallel line of research focuses on non-smooth non-convex optimization without compression (Zhang et al., 2020; Lin et al., 2022; Cutkosky et al., 2023; Jordan et al., 2023; Zhang & Cutkosky, 2024; Kornowski & Shamir, 2024). In this setting, (δ,ϵ)-stationarity serves as the standard performance measure, and the optimal query complexity for finding (δ,ϵ)-stationary points is $O(\epsilon^{-3}\delta^{-1})$, attained by the online-to-non-convex (O2NC) conversion (Cutkosky et al., 2023). Specifically, O2NC reduces the non-smooth non-convex problem to an

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

online convex optimization (OCO) task over a restricted domain, where the goal is to minimize the regret of an online learner (Hazan, 2016). By leveraging the online gradient descent (OGD) method (Zinkevich, 2003), O2NC achieves the optimal query complexity. Thus, a natural question arises *whether it is possible to develop a sign-based method for non-smooth non-convex optimization*.

In this paper, we give an affirmative answer to this question. A natural way to devise a sign-based algorithm for non-smooth non-convex optimization is to integrate the EF mechanism into the O2NC conversion. However, this combination encounters a fundamental obstacle, which arises from the projection onto the feasible domain in each OGD update of O2NC. The error induced by this step couples with the compression error across iterations and therefore accumulates without bound. To overcome this issue, we first develop a sign-based online method, termed Compressed Follow-the-Regularized-Leader (C-FTRL), by adapting EF to the Follow-the-Regularized-Leader (FTRL) framework (Hazan, 2016). This design confines the projection error to the current round and thereby prevents its accumulation with the compression error. Let d denote the dimensionality. C-FTRL achieves an $O(\sqrt{dT})$ regret bound, preserving the optimal dependence on T. We then employ C-FTRL as the online learner within O2NC, which gives rise to the Compressed O2NC (C-O2NC) algorithm. C-O2NC finds (δ, ϵ)-stationary points within $O(d\epsilon^{-3}\delta^{-1})$ gradient queries, which is optimal with respect to ϵ and δ (Cutkosky et al., 2023).

We further consider zeroth-order optimization (Shamir, 2017), where only stochastic function values are accessible. A natural approach is to combine the classical zeroth-order gradient estimator (Agarwal et al., 2010; Shahrampour & Jadbabaie, 2017) with C-O2NC. However, this combination yields a complexity that suffers from suboptimal dependence on dimensionality. The reason is that both the estimation error and the compression error depend on d, and the coupling between them further amplifies this dependence. To overcome this limitation, we propose a *compression-estimation decoupling strategy*, which *disentangles* the joint effect of the two errors and controls them independently. Specifically, we draw R samples to construct a gradient estimator with reduced variance, and *recursively* apply the compressor over R rounds to suppress the compression error. With $R = d$, both errors become independent of d, which yields an $O(d\epsilon^{-3}\delta^{-1})$ query complexity. Remarkably, this complexity *simultaneously* matches that of C-O2NC in the first-order setting and the optimal theoretical guarantee for non-smooth non-convex optimization in the zeroth-order setting without gradient compression (Kornowski & Shamir, 2024).

Our methods exhibit broad versatility along two complementary directions. On one hand, the analysis extends beyond the sign compressor to general compressors with a quality factor $\omega \in (0, 1]$, yielding complexities of $O(\omega^{-1}\epsilon^{-3}\delta^{-1})$ and $O(\max\{\omega^{-1}, d\}\epsilon^{-3}\delta^{-1})$ in the first-order and zeroth-order settings, respectively. On the other hand, we investigate the distributed setting, in which n local workers collaborate with a central server to minimize a global objective (Nedić & Ozdaglar, 2009). For this setting, we develop a distributed variant of the O2NC conversion and further leverage the optimal distributed OCO method with *bidirectional compression* (Yang et al., 2026) as its online learner. To handle the randomness in O2NC which otherwise causes inconsistent decisions across workers, we synchronize the sampling via a shared random seed. Our methods attain query complexities of $O(\omega^{-1}\epsilon^{-3}\delta^{-1})$ and $O(\max\{\omega^{-1}, d\}\epsilon^{-3}\delta^{-1})$ in the first-order and zeroth-order regimes, while substantially reducing the communication cost between the workers and the server. To the best of our knowledge, these results constitute the first theoretical guarantees for gradient compression in non-smooth non-convex optimization. Experimental results in Appendix D further corroborate the effectiveness and efficiency of our methods.

2 Related Work

To reduce communication overhead, substantial effort has been devoted to integrating compression into optimization algorithms (Seide et al., 2014; Bernstein et al., 2018; Stich et al., 2018; Bernstein et al., 2019; Karimireddy et al., 2019; Richtárik et al., 2021; Huang et al., 2022; Richtárik et al., 2024; Khirirat et al., 2025; Gao et al., 2025; Islamov et al., 2025). These methods employ a compression operator $\mathcal{C}(\cdot) : \mathbb{R}^d \to \mathbb{R}^d$ and transmit $\mathcal{C}(\mathbf{x})$ in place of $\mathbf{x}$. Such operators fall into two major families, namely: *contractive compressors*, satisfying $\mathbb{E}[\|\mathcal{C}(\mathbf{x}) - \mathbf{x}\|^2] \leq (1-\omega)\|\mathbf{x}\|^2$ for some $\omega \in (0, 1]$, and *unbiased compressors*, satisfying $\mathbb{E}[\mathcal{C}(\mathbf{x})] = \mathbf{x}$ together with a bounded variance condition. While unbiased compressors are more amenable to theoretical analysis, contractive compressors typically achieve superior empirical performance (Huang et al., 2022).

Seide et al. (2014) first propose transmitting only the sign of the gradient and introduce the error feedback (EF) mechanism to compensate for the bias incurred by the sign operator. Subsequently, Bernstein et al. (2018) establish convergence guarantees for SignSGD under smooth non-convex losses with large batch sizes. In parallel, Wu et al. (2018) analyze SGD with EF for quadratic functions. Karimireddy et al. (2019) further use EF to develop EF-SignSGD and provide a convergence analysis, albeit confined to the single-node setting. The later work (Beznosikov et al., 2023) extends EF to the distributed setting. Richtárik et al. (2021) propose EF21 in the deterministic regime, which applies a compressor to the gradient residual under a unidirectional communication scheme. Huang et al. (2022) investigate gradient compression in the smooth non-convex setting. They design the fast compressed communication (FCC) technique to suppress compression error, and leverage multiple samples to reduce stochastic gradient variance, thereby attaining the nearly optimal rate for distributed smooth non-convex optimization with bidirectional compression. More recently, Fatkhullin et al. (2025) extend EF21 to distributed stochastic optimization with bidirectional compression. Beyond smooth objectives, Islamov et al. (2025) first study gradient compression methods for non-smooth convex losses under safety constraints. They point out that EF21 fails to converge for non-smooth objectives and adopt the classical EF mechanism.

Although extensive efforts have been devoted to communication-efficient optimization, the existing theoretical guarantees predominantly rely on smoothness or convexity assumptions, which are frequently violated in modern machine learning applications such as deep learning with ReLU activations (Zhang et al., 2020). Consequently, despite its practical importance, non-smooth non-convex optimization with compressed gradients remains largely unexplored. A theory that bridges this gap is essential not only for broadening the foundations of gradient compression but also for enabling its principled deployment in large-scale learning systems.

We defer further discussion of non-smooth non-convex optimization to Appendix A.

3 Technical Background

In this section, we first introduce the necessary notation and formal assumptions, and then outline the key techniques employed in this work.

Notation. We use bold lower-case letters to denote vectors in $\mathbb{R}^d$ and $\|\mathbf{x}\|$ to denote the ℓ_2-norm of $\mathbf{x}$. The Euclidean ball centered at $\mathbf{x}$ with radius δ is denoted by $\mathbb{B}(\mathbf{x}, \delta) = \{\mathbf{y} \in \mathbb{R}^d \mid \|\mathbf{x} - \mathbf{y}\| \leq \delta\}$.

We introduce the following standard assumptions on the loss function (Cutkosky et al., 2023).

Assumption 1 The loss $f(\cdot)$ is Lipschitz continuous and lower bounded by f^*, i.e., $f^* = \inf_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) > -\infty$. We define the initial gap $\Delta = f(\mathbf{x}_0) - f^*$, where $\mathbf{x}_0$ is the initial point.

We also adopt the following assumption on the stochastic oracle, which is widely used in prior work (Zhang et al., 2020; Lin et al., 2022; Cutkosky et al., 2023; Liu, 2026).

Assumption 2 Given a function $f(\cdot)$, a point $\mathbf{x} \in \mathbb{R}^d$, and a sample ξ, the stochastic oracle returns the stochastic function value $f(\mathbf{x};\xi)$ and the stochastic gradient $\nabla f(\mathbf{x};\xi)$, which satisfy $\mathbb{E}[f(\mathbf{x};\xi)] = f(\mathbf{x})$ and $\mathbb{E}[\nabla f(\mathbf{x};\xi)] = \nabla f(\mathbf{x})$. The function $f(\cdot;\xi)$ is $L(\xi)$-Lipschitz continuous and $\mathbb{E}\left[L(\xi)^2\right] \leq L_0^2$. Moreover, there exists a constant $G \geq 0$ such that $\mathbb{E}\left[\|\nabla f(\mathbf{x};\xi)\|^2\right] \leq G^2$.

According to Rademacher's theorem (Evans & Gariepy, 2015), any Lipschitz continuous function is differentiable almost everywhere. Thus, the oracle is well-defined almost everywhere. We next recall the notion of the Goldstein subdifferential (Goldstein, 1977).

Definition 1 For a Lipschitz continuous function $f(\cdot) : \mathbb{R}^d \to \mathbb{R}$ and any $\delta > 0$, the Goldstein δ-subdifferential of $f(\cdot)$ at a point $\mathbf{x}$ is $\partial_\delta f(\mathbf{x}) = \mathrm{conv}\{\bigcup_{\mathbf{y}\in\mathbb{B}(\mathbf{x},\delta)} \partial f(\mathbf{y})\}$, where $\partial f(\mathbf{y})$ denotes the Clarke subdifferential (Clarke et al., 1998), given by $\partial f(\mathbf{y}) = \mathrm{conv}\{\mathbf{g} : \mathbf{g} = \lim_{\mathbf{y}_k\to\mathbf{y}} \nabla f(\mathbf{y}_k)\}$.

To evaluate convergence in the non-smooth non-convex setting, we employ the notion of (δ, ϵ)-stationary points (Zhang et al., 2020).

Definition 2 A point $\mathbf{x}$ is a (δ, ϵ)-stationary point of a Lipschitz continuous function $f(\cdot)$ if there exists a finite set $S \subset \mathbb{B}(\mathbf{x}, \delta)$, such that for any $\mathbf{y}$ drawn uniformly at random from S, $\mathbb{E}[\mathbf{y}] = \mathbf{x}$

Algorithm 1 Online-to-non-convex conversion

Require: Initial point $\mathbf{x}_0$, $\mathbf{h}_0 = \mathbf{x}_0$, K, T, $M = KT$, online learning algorithm $\mathcal{A}$
1: **for** $m = 1, \ldots, M$ **do**
2: Receive $\mathbf{w}_m$ from $\mathcal{A}$ and set $\mathbf{h}_m = \mathbf{h}_{m-1} + \mathbf{w}_m$
3: Generate a random number $r_m \in [0, 1]$ uniformly and set $\mathbf{x}_m = \mathbf{h}_{m-1} + r_m \mathbf{w}_m$
4: Query the stochastic gradient $\mathbf{g}_m$ at $\mathbf{x}_m$ and send $\mathbf{g}_m$ to $\mathcal{A}$ as the gradient for round m
5: If $m \bmod T = 0$, then reinitialize the online algorithm $\mathcal{A}$
6: **end for**
7: Set $\mathbf{x}_{k,t} = \mathbf{x}_{(k-1)T+t}$ for $k = 1, \ldots, K$ and $t = 1, \ldots, T$
8: Set $\overline{\mathbf{x}}_k = \frac{1}{T} \sum_{t=1}^{T} \mathbf{x}_{k,t}$ for $k = 1, \ldots, K$
9: **Return:** $\mathbf{x} \sim \text{Unif}\{\overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_K\}$

and $\|\mathbb{E}[\nabla f(\mathbf{y})]\| \leq \epsilon$. Equivalently, for any point $\mathbf{x}$, a Lipschitz continuous function $f(\cdot)$ and $\delta > 0$, we define $\|\nabla f(\mathbf{x})\|_\delta = \min\{\|\mathbf{g}\| : \mathbf{g} \in \partial_\delta f(\mathbf{x})\}$. A point $\mathbf{x}$ is deemed a (δ, ϵ)-stationary point if it satisfies $\|\nabla f(\mathbf{x})\|_\delta \leq \epsilon$.

