An Online Optimization Perspective on First-Order and Zero-Order Decentralized Nonsmooth Nonconvex Stochastic Optimization

Emre Sahinoglu [1] **Shahin Shahrampour** [1]

Abstract

We investigate the finite-time analysis of finding (δ, ϵ)-stationary points for nonsmooth nonconvex objectives in decentralized stochastic optimization. A set of agents aim at minimizing a global function using only their local information by interacting over a network. We present a novel algorithm, called Multi Epoch Decentralized Online Learning (ME-DOL), for which we establish the sample complexity in various settings. First, using a recently proposed online-to-nonconvex technique, we show that our algorithm recovers the optimal convergence rate of smooth nonconvex objectives. We then extend our analysis to the nonsmooth setting, building on properties of randomized smoothing and Goldstein-subdifferential sets. We establish the sample complexity of $O(\delta^{-1}\epsilon^{-3})$, which to the best of our knowledge is the first finite-time guarantee for decentralized nonsmooth nonconvex stochastic optimization in the first-order setting (without weak-convexity), matching its optimal centralized counterpart. We further prove the same rate for the zero-order oracle setting without using variance reduction.

1. Introduction

At the heart of many practical machine learning problems, we must deal with nonconvex optimization of nonsmooth objective functions. Examples include training neural networks with ReLU activation functions, blind deconvolution, sparse dictionary learning, and robust phase retrieval. Despite the significant practical success of such schemes, the vast majority of prior work in theoretical analysis of nonsmooth nonconvex optimization focused on asymptotic convergence results (Rockafellar & Wets, 2009; Clarke et al., 2008). More recently, *finite-time* analysis of this class of problems has attracted significant attention (Jordan et al., 2023; Majewski et al., 2018; Davis & Drusvyatskiy, 2019; Daniilidis & Drusvyatskiy, 2020; Tian et al., 2022).

On the other hand, decentralization is a crucial mechanism to scale up optimization problems. For nonsmooth objectives, though the class of convex problems is well-understood in decentralized optimization (Nedic & Ozdaglar, 2009; Scaman et al., 2018), the characterization of optimal finite-time rates for nonconvex problems has remained elusive (except for weakly-convex problems (Chen et al., 2021)). In the present work, we address the *finite-time* analysis of *decentralized nonsmooth nonconvex stochastic* optimization.

We consider a decentralized optimization problem where a group of n agents aim at minimizing a global function. However, each agent has limited information about this global objective and interacts with its neighbors to solve the global problem, formulated in the following form

$$\min_{x\in\mathbb{R}^d} \Big\{ f(x) = \frac{1}{n}\sum_{i=1}^{n} f^i(x) \Big\}. \tag{1}$$

Local functions f^i are in the form of $f^i(x) = \mathbb{E}_{\xi_i\sim\mathcal{D}_i}[F^i(x,\xi_i)]$, where $F^i(x,\xi_i)$ are stochastic with random index ξ_i, and ξ_i corresponds to a data sample from local dataset of agent i. We assume that the local functions are nonconvex and Lipschitz continuous but do *not* necessarily have Lipschitz continuous gradients, i.e., they are nonsmooth.

In an optimization problem, a tractable optimality criterion is required for finite-time convergence guarantees. For nonsmooth nonconvex objectives, ϵ-stationarity cannot be guaranteed in finite time (Kornowski & Shamir, 2021; Zhang et al., 2020b). Instead, the notion of (δ, ϵ)-stationarity is a tractable criterion (Zhang et al., 2020b), where we seek vectors with norm less than ϵ among the convex hull of the subdifferential set of a ball with radius δ (see Definition 2 for exact mathematical definition). The goal of this paper is to identify a (δ, ϵ)-stationary point of the global function f when agents have access to either the first-order oracle (i.e., $\nabla F^i(\cdot, \xi_i)$) or the zero-order oracle (i.e., $F^i(\cdot, \xi_i)$).

[1]Department of Mechanical and Industrial Engineering, Northeastern University, Boston, MA 02115. Correspondence to: Emre Sahinoglu <sahinoglu.m@northeastern.edu>, Shahin Shahrampour <s.shahrampour@northeastern.edu>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

1.1. Contributions

In this paper, we address the finite-time analysis of decentralized nonsmooth nonconvex stochastic optimization. We present a novel algorithm, called **M**ulti **E**poch **D**ecentralized **O**nline **L**earning (ME-DOL), for which we establish the sample complexity in various settings. Our contributions are three-fold.

- We adopt the online-to-nonconvex conversion technique of Cutkosky et al. (2023) to streamline the finite-time analysis of ME-DOL for decentralized nonsmooth nonconvex optimization. First, for smooth objectives, we prove that the complexity of finding (δ, ϵ)-stationary points is $O(\delta^{-1}\epsilon^{-3})$ (Theorem 1). This rate implies the optimal complexity of $O(\epsilon^{-4})$ for finding ϵ-stationary points of smooth objectives (Arjevani et al., 2023; Lu & De Sa, 2021).
- For nonsmooth stochastic optimization with first-order oracles, ME-DOL achieves the same complexity, $O(\delta^{-1}\epsilon^{-3})$, matching its centralized counterpart (Theorem 2). To the best of our knowledge, this is the *first* finite-time guarantee for decentralized nonsmooth nonconvex stochastic optimization in the first-order oracle setting. Prior to our work, Chen et al. (2021) provided finite-time guarantees only on the Moreau Envelope of *weakly-convex* functions.
- For the zero-order oracle setting, ME-DOL achieves the best known complexity result in terms of δ and ϵ, i.e., $O(\delta^{-1}\epsilon^{-3})$ (Theorem 3), which also matches its centralized zero-order counterpart (Lin et al., 2022). In the decentralized setting, this rate was previously achieved only with the variance reduction mechanism (Lin et al., 2024).

1.2. Highlights of Technical Analysis

Randomized Smoothing. Finite-time analysis of nonsmooth objectives is mainly based on smooth approximations of these objectives. Randomized Smoothing (RS) and Moreau Envelope (ME) are the most common approximation methods. In ME, the original function is approximated with $f_\mu^{ME}(x) = \min_{y\in\mathbb{R}^d}\left\{f(y) + \frac{1}{2\mu}\|y-x\|^2\right\}$ (Davis & Drusvyatskiy, 2019; Scaman et al., 2018; 2020). ME approximation provides theoretical guarantees when applied on structured objectives with regularizer or used with weak-convexity assumption (Davis & Drusvyatskiy, 2019), but it might be practically unrealistic in some applications that use ReLU neural networks and ρ-margin SVMs (Tian et al., 2022). On the other hand, in RS the original function f is approximated by $f_\delta^{RS}(x) = \mathbb{E}[f(x+\delta u)]$, where u comes from a Gaussian distribution or a uniform distribution on the unit ball. In RS, the smoothness parameter depends on the ambient dimension as $\sqrt{d}$, but it provides favorable theoretical properties (see Proposition 1). Specifically, finding a (δ, ϵ)-stationary point of a nonsmooth L-Lipschitz function f can be pursued via finding an ϵ-stationary point of f_δ with a δL approximation error. RS does not require additional assumptions beyond Lipschitz continuity of the function, and this makes RS more tractable in practical applications. In this work, we develop our algorithm based on RS.

Nonconvex to Online Conversion. Another important technique in nonsmooth optimization is a reduction from nonsmooth nonconvex optimization to online learning (Cutkosky et al., 2023). In its original form, the method works on centralized optimization, where an online algorithm runs for a certain period, a candidate point is generated, and at the end of the period the online algorithm is restarted. The implementation of the online algorithm can be written explicitly in the form of gradient clipping (Kornowski & Shamir, 2024). From a technical perspective, the use of regret bounds in online learning streamlines the complexity analysis in the nonsmooth optimization. In our paper, we utilize decentralized online algorithm of Shahrampour & Jadbabaie (2018) to address decentralized nonsmooth nonconvex optimization. Compared to the centralized problem (Cutkosky et al., 2023), in the decentralized setting, the discrepancy between local variables and global variables makes the analysis more challenging.

Geometric Lemma of (Kornowski & Shamir, 2024). The optimality criterion for nonsmooth analysis is constructed on the Goldstein subdifferential set. The technical result of Kornowski & Shamir (2024), which links the Goldstein subdifferential set of f_δ to $f_{\delta+\mu}$ for $\mu, \delta > 0$, plays an important role in our analysis. Basically, for the goal of finding a (δ, ϵ)-stationary point of f, we can use a proportion of δ, namely $a\delta$ (for $0 < a < 1$), for smoothing and use the rest of the budget to identify a $((1-a)\delta, \epsilon)$-stationary point of the smoothed function $f_{a\delta}$ with the smoothness parameter $L_1 = O(a^{-1}\delta^{-1})$. This approach allows us to efficiently control the discrepancy terms.