This notion is well-defined for any Lipschitz function and reduces to classical gradient stationarity when $f(\cdot)$ is smooth. Cutkosky et al. (2023) establish the optimal complexity of $O(\epsilon^{-3}\delta^{-1})$ for finding (δ, ϵ)-stationary points by developing the online-to-non-convex (O2NC) conversion, which is summarized in Algorithm 1. In each round m, the worker receives the decision $\mathbf{w}_m$ of an online algorithm $\mathcal{A}$. It then updates $\mathbf{h}_m = \mathbf{h}_{m-1} + \mathbf{w}_m$ and uniformly samples a point $\mathbf{x}_m$ from the line segment connecting $\mathbf{h}_{m-1}$ and $\mathbf{h}_m$. Subsequently, the worker queries the stochastic gradient $\mathbf{g}_m$ at $\mathbf{x}_m$ and feeds it back to the online algorithm as the gradient for round m, thereby reducing the non-smooth non-convex optimization problem to an online convex optimization (OCO) task.

A compressor is a mapping $\mathcal{C}(\cdot) : \mathbb{R}^d \to \mathbb{R}^d$ whose output admits an efficient representation (Karimireddy et al., 2019). We consider contractive compressors satisfying the following assumption.

Assumption 3 For a compression operator $\mathcal{C}(\cdot) : \mathbb{R}^d \to \mathbb{R}^d$, we assume there exists a constant $\omega \in (0, 1]$ such that $\mathbb{E}\left[\|\mathcal{C}(\mathbf{x}) - \mathbf{x}\|^2\right] \leq (1 - \omega)\|\mathbf{x}\|^2, \forall \mathbf{x} \in \mathbb{R}^d$, where the expectation is over the randomness of the compression operator $\mathcal{C}(\cdot)$.

Example. A prominent example is the scaled sign compressor, i.e., $\mathcal{C}(\mathbf{x}) = \frac{\|\mathbf{x}\|_1}{d} \operatorname{sign}(\mathbf{x})$, where $\operatorname{sign}(\cdot)$ is the sign operation. The scaled sign compressor satisfies Assumption 3 with $\omega = 1/d$.

4 Main Results

In this section, we first present our efficient online method with a corresponding regret bound. Building on the online method, we then develop the sign-based algorithm for non-smooth non-convex optimization under both the first-order and the zeroth-order settings.

4.1 First-Order Optimization

As discussed in Section 3, the O2NC conversion (Cutkosky et al., 2023) reduces the non-smooth non-convex optimization problem to an OCO task. Hence, the central step is to design a sign-based OCO algorithm with a sublinear regret bound. To this end, we adopt the following standard assumption on the feasible domain (Hazan, 2016), which is required for the regret analysis of the online method.

Assumption 4 The feasible domain $\mathcal{W}$ is convex, contains the origin, and is bounded with radius D, i.e., $\|\mathbf{w}\| \leq D$ for all $\mathbf{w} \in \mathcal{W}$.

As highlighted by Karimireddy et al. (2019), a method that naively applies the sign of the gradient fails to converge for convex loss functions even in the offline setting. Thus, we employ the classical EF mechanism (Seide et al., 2014), which is widely recognized for controlling the compression error in unconstrained optimization. Concretely, EF tracks the past compression error and compresses its

Algorithm 2 Compressed Follow-the-Regularized-Leader (C-FTRL)

Require: $\mathbf{w}_1 = \mathbf{0}$, $\mathbf{e}_1 = \mathbf{0}$, compressor $\mathcal{C}(\cdot)$, learning rate η

1: **for** $t = 1, \dots, T$ **do**
2: Submit the decision $\mathbf{w}_t$ and receive the gradient $\mathbf{g}_t$
3: Compute $\mathbf{s}_t = \mathcal{C}(\mathbf{e}_t + \mathbf{g}_t)$ and update $\mathbf{e}_{t+1} = \mathbf{e}_t + \mathbf{g}_t - \mathbf{s}_t$
4: Update $\mathbf{w}_{t+1}$ according to Equation (2)
5: **end for**

sum with the gradient to control the error. However, directly combining EF with the classical online methods such as online gradient descent (Zinkevich, 2003) introduces a fundamental difficulty: the projection error couples with the compression error and propagates across subsequent rounds, rendering the total error uncontrollable. Inspired by Yang et al. (2026), we incorporate EF into the Follow-the-Regularized-Leader (FTRL) framework (Hazan, 2016) to develop C-FTRL, detailed in Algorithm 2. Specifically, the player maintains a variable $\mathbf{e}_t$ to track the accumulated compression error and initializes $\mathbf{e}_1 = \mathbf{0}$. In each round t, it submits the decision $\mathbf{w}_t$ and receives the gradient $\mathbf{g}_t$. Then, the player computes the compressed update $\mathbf{s}_t = \mathcal{C}(\mathbf{e}_t + \mathbf{g}_t)$ and sets $\mathbf{e}_{t+1} = \mathbf{e}_t + \mathbf{g}_t - \mathbf{s}_t$. Crucially, C-FTRL updates the decision according to

$$\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathcal{W}} \left\langle \sum_{i=1}^{t} \mathbf{s}_i, \mathbf{w} \right\rangle + \frac{1}{\eta} \|\mathbf{w}\|^2 , \tag{2}$$

where η is the learning rate. Let $\hat{\mathbf{w}}_{t+1} = \arg\min_{\mathbf{w}\in\mathcal{W}} \left\langle \sum_{k=1}^{t} \mathbf{g}_k, \mathbf{w} \right\rangle + \frac{1}{\eta} \|\mathbf{w}\|^2$ denote the decision obtained with the uncompressed gradient. C-FTRL can ensure $\mathbb{E}\left[\|\mathbf{w}_{t+1} - \hat{\mathbf{w}}_{t+1}\|^2\right] \leq \frac{\eta^2}{4}\mathbb{E}\left[\|\mathbf{e}_{t+1}\|^2\right] \leq O(d^2\eta^2G^2)$, thereby effectively controlling the compression error. We formalize the theoretical guarantee for C-FTRL in the following.

Theorem 1 *Under Assumptions 2 and 4, by using the scaled sign operator and setting the learning rate* $\eta = \frac{D}{G\sqrt{dT}}$, *for any* $\mathbf{w} \in \mathcal{W}$, *C-FTRL ensures* $\mathbb{E}\left[\sum_{t=1}^{T}\langle \mathbf{g}_t, \mathbf{w}_t - \mathbf{w}\rangle\right] \leq 3DG\sqrt{dT}$.

By integrating C-FTRL into the O2NC conversion, we develop the Compressed O2NC (C-O2NC) algorithm. In each round m, it receives the decision $\mathbf{w}_m$ from C-FTRL and queries the stochastic gradient $\mathbf{g}_m$ at the point $\mathbf{x}_m = \mathbf{h}_{m-1} + r_m\mathbf{w}_m$, where r_m is drawn uniformly from $[0, 1]$. Then it sends the gradient $\mathbf{g}_m$ to C-FTRL, which updates its decision via compressed information. Based on Theorem 1, we derive the query complexity of C-O2NC for finding (δ, ϵ)-stationary points.

Theorem 2 *Considering the notation from Algorithm 1 and using Algorithm 2 as the online method, we assume a budget of* N *gradient queries. For any* $\delta > 0$, *set* $D = \delta/T$, $K = \lfloor N/T \rfloor$ *and* $T = \min\left\{\left\lceil\left(\frac{d^{1/2}G\delta N}{\Delta}\right)^{2/3}\right\rceil, \frac{N}{2}\right\}$. *Under Assumptions 1 and 2 on* $f(\cdot)$, *for the scaled sign compressor, C-O2NC ensures* $\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \leq O(d^{1/3}\delta^{-1/3}N^{-1/3})$.

Remark 1 To ensure $\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \leq \epsilon$, the query complexity N is $O(d\epsilon^{-3}\delta^{-1})$, which preserves the optimal dependence on ϵ and δ (Cutkosky et al., 2023).

4.2 Zeroth-Order Optimization

In this section, we consider the zeroth-order optimization setting, where only noisy function evaluations $f(\mathbf{x}; \xi)$ are accessible. The primary challenge in this setting is the absence of the first-order gradient information. To address this issue, we adopt the uniform smoothing technique (Duchi et al., 2012; Yousefian et al., 2012; Nesterov & Spokoiny, 2017; Shamir, 2017), which is a common basis for zeroth-order methods. Concretely, for a Lipschitz function $f(\cdot)$ and a smoothing parameter $\rho > 0$, the uniformly smoothed version of $f(\cdot)$ is defined as

$$f_\rho(\mathbf{x}) = \mathbb{E}_{\mathbf{z}\sim\mathrm{Unif}(\mathbb{B}(\mathbf{0},1))}\left[f(\mathbf{x} + \rho\mathbf{z})\right], \tag{3}$$

which satisfies the properties stated in the following lemma.

Algorithm 3 Gradient estimator

Require: function $f(\cdot)$, point $\mathbf{x} \in \mathbb{R}^d$, smoothing parameter ρ, sample ξ

1: Sample $\mathbf{z}$ uniformly from the unit sphere and query $f(\mathbf{x}+\rho\mathbf{z};\xi)$, $f(\mathbf{x}-\rho\mathbf{z};\xi)$

2: Compute $\hat{\mathbf{g}} = \frac{d}{2\rho}\left(f(\mathbf{x}+\rho\mathbf{z};\xi) - f(\mathbf{x}-\rho\mathbf{z};\xi)\right)\mathbf{z}$

3: **Return:** Gradient estimator $\hat{\mathbf{g}}$

Algorithm 4 FCC$(\mathbf{g}, \mathcal{C}(\cdot), R)$

1: **Input:** data $\mathbf{g}$, compressor $\mathcal{C}(\cdot)$, number of compression rounds R

2: Initialize $\mathbf{r}^1 = \mathbf{0}$

3: **for** $k = 1$ to R **do**

4: Compute $\mathbf{c}^k = \mathcal{C}(\mathbf{g} - \mathbf{r}^k)$

5: Calculate $\mathbf{r}^{k+1} = \mathbf{r}^k + \mathbf{c}^k$

6: **end for**

7: **return** $\mathbf{r}^{R+1}$

Lemma 1 (Proposition 2.2 in Lin et al. (2022)) The function $f_\rho(\cdot)$ is differentiable and L_0-Lipschitz with $|f_\rho(\mathbf{x}) - f(\mathbf{x})| \le \rho L_0$ for all $\mathbf{x} \in \mathbb{R}^d$. The gradient of $f_\rho(\cdot)$ satisfies

$$\nabla f_\rho(\mathbf{x}) = \mathbb{E}_{\mathbf{z}\sim \mathrm{Unif}(\mathbb{B}(\mathbf{0},1))}\left[\nabla f(\mathbf{x}+\rho\mathbf{z})\right].$$

For the smoothed function $f_\rho(\cdot)$, we can construct an unbiased gradient estimator using solely function evaluations (Shamir, 2017), as summarized in Algorithm 3. Specifically, given a point $\mathbf{x}$, we sample a vector $\mathbf{z}$ uniformly from the unit sphere. Then, we query $f(\mathbf{x}+\rho\mathbf{z};\xi)$ and $f(\mathbf{x}-\rho\mathbf{z};\xi)$, and form the estimator as $\hat{\mathbf{g}} = \frac{d}{2\rho}\left(f(\mathbf{x}+\rho\mathbf{z};\xi) - f(\mathbf{x}-\rho\mathbf{z};\xi)\right)\mathbf{z}$.