Remark 1. In decentralized nonsmooth nonconvex optimization, our goal is to find a (δ, ϵ)-stationary point of global function f with randomized smoothing using partial information. For smooth objectives, the complexity of finding ϵ-stationary points in terms of the smoothness parameter L_1 and ϵ is $O(L_1\epsilon^{-4})$ (Lu & De Sa, 2021). For the nonsmooth objectives, a straightforward application of randomized smoothing with $L_1 = O(\delta^{-1})$ leads to the overall complexity of $O(\delta^{-1}\epsilon^{-4})$, which is sub-optimal. To improve this rate, we develop a technique inspired by Cutkosky et al. (2023) and based on decentralized online learning, and we obtain a finite-time bound, in which some of the terms depend on L_1. With the geometric lemma of Kornowski & Shamir (2024) we control the complexity of L_1-dependent terms. As a result, we obtain the same complexity rate (up

to constant factors) for decentralized nonsmooth nonconvex stochastic optimization as previously achieved in the centralized setting (Cutkosky et al., 2023).

1.3. Literature Review

Nonconvex optimization is well-studied under Lipschitz-smoothness assumption. In this setting, the goal is to find an ϵ-stationary point x satisfying $\|\nabla f(x)\| \leq \epsilon$. For the deterministic setting, it is well-known that gradient descent (GD) achieves a $O(\epsilon^{-2})$ sample complexity, and this rate is optimal (Carmon et al., 2020). In the stochastic setting, SGD achieves $O(\epsilon^{-4})$ rate with the assumption of unbiased, bounded variance gradients (Ghadimi & Lan, 2013). This rate is also optimal as shown by Arjevani et al. (2023). We now discuss several strands of related literature.

Nonsmooth Nonconvex Optimization. The first non-asymptotic analysis for nonsmooth nonconvex objectives was provided by Zhang et al. (2020b), showing that finding ϵ-stationary points in finite time is impossible. Furthermore, it was proved by Kornowski & Shamir (2021) that obtaining a near ϵ-stationary point is also impossible in nonsmooth optimization. Therefore, the goal of finding (δ, ϵ)-stationary points considered in Zhang et al. (2020b) is a tractable optimality criterion for nonsmooth objectives.

First-Order Nonsmooth Nonconvex Setting. (δ, ϵ)-Goldstein stationarity of nonsmooth objectives has been analyzed in various settings (Tian et al., 2022). Davis et al. (2022) showed that $\tilde{O}(\delta^{-1}\epsilon^{-3})$ can be achieved for Lipschitz continuous objectives when function values and gradients can be evaluated at points of differentiability. More recently, Cutkosky et al. (2023) proved that the $O(\delta^{-1}\epsilon^{-3})$ sample complexity is optimal in the stochastic first-order setting.

Zero-Order Nonsmooth Nonconvex Setting. Another line of work focuses on zero-order setting in nonsmooth nonconvex optimization. Lin et al. (2022) proposed a gradient-free method GFM and its stochastic counterpart SGFM, which achieve $O(d^{\frac{3}{2}}\delta^{-1}\epsilon^{-4})$ sample complexity. Chen et al. (2023) improved this complexity to $O(d^{\frac{3}{2}}\delta^{-1}\epsilon^{-3})$ by applying variance reduction. Furthermore, Kornowski & Shamir (2024) improved the dimension dependence to $O(d\delta^{-1}\epsilon^{-3})$ based on online-to-nonconvex conversion technique introduced by Cutkosky et al. (2023).

Deterministic Nonsmooth Nonconvex Setting. In the deterministic setting, even in the absence of noise, it is hard to deal with nonsmooth objectives, and randomization is necessary to obtain a dimension independent guarantee. Furthermore, deterministic algorithms require zero-order oracle for finite-time convergence guarantees (Jordan et al., 2022; 2023; Tian et al., 2022).

Distributed Smooth Setting. In the literature the term distributed may refer to different layers of optimization, i.e. application, protocol or network topology (Lu & De Sa, 2021). In federated learning, it refers to the application layer where each agent uses its local data with shared parameters (McMahan et al., 2016). In fully decentralized scenarios, each agent updates its local parameter using local data and communicates through a connected network. For nonconvex objectives, decentralized SGD has gained a lot of attention (Lian et al., 2017) due to the linear speed-up property. Many works analyzed decentralized algorithms under identically distributed data or bounded outer variance assumption for smooth problems (Tang et al., 2018; Koloskova et al., 2019; Li et al., 2020; Wang et al., 2020; Xu et al., 2023).

Decentralized Nonsmooth Setting. For decentralized nonconvex nonsmooth optimization, though the asymptotic analysis was previously explored in Swenson et al. (2022), there exists a scant literature on the *finite-time* analysis. For λ-weakly-convex nonsmooth objectives, Chen et al. (2021) provided finite-time guarantees on ME. More recently, Lin et al. (2024) proposed an algorithm (DGFM) that achieves $O(d^{3/2}\delta^{-1}\epsilon^{-4})$ complexity rate in the zero-order setting. In the same setup, they also proposed DGFM+ by incorporating variance reduction to obtain the $O(d^{3/2}\delta^{-1}\epsilon^{-3})$ complexity rate.

We also focus on decentralized nonsmooth nonconvex stochastic optimization in the present work. We develop a fully decentralized method that mimics a restarting decentralized online learning algorithm. Under mild technical assumptions (e.g., Lipschitz continuity of the objective function and unbiased, bounded variance gradients), we analyze the finite-time performance of the algorithm. We study three settings: (i) smooth first-order, (ii) nonsmooth first-order, and (iii) nonsmooth zero-order. For all of them, we establish the optimal sample complexity as previously derived for the centralized stochastic optimization (Tables 1-2).

2. Problem Setting

Notation: We denote by $\|x\|$ the Euclidean norm, by $[n]$ the set $\{1, 2, 3, ..., n\}$, by $B(x, \delta) := \{y \in \mathbb{R}^d : \|y - x\| \leq \delta\}$, by $conv(\cdot)$ the convex hull operator, and by $\texttt{unif}(A)$ the uniform measure over a set A. $\|\cdot\|_F$ denotes the Frobenius norm and $\|\cdot\|_2$ denotes the spectral norm. We use the standard notation $O(\cdot)$, $\Theta(\cdot)$, $\Omega(\cdot)$ to hide the absolute constants and $\tilde{O}(\cdot)$ to hide poly-logarithmic factors. 1_d and $\mathbb{S}^{d-1}$ denote the vector of all ones and the unit sphere in $\mathbb{R}^d$, respectively.

Network Setup: In decentralized learning, we have n agents that communicate through a network. We assume that the network is connected, i.e., there exists a (potentially multi-hop) path from any agent $i \in [n]$ to $j \neq i$. The net-

Table 1. The sample complexity of finding (δ, ϵ)-stationary points in centralized nonsmooth nonconvex stochastic optimization. d : ambient dimension, $\gamma = f(x_0) - \inf_x f(x)$ where x_0 is the initial point, G^2: bound on the second moment of stochastic gradient (for first-order methods) and bound on the second moment of Lipschitz constant (for zero-order methods). '*': Oracle requires an additional constraint on the directional derivative. The dependence to dimension d is only reported for zero-order methods.

ORACLE	METHOD	REFERENCE	COMPLEXITY
FIRST*	SINGD	(ZHANG ET AL., 2020B)	$\tilde{O}(\frac{\gamma G^3}{\delta\epsilon^4})$
FIRST	PSINGD	(TIAN ET AL., 2022)	$\tilde{O}(\frac{\gamma G^3}{\delta\epsilon^4})$
FIRST	O2NC	(CUTKOSKY ET AL., 2023)	$O(\frac{\gamma G}{\delta\epsilon^3})$
ZERO	SGFM	(LIN ET AL., 2022)	$O(d^{\frac{3}{2}}(\frac{G^4}{\epsilon^4} + \frac{\gamma G^3}{\delta\epsilon^4}))$
ZERO	GFM+	(CHEN ET AL., 2023)	$O(d^{\frac{3}{2}}(\frac{G^3}{\epsilon^3} + \frac{\gamma G^2}{\delta\epsilon^3}))$
ZERO	OSNNO	(KORNOWSKI & SHAMIR, 2024)	$O(\frac{d\gamma G^2}{\delta\epsilon^3})$

Table 2. The sample complexity of finding (δ, ϵ)-stationary points for decentralized nonsmooth nonconvex stochastic optimization. '*' : weakly convex setting, '**' : variance reduction. The dependence to dimension d is only reported for zero-order methods.

ORACLE	METHOD	REFERENCE	COMPLEXITY
FIRST*	DPSM	(CHEN ET AL., 2021)	$O(\epsilon^{-4})$
ZERO	DGFM	(LIN ET AL., 2024)	$O(d^{\frac{3}{2}}\delta^{-1}\epsilon^{-4})$
ZERO**	DGFM+	(LIN ET AL., 2024)	$O(d^{\frac{3}{2}}\delta^{-1}\epsilon^{-3})$
ZERO	ME-DOL	**Our Work**	$O(d\delta^{-1}\epsilon^{-3})$
FIRST	ME-DOL	**Our Work**	$O(\delta^{-1}\epsilon^{-3})$

work topology is governed by a symmetric doubly stochastic matrix $P = [P_{ij}]_{i,j=1}^n \in \mathbb{R}^{n\times n}$, where $1_n^\top P = 1_n^\top$ and $P1_n = 1_n$ (Assumption 1). Throughout the learning process, agent i receives only information about its local function f^i in the form of stochastic gradients or noisy function evaluations. Note that $P_{ij} \in [0, 1]$ and if $P_{ij} = 0$ agents i and j do not directly share information with each other. However, if $P_{ij} > 0$ agents share their decision variables as described in the ME-DOL (Algorithm 1). As such, the neighborhood of agent i is defined as $\mathcal{N}_i := \{j \in [n] : P_{ij} > 0\}$.

Information Oracles: We assume that each agent $i \in [n]$ has access to its information oracles. In the first-order setting, the oracle $\mathcal{O}_f$ returns stochastic gradient at query point x given by $\mathcal{O}_f^i(x) = \nabla F^i(x, \xi_i)$. For the first-order oracles we assume that the oracle returns unbiased estimates of the gradient with bounded variance (Assumption 4). In the zero-order setting, agents have access to the stochastic function value oracle $\mathcal{O}_z$ at query point x, that is $\mathcal{O}_z^i(x) = F^i(x, \xi_i)$ with Assumption 2 in place.

2.1. Stationarity Metric in Nonsmooth Analysis

In nonconvex smooth optimization problems, finding an ϵ-stationary point x, i.e. $\|\nabla f(x)\| \le \epsilon$ is a well-known tractable optimality condition. For nonsmooth objectives, a more relaxed criterion called near ϵ-stationarity can be considered for a point x with $\min\{\|g\| : g \in \cup_{y\in B(x,\delta)}\partial f(y)\} \le \epsilon$. However, both could be intractable criteria for nonsmooth objectives (Kornowski & Shamir, 2021). By Rademacher's Theorem, Lipschitz continuous functions are almost everywhere differentiable. For this class of functions, we can study (δ, ϵ)-stationarity. Let us first define Goldstein δ-subdifferential as follows.

Definition 1. Goldstein δ-subdifferential of f at x is the set

$$\partial_\delta f(x) := conv(\cup_{y\in B(x,\delta)}\partial f(y)),$$

where the Clarke subdifferential set $\partial f(x) := conv\{g : g = \lim_{x_s\to x} \nabla f(x_s)\}$.

Since Goldstein δ-subdifferential is the convex hull of a set of Clarke subdifferentials, it is possible that an element of Goldstein δ-subdifferential is not an element of Clarke subdifferential set of points $y \in B(x, \delta)$ for a nondifferentiable function f. An example of a function that has a (δ, ϵ)-stationary point that is not near ϵ-stationary is given in Kornowski & Shamir (2021) (Proposition 2).

Definition 2. Given a Lipschitz function $f : \mathbb{R}^d \to \mathbb{R}$, a point $x \in \mathbb{R}^d$ and $\delta > 0$, denote $\|\nabla f(x)\|_\delta := \min\{\|g\| : g \in \partial_\delta f(x)\}$. A point x is called a (δ, ϵ)-stationary point of $f(\cdot)$ if $\|\nabla f(x)\|_\delta \le \epsilon$.

This is a weaker notion than ϵ-stationarity or near ϵ-stationarity. In case of differentiable functions with L_1 Lipschitz gradients, an $(\frac{\epsilon}{3L_1}, \frac{\epsilon}{3})$-Goldstein stationary point is also ϵ-stationary (Zhang et al., 2020b).

2.2. Properties of Randomized Smoothing

In the nonsmooth analysis, finding (δ, ϵ)-stationary points of the global function $f(x) = \frac{1}{n}\sum_{i=1}^n f^i(x)$ is a reasonable and tractable optimality criterion using *randomized smoothing*.

Definition 3. Given an L-Lipschitz function f, we denote its smoothed surrogate as $f_\delta(x) := \mathbb{E}_{u\sim\mathcal{P}}[f(x + \delta u)]$, where $\mathcal{P}$ is the uniform distribution on the unit ball, i.e., $\texttt{unif}(B(0, 1))$.

Proposition 1. *(Lin et al., 2022) Suppose that the function $f : \mathbb{R}^d \to \mathbb{R}$ is L-Lipschitz. Then, it holds that:*

- $|f_\delta(\cdot) - f(\cdot)| \le \delta L$.

- $f_\delta(\cdot)$ *is L-Lipschitz.*
- $f_\delta(\cdot)$ *is differentiable with* $c\sqrt{d}L\delta^{-1}$*-Lipschitz gradients for a numeric constant* $c > 0$*.*
- $\nabla f_\delta(\cdot) \in \partial_\delta f(\cdot)$ *, where* $\partial_\delta f(\cdot)$ *is the Goldstein subdifferential.*

RS allows us to work with a "smoothed" objective f_δ with the cost of δL approximation error. Also, the last item in Proposition 1 links finding a (δ, ϵ)-stationary point of a nonsmooth objective f with finding an ϵ-stationary point of the smoothed objective f_δ. Furthermore, we will use the following lemma for nonsmooth analysis.

Lemma 1. *(Kornowski & Shamir, 2024) For any* $\delta, \mu \geq 0$: $\partial_\mu f_\delta(x) \subseteq \partial_{\mu+\delta} f(x)$.

Proposition 2. *By Lemma 1 and Definition 2 for* $\|\cdot\|_\delta$ *we have* $\|\nabla f(x)\|_\delta \leq \|\nabla f_{a\delta}(x)\|_{(1-a)\delta}$ *for any* $a \in (0, 1)$.

For the task of finding (δ, ϵ)-stationary points of a function, δ denotes the radius of the ball as in Definition 1. Lemma 1 allows us to distribute δ such that we can use a portion of it for smoothing and allocate the remaining part to the radius of the ball around the critical point. In Proposition 2 by choosing $a = \frac{1}{2}$ we have $\|\nabla f(x)\|_\delta \leq \|\nabla f_{\frac{\delta}{2}}(x)\|_{\frac{\delta}{2}}$.

Using these lemmas and randomized smoothing, we can facilitate our convergence analysis. In the context of decentralized optimization, we can use the surrogate function f_δ of global function f, and by the linearity of expectation we have $f_\delta = \frac{1}{n}\sum_{i=1}^n (f^i)_\delta$. Furthermore, using the following lemma, summation of gradients of smoothed functions $\frac{1}{n}\sum_{i=1}^n \nabla(f^i)_\delta(x_i)$ can be related to $\frac{1}{n}\sum_{i=1}^n \nabla f_\delta(x_i)$.

Lemma 2. *Suppose that* n *local functions* $\{f^i\}_{i=1}^n$ *have* L_1*-Lipschitz gradients and* $f(x) = \frac{1}{n}\sum_{i=1}^n f^i(x)$*. Consider the set of points* $\{w_{t,i}\}$ *for* $i \in [n], t \in [T]$*, and let* $\bar{w}_t = \frac{1}{n}\sum_{i=1}^n w_{t,i}$ *and* $\|w_{t,i} - \bar{w}_t\| \leq r, \forall i \in [n], \forall t \in [T]$*. Then, we have*

$$\left\|\frac{1}{nT}\sum_{t=1}^T\sum_{i=1}^n \nabla f(w_{t,i})\right\| \leq \left\|\frac{1}{nT}\sum_{t=1}^T\sum_{i=1}^n \nabla f^i(w_{t,i})\right\| + 2rL_1.$$

2.3. Assumptions

We assume that agents communicate synchronously through the network. For example, agent i takes a weighted average of the decision variables in its neighborhood as follows

$$y_{t,i} = \sum_{j\in\mathcal{N}_i} P_{ij}x_{t,j} = \sum_{j=1}^n P_{ij}x_{t,j}, \forall i \in [n],$$

as elaborated in Algorithm 1. The communication matrix P is fixed over time and satisfies the following assumption.

Assumption 1. The network is connected and the communication matrix $P \in \mathbb{R}^{n\times n}$ is symmetric and doubly stochastic. ρ denotes the second largest singular value of the matrix P. Given that the network is connected, we have that $\rho \in [0, 1)$.

Assumption 1 is widely used in the decentralized optimization literature (see e.g., (Shahrampour & Jadbabaie, 2018)). Regardless of whether the network structure is fixed or time-varying, some connectivity assumption is needed to solve the global problem. Here, the quantity ρ determines the connectivity of the network, and a smaller ρ indicates a more well-connected network topology.