A natural extension of C-O2NC to the zeroth-order setting is to replace the first-order gradient with the gradient estimator. However, this straightforward substitution leads to an unsatisfactory complexity. The underlying issue stems from the high variance of the estimator, which is bounded by $\mathbb{E}\left[\|\hat{\mathbf{g}}_m\|^2\right] \le O(dL_0^2)$, scaling linearly with the dimensionality d. If we directly adopt the gradient estimator, the estimation variance couples with the compression error, resulting in an inflated bound on the residual, i.e., $\mathbb{E}\left[\|\mathbf{e}_t\|^2\right] \le O(d^3L_0^2)$. Consequently, this naive approach only attains an unsatisfactory complexity of $O(d^2\epsilon^{-3}\delta^{-1})$ (refer to Appendix C.8 for a detailed proof). Thus, a natural question is *whether the dependence on d can be improved in the zeroth-order setting*.

We answer this question affirmatively by proposing a novel *compression-estimation decoupling strategy*, built upon two ingredients. First, we construct a refined gradient estimator by averaging R independent calls to Algorithm 3 to obtain an estimator satisfying $\mathbb{E}\left[\|\hat{\mathbf{g}}_m\|^2\right] = \mathbb{E}\left[\left\|\frac{1}{R}\sum_{s=1}^R \hat{\mathbf{g}}_{m,s}\right\|^2\right] \le O\left((d/R+1)L_0^2\right)$, where $\hat{\mathbf{g}}_{m,s}$ denotes the output of the s-th call. While this averaging yields a more accurate estimator, it also multiplies the number of function queries per update by R. Hence, on its own, this technique does not improve the overall query complexity. This leads to our second idea: since we are already expending multiple queries per update, we can simultaneously reduce the compression error. To this end, we employ the FCC technique (Huang et al., 2022), summarized in Algorithm 4. FCC recursively compresses the residual over R rounds and outputs the accumulated update $\mathbf{r}^{R+1}$. The following lemma provides its formal guarantee.

Lemma 2 (Lemma 2 in Huang et al. (2022)) *For any ω-contractive compressor $\mathcal{C}(\cdot)$ and any integer $R \ge 1$, Algorithm 4 ensures $\mathbb{E}\left[\|\mathbf{r}^{R+1} - \mathbf{g}\|^2\right] \le (1-\omega)^R\|\mathbf{g}\|^2, \forall \mathbf{g} \in \mathbb{R}^d$.*

For the scaled sign compressor with $\omega = 1/d$, if we set $R \ge d$, we can ensure $\mathbb{E}\left[\|\mathbf{r}^{R+1} - \mathbf{g}\|^2\right] \le \frac{1}{e}\|\mathbf{g}\|^2$, thereby effectively controlling the compression error. With $R = d$, both the estimation variance and the compression error become independent of d, yielding a better theoretical guarantee. Our method, named ReC-O2NC, is presented in Algorithm 5, which employs C-FTRL (Algorithm 2) as the online method and utilizes FCC (Algorithm 4) as the compressor. In each round,

Algorithm 5 Repeated-Compression O2NC (ReC-O2NC)

Require: Initial point $\mathbf{x}_0$, $\mathbf{h}_0 = \mathbf{x}_0$, K, T, $M = KT$, ρ, R, online learning algorithm $\mathcal{A}$

for $m = 1, \dots, M$ **do**
  Get $\mathbf{w}_m$ from $\mathcal{A}$ and set $\mathbf{h}_m = \mathbf{h}_{m-1} + \mathbf{w}_m$
  Generate a random number $r_m \in [0, 1]$ uniformly and set $\mathbf{x}_m = \mathbf{h}_{m-1} + r_m \mathbf{w}_m$
  for $s = 1, \dots, R$ **do**
    Run Algorithm 3 and receive $\hat{\mathbf{g}}_{m,s}$
  end for
  Send $\hat{\mathbf{g}}_m = \frac{1}{R} \sum_{s=1}^{R} \hat{\mathbf{g}}_{m,s}$ to $\mathcal{A}$ as the gradient for round m and replace the compressor with Algorithm 4
  If $m \bmod T = 0$, then reinitialize the online algorithm $\mathcal{A}$
end for
Set $\mathbf{x}_{k,t} = \mathbf{x}_{(k-1)T+t}$ for $k = 1, \dots, K$ and $t = 1, \dots, T$
Set $\overline{\mathbf{x}}_k = \frac{1}{T} \sum_{t=1}^{T} \mathbf{x}_{k,t}$ for $k = 1, \dots, K$
Return: $\mathbf{x} \sim \text{Unif}\{\overline{\mathbf{x}}_1, \dots, \overline{\mathbf{x}}_K\}$

we construct an unbiased gradient estimator with R samples and send it to C-FTRL. Subsequently, C-FTRL performs FCC to recursively apply the scaled sign operator to compress the information for R rounds. The following theorem establishes the query complexity of ReC-O2NC.

Theorem 3 *Considering the notation from Algorithm 5 and using Algorithm 2 as the online method, we assume a total budget of $2N$ function value queries. For any $\delta > 0$, set $K = \lfloor N/(RT) \rfloor$, $\rho = \min\left\{\frac{\delta}{2}, \frac{\Delta}{L_0}\right\}$, $\nu = \delta - \rho$, $D = \nu/T$, $T = \min\left\{\left\lceil \left(\frac{L_0 \nu N}{R(\Delta + \rho L_0)}\right)^{2/3} \right\rceil, \frac{N}{2R}\right\}$ and $R = d$. Under Assumptions 1 and 2 on $f(\cdot)$, for the scaled sign compressor, ReC-O2NC ensures $\mathbb{E}\left[\frac{1}{K} \sum_{k=1}^{K} \|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \le O\left(d^{1/3} \delta^{-1/3} N^{-1/3}\right)$.*

Remark 2 We have established a query complexity of $O(d\epsilon^{-3}\delta^{-1})$, which is of the same order as that of C-O2NC in the first-order setting. Moreover, it matches the optimal bound for zeroth-order non-smooth non-convex optimization without compression (Kornowski & Shamir, 2024).

4.3 General Compressors

The applicability of our methods extends well beyond the sign compressor. Our framework accommodates general compressors satisfying Assumption 3, such as Rand-K. We first establish the corresponding guarantee for the first-order setting in the following theorem.

Theorem 4 *Considering the notation from Algorithm 1 and using Algorithm 2 as the online method, we assume a budget of N gradient queries. For any $\delta > 0$, set $D = \delta/T$, $K = \lfloor N/T \rfloor$, and $T = \min\left\{\left\lceil \left(\frac{G\delta N}{\omega^{1/2}\Delta}\right)^{2/3} \right\rceil, \frac{N}{2}\right\}$. Under Assumptions 1 and 2 on $f(\cdot)$, for any compressor satisfying Assumption 3, C-O2NC ensures $\mathbb{E}\left[\frac{1}{K} \sum_{k=1}^{K} \|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \le O\left(\omega^{-1/3} \delta^{-1/3} N^{-1/3}\right)$.*

In Theorem 3, R is set to d to simultaneously control the estimation and compression errors. However, for general compressors with $\omega < 1/d$, this choice no longer suffices to suppress the compression error. Therefore, we set $R = \max\{\omega^{-1}, d\}$, which ensures the compression error is sufficiently small for any $\omega \in (0, 1]$. The following theorem provides the resulting query complexity for general compressors.

Theorem 5 *Considering the notation from Algorithm 5 and using Algorithm 2 as the online method, we assume a budget of $2N$ function value queries. For any $\delta > 0$, set $K = \lfloor N/(RT) \rfloor$, $\rho = \min\left\{\frac{\delta}{2}, \frac{\Delta}{L_0}\right\}$, $\nu = \delta - \rho$, $D = \nu/T$, $T = \min\left\{\left\lceil \left(\frac{L_0 \nu N}{R(\Delta + \rho L_0)}\right)^{2/3} \right\rceil, \frac{N}{2R}\right\}$, and $R = \max\{\lceil \omega^{-1} \rceil, d\}$. Under Assumptions 1 and 2 on $f(\cdot)$, for any compressor satisfying Assumption 3, ReC-O2NC ensures $\mathbb{E}\left[\frac{1}{K} \sum_{k=1}^{K} \|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \le O\left(\max\{\omega^{-1/3}, d^{1/3}\} \delta^{-1/3} N^{-1/3}\right)$.*

Algorithm 6 D-FTCL

Require: $\mathbf{w}_1 = \mathbf{0}$, $\mathbf{e}_1^i = \mathbf{0}$, $\hat{\mathbf{e}}_1 = \mathbf{0}$, compressor $\mathcal{C}(\cdot)$, number of compression rounds R

for $t = 1, \dots, T$ **do**
  for each worker $i = 1, \dots, n$ **in parallel do**
    Play the decision $\mathbf{w}_t$ and receive $\mathbf{g}_t^i$
    Send $\mathbf{v}_t^i = \mathrm{FCC}(\mathbf{e}_t^i + \mathbf{g}_t^i, \mathcal{C}(\cdot), R)$ to the server and update $\mathbf{e}_{t+1}^i = \mathbf{e}_t^i + \mathbf{g}_t^i - \mathbf{v}_t^i$
  end for
  for server **do**
    Compute $\mathbf{v}_t = \frac{1}{n}\sum_{i=1}^n \mathbf{v}_t^i$
    Send $\mathbf{s}_t = \mathrm{FCC}(\hat{\mathbf{e}}_t + \mathbf{v}_t, \mathcal{C}(\cdot), R)$ to all workers and update $\hat{\mathbf{e}}_{t+1} = \hat{\mathbf{e}}_t + \mathbf{v}_t - \mathbf{s}_t$
  end for
  for each worker $i = 1, \dots, n$ **in parallel do**
    Update $\mathbf{w}_{t+1} = \arg\min_{\mathbf{w}\in\mathcal{W}} \left\langle \sum_{k=1}^t \mathbf{s}_k, \mathbf{w} \right\rangle + \frac{1}{\eta}\|\mathbf{w}\|^2$
  end for
end for

Remark 3 For general compressors, our methods achieve query complexities of $O(\omega^{-1}\epsilon^{-3}\delta^{-1})$ and $O(\max\{\omega^{-1}, d\}\epsilon^{-3}\delta^{-1})$ in the first-order and zeroth-order settings, respectively. Remarkably, in the zeroth-order setting, when $d \le \omega^{-1}$, the complexity of $O(\omega^{-1}\epsilon^{-3}\delta^{-1})$ matches the result of C-O2NC. When $d \ge \omega^{-1}$, the complexity of $O(d\epsilon^{-3}\delta^{-1})$ coincides with the result in the zeroth-order setting without compressed communication (Kornowski & Shamir, 2024).

5 Extension to Distributed Optimization

In this section, we extend our methods to the *distributed* stochastic optimization setting, where n local workers collaborate with a central server to minimize the global objective

$$\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) = \frac{1}{n}\sum_{i=1}^n f^i(\mathbf{x}), \qquad f^i(\mathbf{x}) = \mathbb{E}_{\xi^i\sim\mathcal{D}_i}\left[f^i(\mathbf{x};\xi^i)\right], \tag{4}$$

where $\mathcal{D}_i$ denotes the data distribution of worker i and $f^i(\mathbf{x})$ is its local loss function. The message exchange between the workers and the server may incur substantial communication overhead in large-scale learning scenarios. To alleviate this bottleneck, numerous studies investigate distributed optimization with compressed communication. Existing work in this line, however, predominantly relies on convexity or smoothness assumptions, leaving the non-smooth non-convex regime largely unexplored. We bridge this gap in both the first-order and zeroth-order settings.

A direct application of C-O2NC to the distributed setting yields only unidirectional compression, in which each worker sends the compressed message to the server, whereas the server broadcasts the aggregated information back to the workers without compression. To further reduce the communication cost, we aim to develop a method that supports bidirectional compression. To this end, we adopt D-FTCL (Yang et al., 2026), a distributed OCO algorithm with bidirectional compression, summarized in Algorithm 6. Specifically, D-FTCL incorporates EF on both the worker and server sides, and further leverages the FCC (Huang et al., 2022) to suppress the compression error. With D-FTCL as the online method, we develop DistC-O2NC and DistReC-O2NC for the first-order and zeroth-order settings, respectively, as outlined in Algorithm 7.