Assumption 2. We assume that local objective functions have the form $f^i(x) = \mathbb{E}_\xi[F^i(x, \xi)]$, where ξ denotes the random index. The stochastic component of local functions $F^i(\cdot, \xi) : \mathbb{R}^d \to \mathbb{R}$ is $L(\xi)$-Lipschitz for any ξ, i.e., it holds that

$$\left|F^i(x, \xi) - F^i(y, \xi)\right| \leq L(\xi)\|x - y\|,$$

for any $x, y \in \mathbb{R}^d$ and $i \in [n]$. $L(\xi)$ has a bounded second moment such that $\mathbb{E}_\xi[L(\xi)^2] \leq L^2$.

We note that Assumption 2 is weaker than assuming that $F^i(\cdot, \xi)$ is L-Lipschitz (Chen et al., 2023; Kornowski & Shamir, 2024). Taking expectation from above, it can be shown that local functions f^i are Lipschitz continuous. However, we do *not* assume that gradients are Lipschitz. Furthermore, directional differentiability holds for commonly used nonsmooth functions (e.g., ReLU) and enables the use of Lebesgue path integrals (Zhang et al., 2020b).

Assumption 3. The local objectives $f^i : \mathbb{R}^d \to \mathbb{R}$ are lower bounded $(f^i)^* := \inf_x f^i(x) > -\infty$. Therefore, the global function f is also lower bounded, and we define γ such that $f(\bar{x}_0) - \inf_x f(x) \leq \gamma$, where $\bar{x}_0$ is the average of initial points (among agents) for the algorithm.

We also make the following standard assumptions on the stochastic gradients (Shahrampour & Jadbabaie, 2018; Zhang et al., 2020b) .

Assumption 4. We assume that the first-order oracle returns unbiased, bounded variance estimate of the gradient such that $\mathbb{E}[\nabla F^i(x, \xi)] = \nabla f^i(x)$ and $\mathbb{E}[\left\|\nabla F^i(x, \xi) - \nabla f^i(x)\right\|^2] \leq \sigma^2$. Furthermore, we assume that the second moment of the stochastic gradient is bounded such that $\mathbb{E}[\left\|\nabla F^i(x, \xi)\right\|^2] \leq G^2$.

3. Algorithm and Main Technical Results

In this section, we present our decentralized algorithm for finding a (δ, ϵ)-stationary point of the global objective f in (1). Our algorithm is termed Multi Epoch Decentralized Online Learning (ME-DOL), for which we establish the sample complexity in different settings. First, we present our result

Algorithm 1 Multi Epoch Decentralized Online Learning

Input: $\delta' \in \mathbb{R}_{\geq 0}$, $K \in \mathbb{N}$, $T \in \mathbb{N}$, decentralized online learning algorithm $\mathcal{A}$ with bounded domain $\mathcal{D}$, doubly stochastic communication matrix P.
Initialize: $y^0_{T,i} = 0$ for all $i \in [n]$.
for $k = 1$ **to** K **do**
  Restart $\mathcal{A}$
  Let $y^k_{0,i} = y^{k-1}_{T,i}$, $\forall i \in [n]$
  for $t = 1$ **to** T, $\forall i \in [n]$ **do**
    Get $\Delta^k_{t,i}$ for all agents from $\mathcal{A}$ (Algorithm 4)
    $x^k_{t,i} = y^k_{t-1,i} + \Delta^k_{t,i}$
    $s^k_{t,i} \sim \texttt{unif}[0,1]$
    $w^k_{t,i} = y^k_{t-1,i} + s^k_{t,i}\Delta^k_{t,i}$
    $y^k_{t,i} = \sum_{j=1}^n P_{ij} x^k_{t,j} = \sum_{j\in\mathcal{N}_i} P_{ij} x^k_{t,j}$
    if Information Oracle == Zero-Order **then**
      $g^k_{t,i}$ = Zero-Order Gradient$(F^i, w^k_{t,i}, \delta', \xi^k_{t,i})$
    else
      $g^k_{t,i}$ = First-Order Gradient$(F^i, w^k_{t,i}, \delta', \xi^k_{t,i})$
    end if
    Send $g^k_{t,i}$ to $\mathcal{A}$ as gradient
  end for
  Set $\bar{w}^k = \frac{1}{nT}\sum_{i=1}^n \sum_{t=1}^T w^k_{t,i}$ for $k \in [K]$
end for
Sample $w^{out} \sim \texttt{unif}\{\bar{w}^1, ..., \bar{w}^K\}$
Output : w^{out}

Algorithm 2 First-Order Gradient(F, x, δ', ξ)

Input: Function F, point x, smoothing parameter δ', random seed ξ.
Sample $z \sim \texttt{unif}(B(0,1))$
$g = \nabla F(x + \delta' z, \xi)$
Output: g

Algorithm 3 Zero-Order Gradient(F, x, δ', ξ)

Input: Function F, point x, smoothing parameter δ', random seed ξ, dimension d.
Sample $z \sim \texttt{unif}(\mathbb{S}^{d-1})$
Evaluate $F(x + \delta' z, \xi)$ and $F(x - \delta' z, \xi)$
$g = \frac{d}{2\delta'}\Big(F(x + \delta' z, \xi) - F(x - \delta' z, \xi)\Big) z$
Output: g

on smooth nonconvex objectives (Theorem 1), and we then extend our results to nonsmooth nonconvex objectives with randomized smoothing in first-order and zero-order settings (Theorems 2 and 3, respectively). Our technique is based on a reduction from nonsmooth nonconvex decentralized optimization to decentralized online learning.

In Algorithm 1, we have periods of length T, where in each epoch $k \in [K]$ a decentralized online algorithm $\mathcal{A}$ is used to generate action $\Delta^k_{t,i}$ for agent $i \in [n]$ at iteration $t \in [T]$. The action space $\mathcal{D}$ is bounded such that $\|\Delta\| \leq D$ for all $\Delta \in \mathcal{D}$. At each epoch k, $\mathcal{A}$ must run on a linear optimization problem with the objective of agent i as $\sum_{t=1}^T \langle \Delta, g^k_{t,i}\rangle$, where $g^k_{t,i}$ is the stochastic gradient (respectively, approximation of the stochastic gradient using noisy function evaluations) in the first-order (respectively, zero-order) setting. We use Algorithm 4 (Shahrampour & Jadbabaie, 2018) for $\mathcal{A}$. Based on action $\Delta^k_{t,i}$ the variable $x^k_{t,i}$ is generated and then averaged over neighborhood of i to get $y^k_{t,i}$. The set of nT points $w^k_{t,i}$ $\forall i \in [n], \forall t \in [T]$, at which the gradients are evaluated, are averaged to produce the final output of each epoch, denoted as $\bar{w}^k$. These points are proposed as candidates for identifying a (δ, ϵ)-stationary point of the global function f. Algorithm 1 outputs a randomly selected candidate point $\bar{w}^l$ where $l \sim \texttt{unif}[K]$.

Remark 2. Note that $P_{ij} = 0$ if agents i and j are not neighbors. Therefore, updates $y^k_{t,i} = \sum_{j=1}^n P_{ij} x^k_{t,j}$ (in Algorithm 1) and $\Delta^k_{t+\frac{1}{2},i} = \sum_{j=1}^n P_{ij}\Delta^k_{t,j}$ (in Algorithm 4) do not contradict the decentralized nature of the learning. Basically, for each agent $i \in [n]$, the sum always reduces to a weighted averaging over the neighborhood of agent i.

Remark 3. The original implementation of Algorithm 4 in Shahrampour & Jadbabaie (2018) is based on mirror descent, but here we use the Euclidean distance as the generator of Bregman divergence, reducing the algorithm to decentralized online gradient descent.

3.1. Challenges in the Analysis of Decentralized Algorithm

In centralized optimization, the difference of function values in consecutive iterations, $f(x_t) - f(x_{t-1})$, depends on the update rule. The update rule can be written as $x_t = x_{t-1} + \Delta_t$. For example, in SGD, $\Delta_t = -\eta \nabla F(x_{t-1}, \xi)$. Various algorithms use the past information to generate Δ_t or process the latest information as in the normalized gradient descent (Murray et al., 2019) or gradient clipping (Zhang et al., 2020a). For any algorithm with the update rule $x_t = x_{t-1} + \Delta_t$, one can write $f(x_t) = f(x_{t-1}) + \langle \Delta_t, \nabla_t \rangle$ where $\nabla_t = \int_0^1 \nabla f(x_{t-1} + s\Delta_t) ds$. Cutkosky et al. (2023)

Algorithm 4 Decentralized Online Optimization Algorithm $\mathcal{A}$ (Shahrampour & Jadbabaie, 2018)

Input: Domain $\mathcal{D}$, doubly stochastic matrix P, learning rate η, stochastic gradients $g^k_{t,i}$.
Initialize: $\Delta^k_{\frac{1}{2},i} = 0$ for all $i \in [n]$ and $k \in [K]$.
Iterations : Step $t \geq 1$, update for each $i \in [n]$:
$\Delta^k_{t,i} = \underset{\Delta\in\mathcal{D}}{\operatorname{argmin}} \Big\{ \eta \langle \Delta, g^k_{t-1,i}\rangle + \frac{1}{2}\big\|\Delta - \Delta^k_{t-\frac{1}{2},i}\big\|^2 \Big\}$
$\Delta^k_{t+\frac{1}{2},i} = \sum_{j=1}^n P_{ij}\Delta^k_{t,j}$

observed that Δ_t can be generated via an online learning algorithm (e.g., online gradient descent (OGD)), in which case the summation of the differences for T rounds can be written as follows

$$f(x_T) - f(x_0) = \sum_{t=1}^{T} \langle \Delta_t, \nabla_t \rangle$$
$$= \sum_{t=1}^{T} \langle g_t, \Delta_t - u \rangle + \sum_{t=1}^{T} \langle \nabla_t - g_t, \Delta_t \rangle + \sum_{t=1}^{T} \langle g_t, u \rangle,$$

where $\{g_t\}$ are the stochastic gradients provided to the online learning algorithm. This equation holds for any u in hindsight, and the first summation term corresponds to the regret of the online algorithm. The second term has expectation equal to 0 (if gradients are unbiased), and for the last term we have freedom to select an optimal u.

In the decentralized counterpart of this conversion, which is the focal point of our analysis, the main difference is that the change in the global function depends on the average action, i.e., $\bar{\Delta}_t = \bar{x}_t - \bar{x}_{t-1}$, but $f(\bar{x}_t) = f(\bar{x}_{t-1}) + \langle \bar{\Delta}_t, \tilde{\nabla}_t \rangle$ where $\tilde{\nabla}_t = \int_0^1 \nabla f(\bar{x}_{t-1} + s\bar{\Delta}_t) ds$. The key technical challenge is the discrepancy between $\tilde{\nabla}_t$ and $\bar{\nabla}_t = \sum_{i=1}^n \nabla_{t,i}/n$, where $\nabla_{t,i} := \int_0^1 \nabla f^i(y_{t-1,i} + s\Delta_{t,i}) ds$. In the decentralized analysis, we have the decentralized regret term and an additional discrepancy term that arises due to the difference between $\tilde{\nabla}_t - \bar{\nabla}_t$, which requires careful analysis.

3.2. Smooth Analysis

Let us now present our first result on smooth objectives in the following theorem.

Theorem 1. *Let Assumptions 1, 3, 4 hold and further assume that f^i is L-Lipschitz and has L_1-Lipschitz gradient for all $i \in [n]$. Let $\delta, \epsilon \in (0,1)$ and choose*

- $N := KT = \Theta(\delta^{-1}\epsilon^{-3}(1-\rho)^{-2})$,
- $T = \Theta((1-\rho)^{\frac{1}{3}} (\delta N)^{\frac{2}{3}})$,
- $D = \frac{\delta(1-\rho)}{2T\sqrt{n}}$,
- $\eta = \Theta(\sqrt{(1-\rho)} \frac{D}{\sqrt{T}})$.

Then, running Algorithm 1 with $\delta' = 0$ for N rounds gives an output that satisfies the following inequality for the global function $f(x) = \frac{1}{n}\sum f^i(x)$,

$$\mathbb{E}_{k \sim unif[K]} \left[\left\| \nabla f(\bar{w}^k) \right\|_\delta \right] \leq \epsilon.$$

Remark 4. For smooth objectives this result implies that a (δ, ϵ)-stationary point can be found in $N = O(\delta^{-1}\epsilon^{-3})$ iterations. This rate results in the optimal complexity of $O(\epsilon^{-4})$ for finding an ϵ-stationary point of nonconvex smooth objectives (Arjevani et al., 2023; Lu & De Sa, 2021) as $\delta = O(\epsilon)$. Furthermore, the dependence of N to $(1-\rho)^{-2}$ indicates that in a well-connected network (smaller ρ), we need less iterations to find a (δ, ϵ)-stationary point.

For smooth objectives we do not need randomized smoothing, so we choose $\delta' = 0$. We can use the full budget δ for the search radius, so we set $D = \frac{\delta(1-\rho)}{2T\sqrt{n}}$ in order to satisfy $\|\bar{w}^k - w_{t,i}^k\| \leq \delta$. The complete proof of Theorem 1 can be found in the Appendix (Section A.3).

3.3. Challenges in Nonsmooth Analysis

For nonsmooth objectives, we utilize randomized smoothing. The straightforward application of randomized smoothing, such as merely replacing the task of finding a (δ, ϵ)-stationary point of a nonsmooth function f with the task of finding an ϵ-stationary point of the smoothed function would result in the sub-optimal complexity of $O(\delta^{-1}\epsilon^{-4})$ since the optimal rate for decentralized smooth nonconvex objectives is $O(L_1\epsilon^{-4})$ (Lu & De Sa, 2021), and $L_1 = O(\delta^{-1})$ for the smoothed function according to Proposition 1.

To address this, we must control δ that affects the smoothness parameter L_1. In the proof of Theorem 1, we have the following inequality (see Equation 9), which also plays an important role in the nonsmooth analysis.

$$\mathbb{E}\left[\frac{1}{K} \sum_{k=1}^{K} \left\| \frac{1}{nT} \sum_{t=1}^{T} \sum_{i=1}^{n} \nabla f(w_{t,i}^k) \right\| \right]$$
$$\leq \frac{2\gamma T\sqrt{n}}{\delta N(1-\rho)} + \frac{\sigma}{\sqrt{nT}} + \frac{c_1}{\sqrt{T}} + \frac{\delta L_1 (1-\rho) c_3}{2T\sqrt{n}}.$$

However, for nonsmooth objectives, f must be replaced by a smoothed function $f_{(1-a)\delta}$ in the left-hand side, and using Proposition 2, we can consider finding an $(a\delta, \epsilon)$-stationary point of $f_{(1-a)\delta}$ for $0 < a < 1$. A larger a increases the "radius of possible stationarity" but decreases the "smoothness", and we need to balance this trade-off.

3.4. Nonsmooth Analysis with First-Order Oracle

For nonsmooth objectives, we can choose $a = \frac{1}{2}$, and the goal is to find a $(\frac{\delta}{2}, \epsilon)$-stationary point of $f_{\frac{\delta}{2}}$. Then, we can extend the result of Theorem 1 with the smoothness parameter $L_1 = 2c\sqrt{d}L\delta^{-1}$, where c is a constant that depends on the geometry of the problem (see Appendix B).

Theorem 2. *Let $\delta, \epsilon \in (0,1)$. Suppose that Assumptions 1, 3, 4 hold and that f^i is L-Lipschitz for all $i \in [n]$. Choose N, T, η as in Theorem 1, and set $D = \frac{\delta(1-\rho)}{4T\sqrt{n}}$. Then, running Algorithm 1 with $\delta' = \frac{\delta}{2}$ for N rounds gives an output that satisfies the following inequality*

$$\mathbb{E}_{k \sim unif[K]} \left[\left\| \nabla f(\bar{w}^k) \right\|_\delta \right] \leq c_8 (\delta N)^{-\frac{1}{3}} \leq \epsilon,$$

where $c_8 = O((1-\rho)^{-\frac{2}{3}})$ and it does not depend on δ and N (see Appendix A.4 for the exact quantification of c_8).

To the best of our knowledge, the above theorem establishes the first complexity rate for nonsmooth nonconvex functions in decentralized stochastic optimization (without weak-convexity assumption). In terms of δ and ϵ, the rate $N = O(\delta^{-1}\epsilon^{-3})$ matches the best known rates in the centralized setting (Cutkosky et al., 2023). This rate also recovers the optimal results in the smooth nonconvex setting when $\delta = O(\epsilon)$.

3.5. Nonsmooth Analysis with Zero-Order Oracle

In the first-order setting, we used Assumption 4 that implies access to unbiased, bounded variance gradient estimates. In the zero-order setting, agents do not have access to the gradient information, but they can estimate the gradient using noisy function values. The output of Algorithm 3 provides an unbiased gradient estimator with bounded variance (Shamir, 2017), and it can be used in lieu of stochastic gradients returned by the first-order oracle.

Lemma 3. *(Kornowski & Shamir, 2024) Let $w = x + s\Delta$ be a point with $s \sim \texttt{unif}[0,1]$. The gradient estimator*

$$g = \frac{d}{2\delta'}\Big(F(x + s\Delta + \delta' z, \xi) - F(x + s\Delta - \delta' z, \xi)\Big)z,$$

as generated by Algorithm 3 satisfies the following conditions

$$\mathbb{E}_{\xi,z}[g|x,s,\Delta] = \nabla f_{\delta'}(x + s\Delta) = \nabla f_{\delta'}(w),$$

and

$$\mathbb{E}_{\xi,z}[\|g\|^2 |x,s,\Delta] \leq 16\sqrt{2\pi} d L^2.$$

The bound on the second moment helps us replace G and σ in the first-order setting (Assumption 4) by a quantifiable constant. Running ME-DOL using (the noisy version of) smoothed functions f_δ^i and the gradient estimator in Algorithm 3, we have the following convergence guarantee for the zero-order setting.