A key challenge in extending the O2NC conversion to the distributed setting arises from its inherent randomness. In each round m, every worker updates its decision as $\mathbf{x}_m = \mathbf{h}_{m-1} + r_m\mathbf{w}_m$, where r_m is drawn uniformly from $[0, 1]$. If each worker samples r_m independently, the local decisions diverge across workers, and the resulting gradients no longer correspond to the same global iterate, which violates the distributed optimization protocol (Huang et al., 2022). To resolve this issue, we synchronize r_m across all workers via a shared random seed, so that every worker queries the gradient at an *identical* point. While D-FTCL provides a regret bound for distributed OCO, the transfer of this guarantee to the non-smooth non-convex setting under the O2NC framework requires an extension of our single-node analysis, which we present in the following theorem.

Algorithm 7 DistC-O2NC and DistReC-O2NC

Require: initial point $\mathbf{x}_0$, K, T, $M = KT$, online learning algorithm $\mathcal{A}$

for each worker $i = 1, \ldots, n$ **in parallel do**
  for $m = 1, \ldots, M$ **do**
    Get $\mathbf{w}_m$ from $\mathcal{A}$ and set $\mathbf{h}_m = \mathbf{h}_{m-1} + \mathbf{w}_m$
    Generate a random number $r_m \in [0, 1]$ uniformly ▷ Synchronize r_m across all workers
    Set $\mathbf{x}_m = \mathbf{h}_{m-1} + r_m \mathbf{w}_m$
    Query the gradient $\mathbf{g}_m^i = \frac{1}{R} \sum_{s=1}^{R} \nabla f^i(\mathbf{x}_m; \xi_{m,s}^i)$ and send $\mathbf{g}_m^i$ to $\mathcal{A}$ as the gradient for round m ▷ first-order setting
    for $s = 1$ to R **do**
      Run Algorithm 3 and receive $\hat{\mathbf{g}}_{m,s}^i$
    end for
    Send $\hat{\mathbf{g}}_m^i = \frac{1}{R} \sum_{s=1}^{R} \hat{\mathbf{g}}_{m,s}^i$ to $\mathcal{A}$ as the gradient for round m ▷ zeroth-order setting
    If $m \bmod T = 0$, then reinitialize the online algorithm $\mathcal{A}$
  end for
end for
Set $\mathbf{x}_{k,t} = \mathbf{x}_{(k-1)T+t}$ for $k = 1, \ldots, K$ and $t = 1, \ldots, T$
Set $\overline{\mathbf{x}}_k = \frac{1}{T} \sum_{t=1}^{T} \mathbf{x}_{k,t}$ for $k = 1, \ldots, K$
Return: $\mathbf{x} \sim \text{Unif}\{\overline{\mathbf{x}}_1, \ldots, \overline{\mathbf{x}}_K\}$

Theorem 6 *Considering the notation in Algorithm 7 and using Algorithm 6 as the online method, we assume a budget of N gradient queries for each local worker. For any $\delta > 0$, set $D = \delta/T$, $K = \lfloor N/(RT) \rfloor$, $T = \min\left\{ \left\lceil \left(\frac{G\delta N}{R\Delta}\right)^{2/3} \right\rceil, \frac{N}{2R} \right\}$, and $R = \lceil \omega^{-1} \rceil$. Under Assumptions 1 and 2 on $f^i(\cdot)$, for any compressor satisfying Assumption 3, DistC-O2NC ensures $\mathbb{E}\left[\frac{1}{K} \sum_{k=1}^{K} \|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \leq O\left(\omega^{-1/3} \delta^{-1/3} N^{-1/3}\right)$.*

Remark 4 We establish an $O(\omega^{-1}\epsilon^{-3}\delta^{-1})$ query complexity for distributed non-smooth non-convex optimization with compressed communication, which is the *first* result in this setting.

In the zeroth-order regime, we again employ our compression-estimation decoupling strategy to control both the estimation error and the compression error. We establish the query complexity for DistReC-O2NC in the following theorem.

Theorem 7 *Considering the notation from Algorithm 7 and using Algorithm 6 as the online method, we assume a total budget of $2N$ function value queries for each local worker. For any $\delta > 0$, set $K = \lfloor N/(RT) \rfloor$, $\rho = \min\left\{\frac{\delta}{2}, \frac{\Delta}{L_0}\right\}$, $\nu = \delta - \rho$, $D = \nu/T$, $T = \min\left\{ \left\lceil \left(\frac{L_0 \nu N}{R(\Delta + \rho L_0)}\right)^{2/3} \right\rceil, \frac{N}{2R} \right\}$, and $R = \max\{\lceil \omega^{-1} \rceil, d\}$. Under Assumptions 1 and 2 on $f^i(\cdot)$, for any compressor satisfying Assumption 3, DistReC-O2NC ensures $\mathbb{E}\left[\frac{1}{K} \sum_{k=1}^{K} \|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \leq O\left(\max\{d^{1/3}, \omega^{-1/3}\} \delta^{-1/3} N^{-1/3}\right)$.*

Remark 5 Our method decouples the effects of gradient estimation and compression and achieves a query complexity of $O(\max\{\omega^{-1}, d\}\epsilon^{-3}\delta^{-1})$ in the distributed zeroth-order setting.

6 Conclusion

In this paper, we study sign-based methods for non-smooth non-convex stochastic optimization. First, we develop C-O2NC and establish a query complexity of $O(d\epsilon^{-3}\delta^{-1})$, which is optimal with respect to ϵ and δ. Second, we consider the zeroth-order setting and propose ReC-O2NC, which attains the same $O(d\epsilon^{-3}\delta^{-1})$ complexity as C-O2NC through a compression-estimation decoupling strategy that simultaneously controls the estimation and compression errors. Beyond the sign compressor, our analysis carries over to general compressors. Finally, we investigate the distributed setting with bidirectional compression and design two novel algorithms for the first-order and zeroth-order regimes, respectively, and establish their convergence guarantees.

AI USE STATEMENT

In this work, we used generative AI tools to polish the writing. We take responsibility for the final content of this work, including text and claims produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

All theoretical results are stated together with their required assumptions in the main text, and complete proofs are provided in the appendix. The experimental setup, hyperparameters, and training details are documented in the appendix.

ETHICS STATEMENT

This work is methodological and does not involve human subjects, personal data, or potentially harmful applications. Accordingly, we identify no ethical concerns beyond those covered by standard research integrity practices.

REFERENCES

Alekh Agarwal, Ofer Dekel, and Lin Xiao. Optimal algorithms for online convex optimization with multi-point bandit feedback. In *Proceedings of the 23rd Conference on Learning Theory (COLT)*, pp. 28–40, 2010.

Jeremy Bernstein, Yu-Xiang Wang, Kamyar Azizzadenesheli, and Animashree Anandkumar. Signsgd: Compressed optimisation for non-convex problems. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 560–569, 2018.

Jeremy Bernstein, Jiawei Zhao, Kamyar Azizzadenesheli, and Anima Anandkumar. Signsgd with majority vote is communication efficient and fault tolerant. In *International Conference on Learning Representations*, 2019.

Aleksandr Beznosikov, Samuel Horváth, Peter Richtárik, and Mher Safaryan. On biased compression for distributed learning. *Journal of Machine Learning Research*, 24(276):1–50, 2023.

Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology*, 2(3):1–27, 2011.

Lesi Chen, Jing Xu, and Luo Luo. Faster gradient-free algorithms for nonsmooth nonconvex stochastic optimization. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pp. 5219–5233, 2023.

Xinyan Chen, Weiguo Gao, and Luo Luo. Decentralized nonsmooth nonconvex optimization with client sampling. *arXiv preprint arXiv:2601.19381*, 2026.

Frank H Clarke, Yu S Ledyaev, Ronald J Stern, and RR Wolenski. *Nonsmooth analysis and control theory*. Springer, 1998.

Ashok Cutkosky and Francesco Orabona. Momentum-based variance reduction in non-convex sgd. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2019.

Ashok Cutkosky, Harsh Mehta, and Francesco Orabona. Optimal, stochastic, non-smooth, non-convex optimization through online-to-non-convex conversion. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, 2023.

John C. Duchi, Alekh Agarwal, and Martin J. Wainwright. Dual averaging for distributed optimization: Convergence analysis and network scaling. *IEEE Transactions on Automatic Control*, pp. 592–606, 2011.

John C. Duchi, Peter L. Bartlett, and Martin J. Wainwright. Randomized smoothing for stochastic optimization. *SIAM Journal on Optimization*, 22(2):674–701, 2012.

Lawrence Craig Evans and Ronald F. Gariepy. *Measure Theory and Fine Properties of Functions, Revised Edition*. Chapman and Hall/CRC, 2015.

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

Ilyas Fatkhullin, Igor Sokolov, Eduard Gorbunov, Zhize Li, and Peter Richtárik. Ef21 with bells & whistles: Six algorithmic extensions of modern error feedback. *Journal of Machine Learning Research*, 26(189):1–50, 2025.

Yuan Gao, Anton Rodomanov, Jeremy Rack, and Sebastian U Stich. Accelerated distributed optimization with compression and error feedback. *arXiv preprint arXiv:2503.08427*, 2025.

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

Saeed Ghadimi and Guanghui Lan. Stochastic first- and zeroth-order methods for nonconvex stochastic programming. *SIAM Journal on Optimization*, 23(4):2341–2368, 2013.

Allen A Goldstein. Optimization of lipschitz continuous functions. *Mathematical Programming*, 13 (1):14–22, 1977.

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

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.

Xinmeng Huang, Yiming Chen, Wotao Yin, and Kun Yuan. Lower bounds and nearly optimal algorithms in distributed learning with communication compression. *Advances in Neural Information Processing Systems 35 (NeurIPS)*, pp. 18955–18969, 2022.

Rustem Islamov, Yarden As, and Ilyas Fatkhullin. Safe-ef: Error feedback for nonsmooth constrained optimization. In *Proceedings of the 42nd International Conference on Machine Learning (ICML)*, pp. 26542–26585, 2025.

Michael I. Jordan, Guy Kornowski, Tianyi Lin, Ohad Shamir, and Manolis Zampetakis. Deterministic nonsmooth nonconvex optimization. In *Proceedings of the 36th Conference on Learning Theory (COLT)*, pp. 4570–4597, 2023.

Thomas Kailath, Ali H. Sayed, and Babak Hassibi. *Linear Estimation*. Prentice Hall, 2000.

Sai Praneeth Karimireddy, Quentin Rebjock, Sebastian Stich, and Martin Jaggi. Error feedback fixes signsgd and other gradient compression schemes. In *Proceedings of the 36th International Conference on Machine Learning (ICML)*, pp. 3252–3261, 2019.

Sarit Khirirat, Abdurakhmon Sadiev, Artem Riabinin, Eduard Gorbunov, and Peter Richtárik. Error feedback under (L_0, L_1)-smoothness: Normalization and momentum. In *Advances in Neural Information Processing Systems 38 (NeurIPS)*, 2025.

Guy Kornowski and Ohad Shamir. An algorithm with optimal dimension-dependence for zero-order nonsmooth nonconvex stochastic optimization. *Journal of Machine Learning Research*, 25(122): 1–14, 2024.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521(7553):436–444, 2015.

Tianyi Lin, Zaiwei Zheng, and Michael I. Jordan. Gradient-free methods for deterministic and stochastic nonsmooth nonconvex optimization. In *Advances in Neural Information Processing Systems 35 (NeurIPS)*, pp. 26160–26175, 2022.

Zijian Liu. Online convex optimization with heavy tails: Old algorithms, new regrets, and applications. In *Proceedings of The 37th International Conference on Algorithmic Learning Theory*, pp. 1–47, 2026.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. *arXiv preprint arXiv:1706.06083*, 2017.

Angelia Nedić and Asuman Ozdaglar. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009.

Y. Nesterov and V. Spokoiny. Random gradient-free minimization of convex functions. *Foundations of Computational Mathematics*, 17(2):527–566, 2017.

Peter Richtárik, Igor Sokolov, and Ilyas Fatkhullin. Ef21: A new, simpler, theoretically better, and practically faster error feedback. In *Advances in Neural Information Processing Systems 34 (NeurIPS)*, pp. 4384–4396, 2021.