Theorem 3. *Let $\delta, \epsilon \in (0,1)$. Suppose that Assumptions 1, 2, 3 hold and that the zero-order oracle returns unbiased estimates of the function values. Choose N, T, η as in Theorem 1, and set $D = \frac{\delta(1-\rho)}{4T\sqrt{n}}$. Then, running Algorithm 1 with $\delta' = \frac{\delta}{2}$ for N rounds gives an output that satisfies the following inequality*

$$\mathbb{E}_{k\sim \texttt{unif}[K]}\left[\left\|\nabla f(\bar{w}^k)\right\|_\delta\right] \leq c_{11}(\delta N)^{-\frac{1}{3}} \leq \epsilon,$$

where $c_{11} = O(d^{\frac{1}{3}}(1-\rho)^{-\frac{2}{3}})$ and it does not depend on δ and N (see Appendix A.5 for the exact quantification of c_{11}).

Similar to previous results, the complexity of finding a (δ,ϵ)-stationary point is $N = O(\delta^{-1}\epsilon^{-3})$ in terms of δ and ϵ. In decentralized zero-order nonsmooth nonconvex stochastic optimization, our result matches the best known rate $O(\delta^{-1}\epsilon^{-3})$ as shown in Table 2 *without* recourse to variance reduction.

Remark 5. In the zero-order setting (Table 2), the dimension dependence of our algorithm is $O(d)$, which improves upon DGFM and DGFM+, where the dimension dependence is $O(d^{\frac{3}{2}})$. This result also matches with the optimal dimension dependence in the centralized setting following the analysis of Kornowski & Shamir (2024).

4. Numerical Experiments

To validate the performance of our algorithm, we conduct experiments on several datasets[1].

Model. We consider the nonconvex penalized SVM with capped-ℓ_1 regularizer. The model trains a binary classifier $x \in \mathbb{R}^d$ on the training data $\{a_i, b_i\}_{i=1}^m$, where $a_i \in \mathbb{R}^d$ and $b_i \in \{-1,1\}$ are the (normalized) feature vector and label for the i-th sample, respectively. Local objective functions can be written as

$$f^i(x) = \frac{1}{m_i}\sum_{j=1}^{m_i} l(b_i^j (a_i^j)^\top x) + \nu(x),$$

where $l(y) = \max\{1-y, 0\}$, $m = \sum_{i=1}^n m_i$, $\nu(x) = \lambda \sum_{j=1}^d \min\{|x(j)|, \alpha\}$, and $\lambda, \alpha > 0$. Similar to experiments of Lin et al. (2024), we set $\lambda = 10^{-5}/n$ and $\alpha = 2$. For each dataset, we divide the training samples equally among agents, i.e., $m_i = m/n$.

Setup. We consider a network of $n = 20$ agents with a ring topology. Hyper-parameters of our algorithm, η and D, are selected based on the theorems, where $\eta = \Theta(D/\sqrt{T})$. We set $\eta = 0.01 \times D$ and vary D in the range of 10^{-4} to 10^{-2} in different experiments.

Results. To empirically analyze the performance of ME-DOL, global gradient norms $\left\|\nabla f(\bar{w}^k)\right\|$ for $k \geq 1$ are calculated for both first-order and zero-order settings. We use three datasets (ijcnn, rcv, SUSY) to illustrate the decay of gradient norms with respect to iterations. The plots are reported in Figs. 1 and 2. This observation validates Theorems 2 and 3 in our paper, respectively.

We further evaluate the classification accuracy over the test data. We compare our algorithm in the zero-order setting with DGFM in Lin et al. (2024) on three datasets (a9a, HIGGS, covtype), and accuracy plots are reported in Fig. 5 (see Appendix C). We can see that our algorithm dominates DGFM in terms of the test classification accuracy.

[1]Codes for numerical experiments are available at https://github.com/emreesahinoglu/Decentralized-Nonsmooth.git

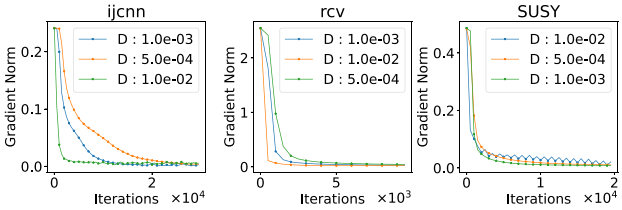

Figure 1. Evaluation of the gradient norm in the first-order setting.

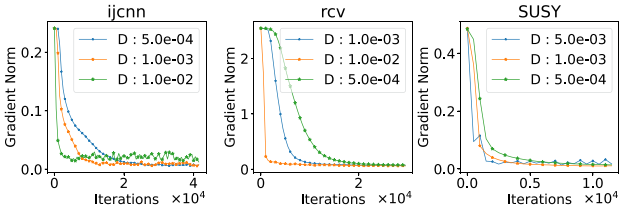

Figure 2. Evaluation of the gradient norm in the zero-order setting.

In the first-order setting, we compare our algorithm with DPSGD in Lian et al. (2017) on three datasets (a9a, HIGGS, covtype), and accuracy plots are depicted in Fig. 6 (see Appendix C), where again our algorithm achieves a better performance in terms of the classification accuracy. Note that DPSGD was originally proposed for smooth problems, but the same algorithm with projection (DPSM) was analyzed in the nonsmooth setting as well (Chen et al., 2021).

Impact of Network: We also evaluate the effect of network connectivity on the ring-based graphs with $n = 20$ agents, using the number of neighbors from $\{7, 9, 11, 13\}$. The corresponding ρ values are $\{0.81, 0.70, 0.57, 0.44\}$, respectively. As the number of neighbors increases, the graph becomes more connected, and the value of ρ decreases. In Fig. 3 we observe that better connectivity (smaller ρ) results in a faster convergence in the first-order setting.

To further evaluate the effect of network topology, we design the communication matrices based on Erdos-Renyi random graph $G(20, p)$, where p represents the probability of existence of an edge. In this experiment, p is selected from $\{0.5, 0.6, 0.7, 0.8\}$, for which the corresponding ρ values

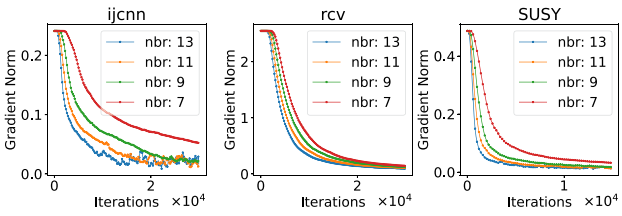

Figure 3. Ring graphs in the first-order setting.

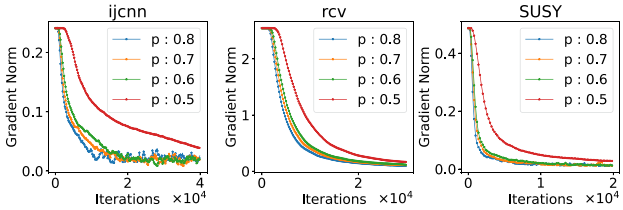

Figure 4. Random graphs in the first-order setting.

are $\{0.83, 0.63, 0.56, 0.47\}$. The plots are presented in Fig. 4, where again we observe that larger probability, which implies (possibly) better connectivity, results in a faster convergence. Note that since in this experiment the graphs are generated randomly, one might get different values for ρ even by trying the same edge probabilities.

5. Conclusion

We presented a novel algorithm for decentralized nonsmooth nonconvex stochastic optimization in first-order and zero-order oracle settings. We adopted recent techniques on online-to-nonconvex conversion (Cutkosky et al., 2023) and the geometric lemma on Goldstein subdifferential sets (Kornowski & Shamir, 2024) to streamline the finite-time analysis of the algorithm. Our algorithm achieved the optimal sample complexity of $O(\delta^{-1}\epsilon^{-3})$ for finding (δ, ϵ)-stationary points of the global objective in three settings, namely (i) smooth first-order, (ii) nonsmooth first-order, and (iii) nonsmooth zero-order. Notably, to the best of our knowledge, we provided the first finite-time convergence characterization in the nonsmooth first-order setting (without weak-convexity assumption (Chen et al., 2021)), and our result on the nonsmooth zero-order setting does not use variance reduction. Future directions include the investigation of high probability bounds (as opposed to expectation), the optimal dependence to network parameters, as well as convergence in the deterministic regime.

In our theorems, we found that $N = O((1-\rho)^{-2})$, but it is challenging to evaluate the optimality with respect to ρ in the nonsmooth setting using the (δ, ϵ)-stationarity concept. There is currently no lower bound on the communication complexity for the decentralized nonsmooth nonconvex stochastic optimization, i.e., in the nonsmooth setting the optimal dependence on ρ in finding (δ, ϵ)-stationary points has not been explored yet. For the smooth decentralized setting, the lower bound on ρ-dependency is given as $O((1-\rho)^{-\frac{1}{2}})$ (Lu & De Sa, 2021), using a carefully designed communication protocol that allows for network structure change. Whether the optimal dependence to ρ in the nonsmooth setting is the same and whether that potential gap can be closed are interesting research questions.