Peter Richtárik, Elnur Gasanov, and Konstantin Burlachenko. Error feedback reloaded: From quadratic to arithmetic mean of smoothness constants. In *Proceedings of the 12th International Conference on Learning Representations (ICLR)*, 2024.

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 *Interspeech*, pp. 1058–1062, 2014.

Shahin Shahrampour and Ali Jadbabaie. Distributed online optimization in dynamic environments using mirror descent. *IEEE Transactions on Automatic Control*, 63(3):714–725, 2017.

Ohad Shamir. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *Journal of Machine Learning Research*, 18(52):1703–1713, 2017.

Sebastian U. Stich, Jean-Baptiste Cordonnier, and Martin Jaggi. Sparsified sgd with memory. *Advances in Neural Information Processing Systems 31 (NeurIPS)*, pp. 4447–4458, 2018.

Lai Tian, Kaiwen Zhou, and Anthony Man-Cho So. On the finite-time complexity and practical computation of approximate stationarity concepts of lipschitz functions. In *Proceedings of the 39th International Conference on Machine Learning (ICML)*, volume 162, pp. 21360–21379, 2022.

Jiaxiang Wu, Weidong Huang, Junzhou Huang, and Tong Zhang. Error compensated quantized sgd and its applications to large-scale distributed optimization. In *Proceedings of the 35th International Conference on Machine Learning (ICML)*, pp. 5325–5333, 2018.

Yonggui Yan, Jie Chen, Pin-Yu Chen, Xiaodong Cui, Songtao Lu, and Yangyang Xu. Compressed decentralized proximal stochastic gradient method for nonconvex composite problems with heterogeneous data. In *Proceedings of the 40th International Conference on Machine Learning (ICML)*, pp. 39035–39061, 2023.

Sifan Yang, Dan-Yue Li, and Lijun Zhang. Distributed online convex optimization with compressed communication: Optimal regret and applications. *arXiv preprint arXiv:2604.09276*, 2026.

Farzad Yousefian, Angelia Nedić, and Uday V. Shanbhag. On stochastic gradient and subgradient methods with adaptive steplength sequences. *Automatica*, 48(1):56–67, 2012.

Dingzhi Yu, Rui Pan, Yuxing Liu, Difan Zou, and Tong Zhang. StoSignSGD: Unbiased structural stochasticity fixes SignSGD for training large language models. *arXiv preprint arXiv:2604.15416*, 2026.

Jingzhao Zhang, Hongzhou Lin, Stefanie Jegelka, Suvrit Sra, and Ali Jadbabaie. Complexity of finding stationary points of nonconvex nonsmooth functions. In *Proceedings of the 37th International Conference on Machine Learning (ICML)*, pp. 11173–11182, 2020.

Qinzi Zhang and Ashok Cutkosky. Random scaling and momentum for non-smooth non-convex optimization. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, pp. 58780–58799, 2024.

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

A Additional Discussions on Related Work

Non-smooth non-convex optimization has garnered significant attention due to its prevalence in applications, such as training deep neural networks with non-smooth activations (LeCun et al., 2015; He et al., 2016), adversarial training (Madry et al., 2017) and control theory (Kailath et al., 2000). Unlike smooth optimization, finding an ϵ-stationary point for non-smooth non-convex functions is theoretically intractable in finite time (Zhang et al., 2020). To address this, Zhang et al. (2020) introduce the notion of (δ, ϵ)-stationary points. While their algorithm achieves a query complexity of $O(\epsilon^{-4}\delta^{-1})$ in the stochastic setting, it relies on a directional gradient oracle that requires subgradients to obey a specific linear equation, a condition that is hard to meet in many applications. Tian et al. (2022) later remove this requirement and achieve the same complexity under the standard stochastic gradient oracle. Recently, Cutkosky et al. (2023) make a pivotal advance by proposing the online-to-non-convex conversion (O2NC). By reducing the non-smooth non-convex problem to minimizing regret in online convex optimization, they achieve the optimal $O(\epsilon^{-3}\delta^{-1})$ query complexity. Subsequently, Zhang & Cutkosky (2024) propose an exponentiated variant of the O2NC framework, which has better performance in practice. More recently, Liu (2026) studies non-smooth non-convex optimization with heavy-tailed noise and establishes upper and lower complexity bounds.

Lin et al. (2022) initiate the study of non-smooth non-convex zeroth-order optimization, where the learner has access only to noisy function evaluations rather than first-order gradients. To tackle this challenge, they adopt the randomized smoothing technique (Agarwal et al., 2010; Ghadimi & Lan, 2013) to construct a smooth surrogate function. Since the surrogate function is smooth, it renders the problem amenable to standard smooth optimization analysis, further leading to a complexity of $O(d^{3/2}\epsilon^{-4}\delta^{-1})$. While Chen et al. (2023) improve the dependence on ϵ via the variance reduction technique (Fang et al., 2018; Cutkosky & Orabona, 2019), the $O(d^{3/2})$ factor stemming from the smoothing approximation still remains. Later, Kornowski & Shamir (2024) extend the O2NC conversion to the zeroth-order setting and establish a complexity of $O(d\epsilon^{-3}\delta^{-1})$, which is optimal in the dimensionality.

We also note that a recent work (Chen et al., 2026) investigates decentralized non-smooth non-convex optimization, where local learners are connected through a network modeled by an undirected graph. They use a client-sampling technique to reduce communication rounds in both first-order and zeroth-order settings.

Comparison with Yan et al. (2023). Yan et al. (2023) study compressed decentralized stochastic optimization for composite objectives of the form $\varphi(x) = n^{-1}\sum_{i=1}^{n} f_i(x) + r(x)$, where each f_i is L-smooth and possibly non-convex, and r is a closed convex and possibly non-smooth regularizer. Although the overall objective can be non-smooth, it is L-weakly convex because each f_i is L-smooth and r is convex. Their analysis explicitly relies on this weak-convexity structure: they construct a smooth surrogate of the objective through proximal regularization (the Moreau envelope) and measure stationarity using the gradient of this surrogate. Their oracle model also differs from ours, since it requires separate access to stochastic gradients of the smooth components and to the proximal mapping of r. They measure convergence using the gradient of the Moreau envelope together with a consensus-error term, whereas our work adopts (δ, ϵ)-Goldstein stationarity, which is widely used in the non-smooth non-convex literature.

B Limitations and Future Work

While our work establishes the first theoretical guarantees for gradient compression in non-smooth non-convex optimization, several promising directions remain open for future investigation.

First, our guarantees hold in expectation, and extending them to high-probability bounds is a theoretically appealing direction. However, this extension is non-trivial: the additional randomness introduced by the compression operator complicates the concentration analysis.

Second, our algorithms require prior knowledge of the Lipschitz constant of the loss function. While such an assumption is standard in the existing literature on non-smooth non-convex optimization (Zhang et al., 2020; Lin et al., 2022; Cutkosky et al., 2023; Kornowski & Shamir, 2024), developing parameter-free counterparts that automatically adapt to unknown problem parameters would

substantially enhance the practical applicability of our methods, and we leave this as a promising direction for future work.

C Proof of Theorems

In this section, we present the proofs of all theorems.

C.1 Proof of Theorem 1

We first define a virtual decision and introduce some useful lemmas.

$$\hat{\mathbf{w}}_{t+1} = \arg\min_{\mathbf{w}\in\mathcal{W}} \left\langle \sum_{k=1}^{t} \mathbf{g}_k, \mathbf{w} \right\rangle + \frac{1}{\eta} \|\mathbf{w}\|^2 . \tag{5}$$

It is not hard to verify that $\hat{\mathbf{w}}_{t+1}$ is the decision updated with the uncompressed gradient.

Lemma 3 (Lemma 6.6 in Garber & Hazan (2016)) *Let $\{\ell_t(\mathbf{w})\}_{t=1}^T$ be a sequence of functions and $\mathbf{w}_t^* \in \arg\min_{\mathbf{w}\in\mathcal{W}} \sum_{k=1}^t \ell_k(\mathbf{w})$ for any $t \in [T]$. Then, it holds that*

$$\sum_{t=1}^{T} \ell_t(\mathbf{w}_t^*) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} \ell_t(\mathbf{w}) \leq 0.$$

Lemma 4 (Lemma 5 in Duchi et al. (2011)) *Let $\Pi_{\mathcal{W}}(\mathbf{u}, \eta) = \arg\min_{\mathbf{w}\in\mathcal{W}} \langle \mathbf{u}, \mathbf{w}\rangle + \frac{1}{\eta}\|\mathbf{w}\|^2$. For any $\mathbf{u}, \mathbf{v} \in \mathbb{R}^d$, we have*

$$\|\Pi_{\mathcal{W}}(\mathbf{u}, \eta) - \Pi_{\mathcal{W}}(\mathbf{v}, \eta)\| \leq \frac{\eta}{2}\|\mathbf{u} - \mathbf{v}\|.$$

We define $\ell_1(\mathbf{w}) = \langle \mathbf{g}_1, \mathbf{w}\rangle + \frac{1}{\eta}\|\mathbf{w}\|^2$ and $\ell_t(\mathbf{w}) = \langle \mathbf{g}_t, \mathbf{w}\rangle$ for $t = 2, \ldots, T$. By combining Lemma 3 with Equation (5), we have

$$\sum_{t=1}^{T} \langle \mathbf{g}_t, \hat{\mathbf{w}}_{t+1} - \mathbf{w}\rangle + \frac{\|\hat{\mathbf{w}}_2\|^2 - \|\mathbf{w}\|^2}{\eta} \leq \sum_{t=1}^{T} (\ell_t(\hat{\mathbf{w}}_{t+1}) - \ell_t(\mathbf{w})) \leq 0. \tag{6}$$

It is not hard to verify that

$$\begin{aligned} \sum_{t=1}^{T} \langle \mathbf{g}_t, \hat{\mathbf{w}}_t - \mathbf{w}\rangle &= \sum_{t=1}^{T} \langle \mathbf{g}_t, \hat{\mathbf{w}}_t - \hat{\mathbf{w}}_{t+1}\rangle + \sum_{t=1}^{T} \langle \mathbf{g}_t, \hat{\mathbf{w}}_{t+1} - \mathbf{w}\rangle \\ &\leq \sum_{t=1}^{T} \|\mathbf{g}_t\| \, \|\hat{\mathbf{w}}_t - \hat{\mathbf{w}}_{t+1}\| + \frac{\|\mathbf{w}\|^2 - \|\hat{\mathbf{w}}_2\|^2}{\eta} \\ &\leq \frac{\eta G^2 T}{2} + \frac{D^2}{\eta}, \end{aligned} \tag{7}$$

where the first inequality is due to Equation (6).

In the following, we derive the bound on $\mathbb{E}\left[\|\mathbf{w}_t - \hat{\mathbf{w}}_t\|\right]$, which is the compression error.

$$\mathbb{E}\left[\|\mathbf{w}_t - \hat{\mathbf{w}}_t\|\right] \leq \frac{\eta}{2}\left(\mathbb{E}\left[\left\|\sum_{k=1}^{t-1} \mathbf{g}_k - \mathbf{s}_k\right\|\right]\right) = \frac{\eta}{2}\left(\mathbb{E}\left[\|\mathbf{e}_t\|\right]\right),$$

where the first inequality is due to Lemma 4. To give the bound of the compression error, we introduce the following lemma.

Lemma 5 (Lemma 3 in Karimireddy et al. (2019)) *Under the Assumptions 1, 2 and 4, C-FTRL with any compressor satisfying Assumption 3 has the following bound on the expected squared norm of the error at each round t*

$$\mathbb{E}\left[\|\mathbf{e}_t\|^2\right] \leq \frac{4G^2}{\omega^2}.$$

To begin with, we derive the regret bound for C-FTRL.

$$\begin{aligned}\sum_{t=1}^{T}\mathbb{E}\left[\langle \mathbf{g}_t, \mathbf{w}_t - \mathbf{w}\rangle\right] &= \sum_{t=1}^{T}\mathbb{E}\left[\langle \mathbf{g}_t, \mathbf{w}_t - \hat{\mathbf{w}}_t + \hat{\mathbf{w}}_t - \mathbf{w}\rangle\right] \\ &= \sum_{t=1}^{T}\mathbb{E}\left[\langle \mathbf{g}_t, \mathbf{w}_t - \hat{\mathbf{w}}_t\rangle\right] + \sum_{t=1}^{T}\mathbb{E}\left[\langle \mathbf{g}_t, \hat{\mathbf{w}}_t - \mathbf{w}\rangle\right] \\ &\leq \sum_{t=1}^{T}\mathbb{E}\left[\langle \mathbf{g}_t, \hat{\mathbf{w}}_t - \mathbf{w}\rangle\right] + \sum_{t=1}^{T} G\mathbb{E}\left[\|\mathbf{w}_t - \hat{\mathbf{w}}_t\|\right] \\ &\leq \frac{\eta G^2 T}{2} + \frac{D^2}{\eta} + \frac{\eta T G^2}{\omega},\end{aligned}$$

where the last inequality is due to Equation (7) and Lemma 5.

By setting the learning rate $\eta = \frac{D}{G\sqrt{\omega^{-1}T}}$, we can obtain

$$\mathbb{E}\left[\sum_{t=1}^{T}\langle \mathbf{g}_t, \mathbf{w}_t - \mathbf{w}\rangle\right] \leq DG\sqrt{\omega^{-1}T} + \frac{3}{2}DG\sqrt{\omega^{-1}T} \leq 3DG\sqrt{\omega^{-1}T} = O(DG\sqrt{\omega^{-1}T}). \quad (8)$$

For the scaled sign compressor with $\omega = 1/d$, we can directly derive the regret bound in Theorem 1.

C.2 Proof of Theorem 2

The proof of Theorem 2 can be directly derived from the subsequent analysis for Theorem 4 by using the scaled sign compressor.

C.3 Proof of Theorem 4

As mentioned before, the O2NC conversion reduces the non-smooth non-convex optimization task to an online learning problem. Therefore, the objective becomes minimizing the K-shifting regret, defined as follows:

Definition 3 (K-shifting regret) *The regret over KT rounds with respect to an arbitrary sequence of K comparators $\mathbf{u}_1, \ldots, \mathbf{u}_K$ that changes every T iterations is called K-shifting regret, i.e.,*

$$\mathrm{REG}_T(\mathbf{u}_1, \ldots, \mathbf{u}_K) \triangleq \sum_{k=1}^{K}\underbrace{\sum_{m=(k-1)T+1}^{kT}\langle \mathbf{g}_m, \mathbf{w}_m - \mathbf{u}_k\rangle}_{\text{Static Regret}},$$

where $\mathbf{g}_m$ is the gradient in round m.

The K-shifting regret can be decomposed into the sum of K static regrets (Zinkevich, 2003). Since the classical OCO methods, e.g., OGD, can achieve the optimal regret bound of $O(DG\sqrt{T})$, a simple strategy of restarting the online algorithm every T iterations yields a K-shifting regret bound of $O(KDG\sqrt{T})$.

We first introduce a lemma to provide the guarantee of O2NC conversion.

Lemma 6 (Theorem 8 in Cutkosky et al. (2023)) *With the notation of Algorithm 1, we define $\mathbf{u}_k = -D\frac{\sum_{t=1}^{T}\nabla f(\mathbf{x}_{k,t})}{\left\|\sum_{t=1}^{T}\nabla f(\mathbf{x}_{k,t})\right\|}$ for some $D > 0$ for $k = 1, \ldots, K$. If $f(\cdot)$ satisfies Assumptions 2 and 1, we have*

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{T}\sum_{t=1}^{T}\nabla f(\mathbf{x}_{k,t})\right\|\right] \leq \frac{\Delta}{DM} + \frac{\mathbb{E}[\mathrm{REG}_T(\mathbf{u}_1, \ldots, \mathbf{u}_K)]}{DM} + \frac{G}{\sqrt{T}}.$$

Therefore, by using C-FTRL as the online algorithm and restarting C-FTRL every T rounds, we can ensure $\mathbb{E}[\mathrm{REG}_T(\mathbf{u}_1, \ldots, \mathbf{u}_K)] \leq 3KDG\sqrt{\omega^{-1}T}$ for any $\mathbf{u}_k$ satisfying $\|\mathbf{u}_k\| \leq D$. In the following, we will select the parameters K and D to obtain the complexity. We set $D = \delta/T$. As shown by Cutkosky et al. (2023), for all $m < m' \leq m + T - 1$, we have

$$\|\mathbf{x}_m - \mathbf{x}_{m'}\| \leq DT = \delta,$$

which further ensures $\|\mathbf{x}_{k,t} - \overline{\mathbf{x}}_k\| \leq \delta$.

Suppose we have a budget of N gradient queries. By setting $D = \delta/T, K = \lfloor N/T \rfloor, T = \min\left\{\left\lceil \left(\frac{G\delta N}{\omega^{1/2}\Delta}\right)^{2/3}\right\rceil, \frac{N}{2}\right\}$, we can derive

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{T}\sum_{t=1}^{T}\nabla f(\mathbf{x}_{k,t})\right\|\right] \leq & 2\frac{\Delta}{DN} + \frac{6KDG\sqrt{\omega^{-1}T}}{DN} + \frac{G}{\sqrt{T}} \\
\leq & \frac{2T\Delta}{\delta N} + \frac{7G}{\sqrt{\omega T}} \\
\leq & \max\left\{\frac{14G^{2/3}\Delta^{1/3}}{\omega^{1/3}(\delta N)^{1/3}}, \frac{10G}{\sqrt{\omega N}}\right\} + \frac{2\Delta}{\delta N},
\end{aligned}$$

where the first step follows from $N \leq 2KT \leq 2M$. When N is sufficiently large, we have

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{T}\sum_{t=1}^{T}\nabla f(\mathbf{x}_{k,t})\right\|\right] \leq O\left(\omega^{-1/3}\delta^{-1/3}N^{-1/3}\right).$$

By adopting the scaled sign compressor with $\omega = 1/d$, we can derive Theorem 2.

C.4 Proof of Theorem 3

The proof of Theorem 3 can be directly derived from the subsequent analysis for Theorem 5 by using the scaled sign compressor.

C.5 Proof of Theorem 5

Firstly, we introduce a lemma to derive the bound of the gradient estimator.

Lemma 7 (Lemma D.1 in Lin et al. (2022)) *For any $\hat{\mathbf{g}}_t$ generated in Algorithm 3, we have*

$$\mathbb{E}[\|\hat{\mathbf{g}}_t\|^2] \leq 16\sqrt{2\pi}dL_0^2.$$

Let $\hat{\mathbf{g}}_m = \frac{1}{R}\sum_{s=1}^{R}\hat{\mathbf{g}}_{m,s}$, where the estimators $\hat{\mathbf{g}}_{m,s}$ are generated independently using Algorithm 3. Then, we have

$$\mathbb{E}[\|\hat{\mathbf{g}}_m\|^2] \leq \left(\frac{d}{R} + 1\right)16\sqrt{2\pi}L_0^2.$$

By applying Lemma 7 with $R \geq d$, we obtain the following bound for the gradient estimator $\hat{\mathbf{g}}_m$.

$$\mathbb{E}[\|\hat{\mathbf{g}}_m\|^2] \leq 32\sqrt{2\pi}L_0^2 \leq 81L_0^2. \tag{9}$$

By using Lemma 2 and setting $R = \max\{d, \lceil 1/\omega \rceil\}$, we can ensure

$$\mathbb{E}\left[\|\mathrm{FCC}(\mathbf{x}, \mathcal{C}(\cdot), R) - \mathbf{x}\|^2\right] \leq (1-\omega)^R\|\mathbf{x}\|^2 \leq \frac{1}{e}\|\mathbf{x}\|^2. \tag{10}$$

Therefore, FCC in Algorithm 5 can be viewed as a compressor with quality factor $\omega' = 1 - 1/e$.

Then, by using Equation (9) and Equation (10) and setting the learning rate $\eta = \frac{D\sqrt{(1-1/e)}}{9L_0\sqrt{T}}$, for Algorithm 2, we can ensure

$$\mathbb{E}\left[\sum_{t=1}^{T}\langle\hat{\mathbf{g}}_t, \mathbf{w}_t - \mathbf{w}\rangle\right] \leq 34DL_0\sqrt{T}. \tag{11}$$

Suppose we have a budget of $2N$ function value evaluations. We set $\mathbf{u}_k = -D\frac{\sum_{t=1}^{T}\nabla f_\rho(\mathbf{x}_{k,t})}{\left\|\sum_{t=1}^{T}\nabla f_\rho(\mathbf{x}_{k,t})\right\|}$ for some $D>0$ for $k=1,\ldots,K$, $D=\nu/T, K=\lfloor N/(RT)\rfloor, T=\min\left\{\left\lceil\left(\frac{L_0\nu N}{R\Delta_\rho}\right)^{2/3}\right\rceil, \frac{N}{2R}\right\}$ and $R=\max\{d,\lceil 1/\omega\rceil\}$. If $f(\cdot)$ satisfies Assumptions 2 and 1, and the compressor satisfies Assumption 3, ReC-O2NC can ensure

$$\begin{aligned}\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\nabla f_\rho(\overline{\mathbf{x}}_k)\right\|_\nu\right] \leq& 4\frac{\Delta_\rho R}{DN}+\frac{34KDL_0R\sqrt{T}}{DN}+\frac{9L_0}{\sqrt{T}}\\ \leq& \frac{4TR\Delta}{\nu N}+\frac{45L_0}{\sqrt{T}}\\ \leq& \max\left\{\frac{49R^{1/3}L_0^{2/3}\Delta^{1/3}}{(\nu N)^{1/3}}, \frac{90\sqrt{R}L_0}{\sqrt{N}}\right\}+\frac{2\Delta}{\nu N}\\ \leq& O(R^{1/3}\nu^{-1/3}N^{-1/3}).\end{aligned}$$

We establish the guarantee of finding (ν,ϵ)-stationary points of the function $f_\rho(\cdot)$. To derive the guarantee for the loss function $f(\cdot)$, we introduce the following lemma.

Lemma 8 (Lemma 4 in Kornowski & Shamir (2024)) *If $\mathbf{x}$ is a (ν,ϵ)-stationary point of $f_\rho(\cdot)$, then it is a $(\nu+\rho,\epsilon)$-stationary point of $f(\cdot)$.*

According to Lemma 8, if we can ensure $\nu+\rho\leq\delta$, $\mathbf{x}$ is also a (δ,ϵ)-stationary point of $f(\cdot)$. Thus, by setting $\rho=\min\left\{\frac{\delta}{2}, \frac{f_\rho(\mathbf{x}_0)-f_\rho^*}{L_0}\right\}, \nu=\delta-\rho$, we can derive

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\nabla f(\overline{\mathbf{x}}_k)\right\|_\delta\right]\leq O(R^{1/3}\delta^{-1/3}N^{-1/3}).$$

By adopting the scaled sign compressor with $\omega=1/d$, we can derive Theorem 3.

C.6 Proof of Theorem 6

We first define $\mathbf{g}_t=\frac{1}{n}\sum_{i=1}^{n}\mathbf{g}_t^i$ as the stochastic gradient of the global loss function and further define the global K-shifting regret:

$$\mathrm{REG}_T^n(\mathbf{u}_1,\ldots,\mathbf{u}_K)\triangleq\frac{1}{n}\sum_{i=1}^{n}\sum_{k=1}^{K}\sum_{m=(k-1)T+1}^{kT}\left\langle\mathbf{g}_m^i,\mathbf{w}_m-\mathbf{u}_k\right\rangle=\sum_{k=1}^{K}\sum_{m=(k-1)T+1}^{kT}\left\langle\mathbf{g}_m,\mathbf{w}_m-\mathbf{u}_k\right\rangle.$$

Then, we introduce a lemma to give the regret bound of D-FTCL.

Lemma 9 (Theorem 4.1 in Yang et al. (2026)) *Under the Assumptions 2 and 4, for compressors satisfying Assumption 3 with quality factor ω, by setting the learning rate $\eta=\frac{\omega D}{2G\sqrt{T}}$, D-FTCL can ensure*

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{t=1}^{T}\langle\mathbf{g}_t^i,\mathbf{w}_t-\mathbf{w}\rangle\leq 6DG\omega^{-1}\sqrt{T}\leq O(DG\omega^{-1}\sqrt{T}),$$

for all $\mathbf{w}\in\mathcal{W}$.