Impact Statement

We do not anticipate any future societal consequences as this work contributes to the theory of decentralized optimization.

Acknowledgements

The authors gratefully acknowledge the support of Mechanical and Industrial Engineering (MIE) Chair Fellowship at Northeastern University as well as NSF ECCS-2240788 Award for this research.

References

Arjevani, Y., Carmon, Y., Duchi, J. C., Foster, D. J., Srebro, N., and Woodworth, B. Lower bounds for non-convex stochastic optimization. *Mathematical Programming*, 199 (1-2):165–214, 2023.

Carmon, Y., Duchi, J. C., Hinder, O., and Sidford, A. Lower bounds for finding stationary points I. *Mathematical Programming*, 184(1-2):71–120, 2020.

Chen, L., Xu, J., and Luo, L. Faster gradient-free algorithms for nonsmooth nonconvex stochastic optimization. In *International Conference on Machine Learning (ICML)*, pp. 5219–5233. PMLR, 2023.

Chen, S., Garcia, A., and Shahrampour, S. On distributed nonconvex optimization: Projected subgradient method for weakly convex problems in networks. *IEEE Transactions on Automatic Control*, 67(2):662–675, 2021.

Clarke, F. H., Ledyaev, Y. S., Stern, R. J., and Wolenski, P. R. *Nonsmooth analysis and control theory*, volume 178. Springer Science & Business Media, 2008.

Cutkosky, A., Mehta, H., and Orabona, F. Optimal stochastic non-smooth non-convex optimization through online-to-non-convex conversion. In *International Conference on Machine Learning (ICML)*, pp. 6643–6670. PMLR, 2023.

Daniilidis, A. and Drusvyatskiy, D. Pathological subgradient dynamics. *SIAM Journal on Optimization*, 30(2): 1327–1338, 2020.

Davis, D. and Drusvyatskiy, D. Stochastic model-based minimization of weakly convex functions. *SIAM Journal on Optimization*, 29(1):207–239, 2019.

Davis, D., Drusvyatskiy, D., Lee, Y. T., Padmanabhan, S., and Ye, G. A gradient sampling method with complexity guarantees for lipschitz functions in high and low dimensions. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:6692–6703, 2022.

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

Jordan, M., Kornowski, G., Lin, T., Shamir, O., and Zampetakis, M. Deterministic nonsmooth nonconvex optimization. In *The Thirty Sixth Annual Conference on Learning Theory*, pp. 4570–4597. PMLR, 2023.

Jordan, M. I., Lin, T., and Zampetakis, M. On the complexity of deterministic nonsmooth and nonconvex optimization. *arXiv preprint arXiv:2209.12463*, 2022.

Koloskova, A., Stich, S., and Jaggi, M. Decentralized stochastic optimization and gossip algorithms with compressed communication. In *International Conference on Machine Learning (ICML)*, pp. 3478–3487, 2019.

Kornowski, G. and Shamir, O. Oracle complexity in nonsmooth nonconvex optimization. *Advances in Neural Information Processing Systems (NeurIPS)*, 34:324–334, 2021.

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

Li, T., Sahu, A. K., Zaheer, M., Sanjabi, M., Talwalkar, A., and Smith, V. Federated optimization in heterogeneous networks. *Proceedings of Machine learning and systems*, 2:429–450, 2020.

Lian, X., Zhang, C., Zhang, H., Hsieh, C.-J., Zhang, W., and Liu, J. Can decentralized algorithms outperform centralized algorithms? a case study for decentralized parallel stochastic gradient descent. *Advances in neural information processing systems (NeurIPS)*, 30, 2017.

Lin, T., Zheng, Z., and Jordan, M. Gradient-free methods for deterministic and stochastic nonsmooth nonconvex optimization. *Advances in Neural Information Processing Systems (NeurIPS)*, 35:26160–26175, 2022.

Lin, Z., Xia, J., Deng, Q., and Luo, L. Decentralized gradient-free methods for stochastic non-smooth nonconvex optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 17477–17486, 2024.

Liu, J. S. and Liu, J. S. *Monte Carlo strategies in scientific computing*, volume 75. Springer, 2001.

Lu, Y. and De Sa, C. Optimal complexity in decentralized training. In *International Conference on Machine Learning (ICML)*, pp. 7111–7123. PMLR, 2021.

Majewski, S., Miasojedow, B., and Moulines, E. Analysis of nonsmooth stochastic approximation: the differential inclusion approach. *arXiv preprint arXiv:1805.01916*, 2018.

McMahan, H. B., Yu, F., Richtarik, P., Suresh, A., and Bacon, D. Federated learning: Strategies for improving communication efficiency. In *Proceedings of the 29th Conference on Neural Information Processing Systems (NeurIPS), Barcelona, Spain*, pp. 5–10, 2016.

Murray, R., Swenson, B., and Kar, S. Revisiting normalized gradient descent: Fast evasion of saddle points. *IEEE Transactions on Automatic Control*, 64(11):4818–4824, 2019.

Nedic, A. and Ozdaglar, A. Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control*, 54(1):48–61, 2009. doi: 10.1109/TAC.2008.2009515.

Rockafellar, R. T. and Wets, R. J.-B. *Variational analysis*, volume 317. Springer Science & Business Media, 2009.

Scaman, K., Bach, F., Bubeck, S., Massoulié, L., and Lee, Y. T. Optimal algorithms for non-smooth distributed optimization in networks. *Advances in Neural Information Processing Systems (NeurIPS)*, 31, 2018.

Scaman, K., Dos Santos, L., Barlier, M., and Colin, I. A simple and efficient smoothing method for faster optimization and local exploration. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:6503–6513, 2020.

Shahrampour, S. and Jadbabaie, A. Distributed online optimization in dynamic environments using mirror descent. *IEEE Transactions on Automatic Control*, 63(3):714–725, 2018.

Shamir, O. An optimal algorithm for bandit and zero-order convex optimization with two-point feedback. *The Journal of Machine Learning Research*, 18(1):1703–1713, 2017.

Swenson, B., Murray, R., Poor, H. V., and Kar, S. Distributed stochastic gradient descent: Nonconvexity, nonsmoothness, and convergence to local minima. *Journal of Machine Learning Research*, 23(328):1–62, 2022.

Tang, H., Lian, X., Yan, M., Zhang, C., and Liu, J. d^2: Decentralized training over decentralized data. In *International Conference on Machine Learning (ICML)*, pp. 4848–4856. PMLR, 2018.

Tian, L., Zhou, K., and So, A. M.-C. On the finite-time complexity and practical computation of approximate stationarity concepts of lipschitz functions. In *International Conference on Machine Learning (ICML)*, pp. 21360–21379. PMLR, 2022.

Wang, J., Liu, Q., Liang, H., Joshi, G., and Poor, H. V. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems (NeurIPS)*, 33:7611–7623, 2020.

Xu, C., Qu, Y., Xiang, Y., and Gao, L. Asynchronous federated learning on heterogeneous devices: A survey. *Computer Science Review*, 50:100595, 2023.

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

Zhang, J., He, T., Sra, S., and Jadbabaie, A. Why gradient clipping accelerates training: A theoretical justification for adaptivity. In *International Conference on Learning Representations (ICLR)*, 2020a.

Zhang, J., Lin, H., Jegelka, S., Sra, S., and Jadbabaie, A. Complexity of finding stationary points of nonconvex nonsmooth functions. In *International Conference on Machine Learning (ICML)*, pp. 11173–11182, 2020b.

A. Proof of Theorems

First, we state the following lemma to use in the proof of Theorem 1.