Since FCC in Algorithm 5 can be viewed as a compressor with quality factor $\omega'=1-1/e$, we can ensure

$$\frac{1}{n}\sum_{i=1}^{n}\sum_{t=1}^{T}\langle\mathbf{g}_t^i,\mathbf{w}_t-\mathbf{w}\rangle\leq 12DG\sqrt{T}\leq O(DG\sqrt{T}).$$

Here, the T online-learning iterations require RT communication rounds. Restarting the online algorithm every T online-learning iterations therefore yields

$$\sum_{k=1}^{K}\sum_{m=(k-1)T+1}^{kT}\langle\mathbf{g}_m,\mathbf{w}_m-\mathbf{u}_k\rangle\leq 12DGK\sqrt{T}\leq O(DGK\sqrt{T}).$$

By using Lemma 6, we have

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{T}\sum_{t=1}^{T}\nabla f(\mathbf{x}_{k,t})\right\|\right] \leq \frac{f(\mathbf{x}_0)-f^*}{DM} + \frac{\mathbb{E}[\mathrm{REG}_T^n(\mathbf{u}_1,\ldots,\mathbf{u}_K)]}{DM} + \frac{G}{\sqrt{T}}. \tag{12}$$

Furthermore, we define $\mathbf{u}_k = -D\frac{\sum_{t=1}^{T}\nabla f(\mathbf{x}_{k,t})}{\left\|\sum_{t=1}^{T}\nabla f(\mathbf{x}_{k,t})\right\|}$ for some $D>0$ for $k=1,\ldots,K$, $D=\delta/T, K=\lfloor N/(RT)\rfloor, T=\min\left\{\left\lceil\left(\frac{G\delta N}{R\Delta}\right)^{2/3}\right\rceil, \frac{N}{2R}\right\}$, $R=\lceil\omega^{-1}\rceil$. Here, $N=RKT$ denotes the total number of gradient queries for each worker. We then obtain

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \leq& \frac{2\Delta}{DM} + \frac{12KGD\sqrt{T}}{DM} + \frac{G}{\sqrt{T}} \\
\leq& \frac{2\Delta R}{DN} + \frac{12KGDR\sqrt{T}}{DN} + \frac{G}{\sqrt{T}} \\
\leq& \frac{2TR\Delta}{\delta N} + \frac{13G}{\sqrt{T}} \\
\leq& \frac{2\Delta}{\delta N} + \max\left\{\frac{15R^{1/3}G^{2/3}\Delta^{1/3}}{(\delta N)^{1/3}}, \frac{26\sqrt{R}G}{\sqrt{N}}\right\}.
\end{aligned}$$

When N is sufficiently large, we have

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \leq O\left(\omega^{-1/3}\delta^{-1/3}N^{-1/3}\right).$$

C.7 Proof of Theorem 7

It is not hard to verify that using the second-moment bound defined in Equation (9) and FCC, D-FTCL achieves a regret of $R(T)\leq 108DL_0\sqrt{T}=O(DL_0\sqrt{T})$. Based on Equation (12), we can derive

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{T}\sum_{t=1}^{T}\nabla f_\rho(\mathbf{x}_{k,t})\right\|\right] \leq \frac{f_\rho(\mathbf{x}_0)-f_\rho^*}{DM} + \frac{108KL_0D\sqrt{T}}{DM} + \frac{9L_0}{\sqrt{T}}. \tag{13}$$

We define $\mathbf{u}_k = -D\frac{\sum_{t=1}^{T}\nabla f_\rho(\mathbf{x}_{k,t})}{\left\|\sum_{t=1}^{T}\nabla f_\rho(\mathbf{x}_{k,t})\right\|}$ for some $D>0$ for $k=1,\ldots,K$, and set $D=\nu/T, K=\lfloor N/(RT)\rfloor, T=\min\left\{\left\lceil\left(\frac{L_0\nu N}{R\Delta_\rho}\right)^{2/3}\right\rceil, \frac{N}{2R}\right\}$, $R=\max\{d,\lceil\omega^{-1}\rceil\}$. Let $N=RKT$ be the total number of gradient queries for each worker. We have

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f_\rho(\overline{\mathbf{x}}_k)\|_\nu\right] \leq& \frac{2\Delta_\rho}{DM} + \frac{108KL_0D\sqrt{T}}{DM} + \frac{9L_0}{\sqrt{T}} \\
\leq& \frac{2\Delta_\rho R}{DN} + \frac{108KL_0DR\sqrt{T}}{DN} + \frac{9L_0}{\sqrt{T}} \\
\leq& \frac{2TR\Delta_\rho}{\nu N} + \frac{117L_0}{\sqrt{T}} \\
\leq& \frac{2\Delta_\rho}{\nu N} + \max\left\{\frac{119R^{1/3}L_0^{2/3}\Delta_\rho^{1/3}}{(\nu N)^{1/3}}, \frac{234\sqrt{R}L_0}{\sqrt{N}}\right\}.
\end{aligned}$$

As in the proof of Theorem 3, we set $\rho=\min\left\{\frac{\delta}{2},\frac{\Delta}{L_0}\right\}, \nu=\delta-\rho, D=\nu/T, T=\min\left\{\left\lceil\left(\frac{L_0\nu N}{R(\Delta+\rho L_0)}\right)^{2/3}\right\rceil, \frac{N}{2R}\right\}$. With $N=RM$, we can ensure

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \leq O\left(R^{1/3}\delta^{-1/3}N^{-1/3}\right) \leq O\left(\max\{d^{1/3},\omega^{-1/3}\}\delta^{-1/3}N^{-1/3}\right),$$

which completes the proof.

C.8 Proof of Single-Sample Variant

If we adopt the single-sample estimator with one compression step, by setting $\eta = \frac{D}{4d(2\pi)^{1/4}L_0\sqrt{T}}$ we can derive

$$\sum_{t=1}^{T}\langle \mathbf{g}_t, \mathbf{w}_t - \mathbf{w}\rangle \leq 8d(2\pi)^{1/4}DL_0\sqrt{T}.$$

Thus, by setting $D = \nu/T, K = \lfloor N/T \rfloor, T = \min\left\{\left\lceil\left(\frac{dL_0\nu N}{\Delta_\rho}\right)^{2/3}\right\rceil, \frac{N}{2}\right\}$, we have

$$\begin{aligned}
\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{T}\sum_{t=1}^{T}\nabla f_\rho(\mathbf{x}_{k,t})\right\|\right] \leq & 2\frac{\Delta_\rho}{DN} + \frac{8d(2\pi)^{1/4}KDL_0\sqrt{T}}{DN} + \frac{4(2\pi)^{1/4}d^{1/2}L_0}{\sqrt{T}} \\
\leq & 2\frac{\Delta_\rho}{DN} + \frac{14dKDL_0\sqrt{T}}{DN} + \frac{7d^{1/2}L_0}{\sqrt{T}} \\
\leq & \frac{2T\Delta_\rho}{\nu N} + \frac{21dL_0}{\sqrt{T}} \\
\leq & \max\left\{\frac{23d^{2/3}L_0^{2/3}\Delta_\rho^{1/3}}{(\nu N)^{1/3}}, \frac{42d^{2/3}L_0}{\sqrt{N}}\right\} + \frac{2\Delta_\rho}{\nu N} \\
\leq & O\left(d^{2/3}\delta^{-1/3}N^{-1/3}\right).
\end{aligned}$$

Furthermore, by setting $\rho = \min\left\{\frac{\delta}{2}, \frac{f_\rho(\mathbf{x}_0) - f_\rho^*}{L_0}\right\}, \nu = \delta - \rho$, we can derive

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\|\nabla f(\overline{\mathbf{x}}_k)\|_\delta\right] \leq O(d^{2/3}\delta^{-1/3}N^{-1/3}).$$

D Experiments

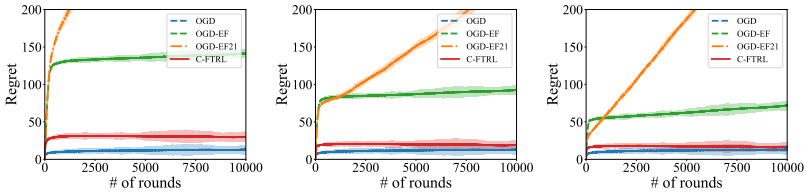
In this section, we empirically evaluate the proposed methods on three tasks. All experiments are conducted on a server equipped with 8 NVIDIA RTX 4090 GPUs and an Intel Xeon Platinum 8358P CPU running at 2.60 GHz.

D.1 Online Classification

Experimental Setup. We first evaluate C-FTRL on an online classification task. We use the ijcnn1 dataset from LIBSVM (Chang & Lin, 2011). In each round $t \in [T]$, a batch of training examples $\{(\mathbf{z}_{t,1}, y_{t,1}), \ldots, (\mathbf{z}_{t,m}, y_{t,m})\}$ arrives, where $(\mathbf{z}_{t,k}, y_{t,k}) \in [-1,1]^d \times \{-1,1\}, k = 1, \ldots, m$. The online learner selects a linear model $\mathbf{w}_t$ and incurs the non-smooth hinge loss

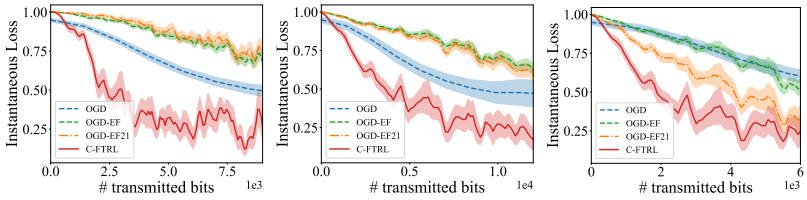
$$f_t(\mathbf{w}) = \frac{1}{m}\sum_{k=1}^{m}\max\{0, 1 - y_{t,k}\mathbf{w}^\top \mathbf{z}_{t,k}\},$$

where m is the batch size. We compare our C-FTRL with uncompressed OGD (Zinkevich, 2003), OGD with EF (OGD-EF) (Karimireddy et al., 2019) and OGD with EF21 (OGD-EF21) (Richtárik et al., 2021). In our experiments, we set the number of rounds to $T = 10000$ and the domain radius to $D = 2.5$. We conduct the experiments with 3 types of compressors: Rand-K with $\omega = 0.05$, Top-K with $\omega = 0.1$ and Sign with $\omega = 1/d$. Each experiment is repeated 5 times and the average



(a) Rand-K with $\omega = 0.05$ (b) Top-K with $\omega = 0.1$ (c) Sign with $\omega = 1/d$

Figure 1: Regret on the ijcnn1 dataset in the first-order setting.



(a) Rand-K with $\omega = 0.05$ (b) Top-K with $\omega = 0.1$ (c) Sign with $\omega = 1/d$

Figure 2: Instantaneous loss on the ijcnn1 dataset in the first-order setting.

performance is reported. We plot the regret versus the iterations and the instantaneous loss versus the transmitted bits.

Results. As demonstrated in Figures 1, 2, 3 and 4, although OGD-EF and OGD-EF21 converge, they incur higher regret due to the accumulation of projection errors. In contrast, C-FTRL achieves comparable performance to uncompressed OGD and outperforms the compressed baselines under the same communication budget.

D.2 Non-Convex Penalized SVM

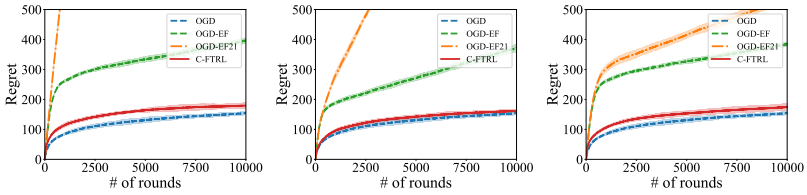
We further evaluate our methods on a non-convex penalized SVM task under the zeroth-order setting, in both single-node and distributed scenarios.

Experiment Setup. We conduct experiments on the a9a dataset from LIBSVM (Chang & Lin, 2011). Given a binary classification dataset $\{(\mathbf{z}_k, y_k)\}_{k=1}^m$ with $\mathbf{z}_k \in \mathbb{R}^d$ and $y_k \in \{-1, 1\}$, we consider the objective

$$\min_{\mathbf{x}\in\mathbb{R}^d} f(\mathbf{x}) := \frac{1}{m}\sum_{k=1}^{m} \max\{1 - y_k \mathbf{z}_k^\top \mathbf{x}, 0\} + \lambda \sum_{j=1}^{d} \min\{|\mathbf{x}_j|, \alpha\},$$

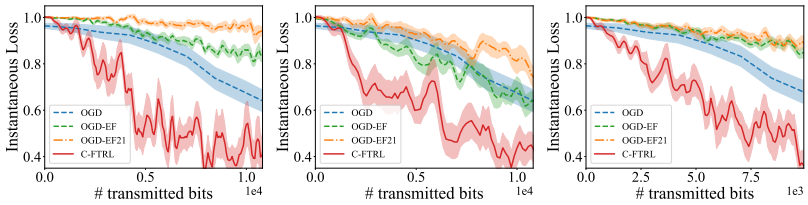
where $\lambda, \alpha > 0$ are hyperparameters. The hinge loss makes the problem non-smooth, while the capped-ℓ_1 regularizer introduces nonconvexity. Following a standard setup (Chen et al., 2026), we set $\lambda = 10^{-5}/m$, $\alpha = 2$ and the batch size to 32. For the distributed setting, we use $n = 8$ workers and partition the samples evenly across them. In the single-node setting, we compare ReC-O2NC with zeroth-order O2NC (Kornowski & Shamir, 2024), O2NC with EF21 (O2NC-EF21) (Fatkhullin et al., 2025) and ReC-O2NC with $R = 1$ (O2NC-C). In the distributed setting, we compare DistReC-O2NC with distributed zeroth-order O2NC (Kornowski & Shamir, 2024), distributed O2NC with EF21 (D-O2NC-EF21) (Fatkhullin et al., 2025) and DistReC-O2NC with $R = 1$. We consider two compressors, Rand-K with $\omega = 0.1$ and Sign with $\omega = 1/d$. We tune the smoothing radius $\rho \in \{10^{-2}, 10^{-1}\}$, the learning rate $\eta \in \{10^{-1}, 5\times10^{-2}, 10^{-2}, 5\times10^{-3}, 10^{-3}\}$ and $D \in \{10^{-3}, 10^{-2}, 5\times10^{-2}, 10^{-1}\}$. Since the theoretical choice of the number of compression rounds R is derived from a worst-case analysis, we further tune it over $\{\lceil 1/(10\omega)\rceil, \lceil 1/(6\omega)\rceil, \lceil 1/(4\omega)\rceil, \lceil 1/\omega\rceil\}$. Each experiment is repeated 5 times, and we report the average loss against the iterations.

Results. As shown in Figures 5 and 6, ReC-O2NC and DistReC-O2NC perform comparably to their respective uncompressed baselines and outperform the corresponding EF21-based methods.



(a) Rand-K with $\omega = 0.05$ (b) Top-K with $\omega = 0.1$ (c) Sign with $\omega = 1/d$

Figure 3: Regret on the ijcnn1 dataset in the zeroth-order setting.



(a) Rand-K with $\omega = 0.05$ (b) Top-K with $\omega = 0.1$ (c) Sign with $\omega = 1/d$

Figure 4: Instantaneous loss on the ijcnn1 dataset in the zeroth-order setting.

Moreover, our ReC-O2NC and DistReC-O2NC have better performance than their respective $R = 1$ variants, benefiting from our compression-estimation decoupling strategy.

D.3 Image Classification

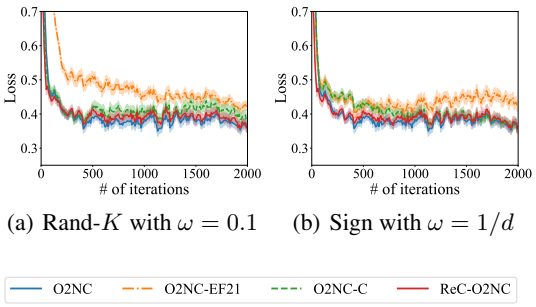
We finally evaluate our first-order methods on the CIFAR-10 image classification task in both single-node and distributed settings.

Experimental Setup. In the single-node setting, we train a ResNet18 model (He et al., 2016) on the CIFAR-10 dataset and compare C-O2NC with uncompressed O2NC (Cutkosky et al., 2023) and O2NC with EF21 (O2NC-EF21) (Fatkhullin et al., 2025). For the distributed setting, we further train ResNet18 in a multi-node environment with 8 and 16 workers, and compare our DistC-O2NC with uncompressed distributed O2NC (Cutkosky et al., 2023), O2NC with EF21 (D-O2NC-EF21) (Fatkhullin et al., 2025), and DistC-O2NC with $R = 1$. The learning rate η and the radius D are tuned over $\{2 \times 10^{-3}, 2.5 \times 10^{-3}, 3 \times 10^{-3}, 4 \times 10^{-3}\}$ and $\{2 \times 10^{-2}, 2.5 \times 10^{-2}, 3 \times 10^{-2}, 4 \times 10^{-2}\}$, respectively. We tune the number of compression rounds R over $\{2, 3\}$, following Huang et al. (2022). A weight decay of 5×10^{-4} and batch size of 128 are applied uniformly across all algorithms. Each experiment is repeated 5 times.

Results. Single-node results are presented in Figures 7 and 8. Under the Rand-K compressor, C-O2NC outperforms O2NC-EF21 in training accuracy and achieves performance comparable to the uncompressed O2NC. Under the sign compressor, while O2NC-EF21 attains a training loss similar to that of C-O2NC, its test accuracy is lower. The distributed results are summarized in Figures 9, 10, 11, and 12. The training loss of DistC-O2NC decreases rapidly, and its test accuracy surpasses that of D-O2NC-EF21 under both the 8-node and 16-node configurations. Furthermore, DistC-O2NC outperforms its $R = 1$ variant, validating the benefit of recursive compression.

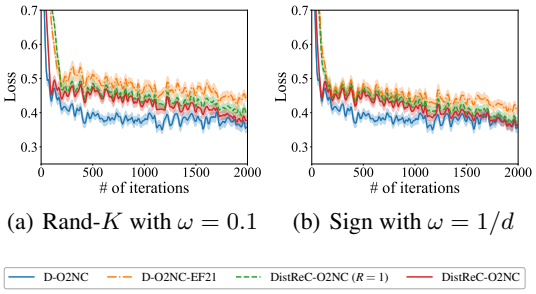
D.4 Sensitivity to Batch Size

In this section, we report additional experiments on the CIFAR-10 dataset to assess the sensitivity of the proposed method to the batch size. We evaluate four configurations drawn from the set $\{64, 128, 256, 512\}$, with the corresponding results displayed in Figure 13. The algorithm converges without requiring large batches and remains robust across the tested range.



(a) Rand-K with $\omega = 0.1$ (b) Sign with $\omega = 1/d$

Figure 5: Results on the a9a dataset in the single-node setting.



(a) Rand-K with $\omega = 0.1$ (b) Sign with $\omega = 1/d$

Figure 6: Results on the a9a dataset in the distributed setting.

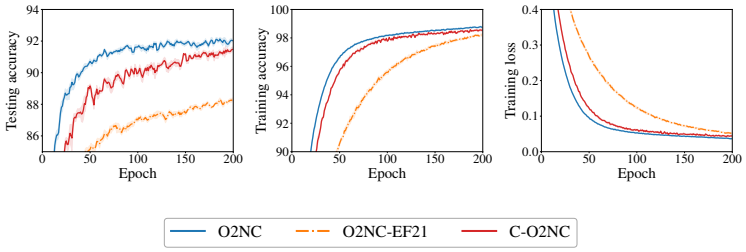


Figure 7: Results on the CIFAR-10 dataset with the Rand-K compressor in the single-node setting.

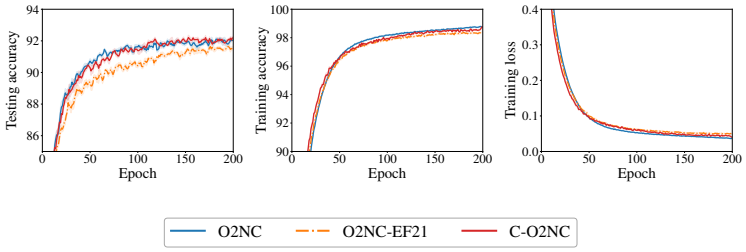


Figure 8: Results on the CIFAR-10 dataset with the sign compressor in the single-node setting.

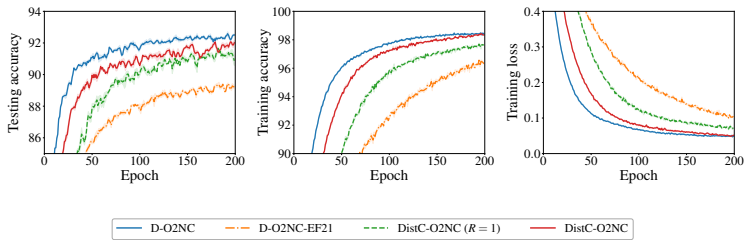


Figure 9: Results on the CIFAR-10 dataset with the Rand-K compressor in the 8-node distributed setting.

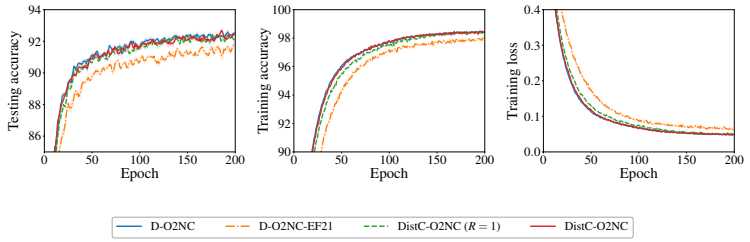


Figure 10: Results on the CIFAR-10 dataset with the sign compressor in the 8-node distributed setting.

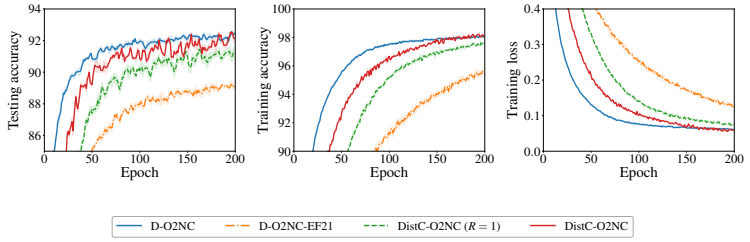


Figure 11: Results on the CIFAR-10 dataset with the Rand-K compressor in the 16-node distributed setting.

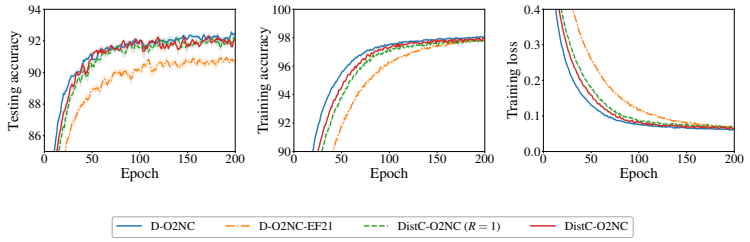


Figure 12: Results on the CIFAR-10 dataset with the sign compressor in the 16-node distributed setting.

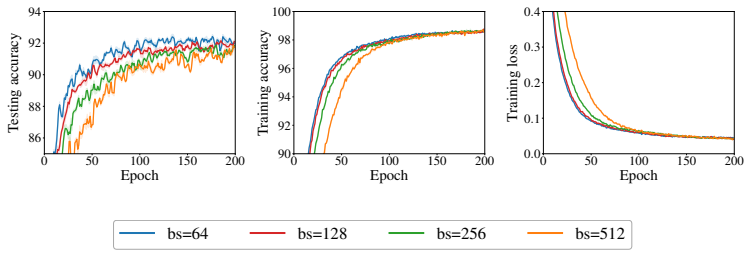


Figure 13: Results on the CIFAR-10 dataset with different batch sizes.