Lemma 4. *Given Assumption 1, for the update in Algorithm 1, we have for any $i \in [n]$, $t \in [T]$, $k \in [K]$ that*

$$\left\|\bar{y}_t^k - y_{t,i}^k\right\| \leq \frac{D\sqrt{n}}{1-\rho}.$$

Proof. We have based on the update rule that

$$y_{t,i}^k = \sum_{j=1}^n P_{ij} x_{t,j}^k = \sum_{j=1}^n P_{ij} y_{t-1,j}^k + \sum_{j=1}^n P_{ij} \Delta_{t,j}^k.$$

Let $y_t^k \in \mathbb{R}^{nd}$ be the concatenation of vectors $y_{t,1}^k, y_{t,2}^k, \ldots, y_{t,n}^k$, and $\zeta_t^k \in \mathbb{R}^{nd}$ be the concatenation of vectors $\sum_{j=1}^n P_{1j}\Delta_{t,j}^k, \sum_{j=1}^n P_{2j}\Delta_{t,j}^k, \ldots, \sum_{j=1}^n P_{nj}\Delta_{t,j}^k$. We then have

$$y_t^k = (P \otimes I) y_{t-1}^k + \zeta_t^k.$$

Without loss of generality let $y_0^k = 0$. Then,

$$y_{t,i}^k = \sum_{j=1}^n \sum_{\tau=0}^{t-1} [P^{t-1-\tau}]_{ij} \zeta_{\tau+1,j}^k \Rightarrow y_{t,i}^k - \bar{y}_t^k = \sum_{j=1}^n \sum_{\tau=0}^{t-1} \Big([P^{t-1-\tau}]_{ij} - \frac{1}{n}\Big) \zeta_{\tau+1,j}^k.$$

Combining the geometric mixing bound of $\sum_{j=1}^n |P_{ij}^t - \frac{1}{n}| \leq \sqrt{n}\rho^t$ (Liu & Liu, 2001) and the fact that $\|\zeta_{\tau+1,j}^k\| \leq D$, the proof is complete. □

A.1. Proof of Proposition 2

Proof. By Lemma 1 we have $\partial_\mu f_\delta(x) \subseteq \partial_{\mu+\delta} f(x)$. Using Definition 2, we have $\|\nabla f(x)\|_{\mu+\delta} \leq \|\nabla f_\delta(x)\|_\mu$. Replacing δ with $a\delta$ and μ with $(1-a)\delta$ for $a \in (0,1)$ gives $\|\nabla f(x)\|_\delta \leq \|\nabla f_{a\delta}(x)\|_{(1-a)\delta}$. □

A.2. Proof of Lemma 2

Proof. We know that

$$\frac{1}{n}\sum_{i=1}^n \nabla f^i(\bar{w}_t) = \nabla f(\bar{w}_t) = \frac{1}{n}\sum_{i=1}^n \nabla f(\bar{w}_t).$$

Using L_1 smoothness of f^i and f and $\|w_{t,i} - \bar{w}_t\| \leq r$, we have that $\left\|\nabla f^i(w_{t,i}) - \nabla f^i(\bar{w}_t)\right\| \leq rL_1$ and $\|\nabla f(w_{t,i}) - \nabla f(\bar{w}_t)\| \leq rL_1$. Therefore,

$$\begin{aligned}
\left\|\frac{1}{nT}\sum_{t=1}^T\sum_{i=1}^n \nabla f(w_{t,i})\right\| &\leq \frac{1}{nT}\sum_{t=1}^T \left\|\sum_{i=1}^n \nabla f(w_{t,i}) - \sum_{i=1}^n \nabla f(\bar{w}_t)\right\| + \frac{1}{nT}\sum_{t=1}^T \left\|\sum_{i=1}^n \nabla f^i(\bar{w}_t) - \sum_{i=1}^n \nabla f^i(w_{t,i})\right\| \\
&\quad + \left\|\frac{1}{nT}\sum_{t=1}^T\sum_{i=1}^n \nabla f^i(w_{t,i})\right\| \\
&\leq \left\|\frac{1}{nT}\sum_{t=1}^T\sum_{i=1}^n \nabla f^i(w_{t,i})\right\| + 2rL_1,
\end{aligned}$$

which completes the proof. □

A.3. Proof of Theorem 1

Proof. Throughout the proof, superscript k denotes the k-th epoch, subscript i denotes the agent index, and subscript t represents the iteration.

Let us start with the following definitions:

$g_{t,i}^k := \mathcal{O}_f^i(w_{t,i}^k)$ (stochastic gradient returned by the oracle)

$\bar{g}_t^k := \frac{1}{n}\sum_{i=1}^n g_{t,i}^k$ (average of local stochastic gradients)

$\Delta_{t,i}^k$ is generated based on the decentralized online learning algorithm $\mathcal{A}$ (Algorithm 4)

$\bar{\Delta}_t^k := \frac{1}{n}\sum_{i=1}^n \Delta_{t,i}^k$ (average of local actions)

$\nabla_{t,i}^k := \int_0^1 \nabla f^i(y_{t-1,i}^k + s\Delta_{t,i}^k)ds$ (expected local gradient)

$\bar{\nabla}_t^k := \frac{1}{n}\sum_{i=1}^n \nabla_{t,i}^k$ (average of expected local gradients)

$\tilde{\nabla}_t^k := \int_0^1 \nabla f(\bar{x}_{t-1}^k + s\bar{\Delta}_t^k)ds$ (expected global gradient)

For any Lipschitz continuous function and an update rule $x_t = x_{t-1} + \Delta_t$ we have

$$f(x_t) - f(x_{t-1}) = \int_0^1 \langle \nabla f(x_{t-1} + s\Delta_t), \Delta_t\rangle ds = \langle \nabla_t, \Delta_t\rangle.$$

In our decentralized update rule it follows by doubly stochasticity of P that

$$\bar{y}_t^k := \frac{1}{n}\sum_{i=1}^n y_{t,i}^k = \frac{1}{n}\sum_{i=1}^n\sum_{j=1}^n P_{ij}x_{t,j}^k = \frac{1}{n}\sum_{j=1}^n\sum_{i=1}^n P_{ij}x_{t,j}^k = \frac{1}{n}\sum_{j=1}^n x_{t,j}^k =: \bar{x}_t^k,$$

and since $x_{t,i}^k = y_{t-1,i}^k + \Delta_{t,i}^k$, we get that $\bar{x}_t^k = \bar{x}_{t-1}^k + \bar{\Delta}_t^k$.

Therefore, we can write the following for the global function $f(x) = \frac{1}{n}\sum_{i=1}^n f^i(x)$,

$$f(\bar{x}_t^k) - f(\bar{x}_{t-1}^k) = \langle \tilde{\nabla}_t^k, \bar{\Delta}_t^k\rangle,$$

and summing both sides over $t \in [T]$ gives

$$f(\bar{x}_T^k) - f(\bar{x}_0^k) = \sum_{i=1}^T \langle \tilde{\nabla}_t^k, \bar{\Delta}_t^k\rangle.$$

There are two sources of randomness in the algorithm, namely $\xi_{t,i}^k$ and $s_{t,i}^k$. Taking expectation over those, we can decompose above into four terms:

$$\begin{aligned}\mathbb{E}[f(\bar{x}_T^k) - f(\bar{x}_0^k)] &= \sum_{t=1}^T \mathbb{E}[\langle \bar{\Delta}_t^k, \tilde{\nabla}_t^k\rangle] \\ &= \underbrace{\sum_{t=1}^T \mathbb{E}[\langle \bar{g}_t^k, \bar{\Delta}_t^k - u^k\rangle]}_{R_T^k(u^k)} + \underbrace{\sum_{t=1}^T \mathbb{E}[\langle \bar{g}_t^k, u^k\rangle]}_{T_2} + \underbrace{\sum_{t=1}^T \mathbb{E}[\langle \bar{\Delta}_t^k, \tilde{\nabla}_t^k - \bar{\nabla}_t^k\rangle]}_{T_3} + \sum_{t=1}^T \mathbb{E}[\langle \bar{\Delta}_t^k, \bar{\nabla}_t^k - \bar{g}_t^k\rangle], \quad (2)\end{aligned}$$

where the last term equals zero due to the unbiased gradient assumption that $\mathbb{E}[\bar{g}_t^k] = \bar{\nabla}_t^k$. The above holds for any u^k, and choosing $u^k = -D\frac{\sum_{t=1}^T\sum_{i=1}^n \nabla f^i(w_{t,i}^k)}{\left\|\sum_{t=1}^T\sum_{i=1}^n \nabla f^i(w_{t,i}^k)\right\|}$, we have that

$$\begin{aligned}T_2 &= \mathbb{E}[\langle \sum_{t=1}^T \bar{g}_t^k, u^k\rangle] = \mathbb{E}[\langle u^k, \frac{1}{n}\sum_{t=1}^T\sum_{i=1}^n \nabla f^i(w_{t,i}^k)\rangle] + \mathbb{E}[\langle u^k, \sum_{t=1}^T \bar{g}_t^k - \frac{1}{n}\sum_{t=1}^T\sum_{i=1}^n \nabla f^i(w_{t,i}^k)\rangle] \\ &\le \mathbb{E}\left[-DT\left\|\frac{1}{nT}\sum_{t=1}^T\sum_{i=1}^n \nabla f^i(w_{t,i}^k)\right\|\right] + \mathbb{E}\left[\frac{D}{n}\left\|\sum_{t=1}^T\sum_{i=1}^n (\nabla f^i(w_{t,i}^k) - g_{t,i}^k)\right\|\right] \\ &\le \mathbb{E}\left[-DT\left\|\frac{1}{nT}\sum_{t=1}^T\sum_{i=1}^n \nabla f^i(w_{t,i}^k)\right\|\right] + D\sigma\sqrt{\frac{T}{n}},\end{aligned}$$

where we used Assumption 4 and Jensen's inequality in the last line.

Rearranging Equation (2) using above and dividing by DT yields

$$\underbrace{\mathbb{E}\left[\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f^i(w_{t,i}^k)\right\|\right]}_{\epsilon-term} \leq \underbrace{\frac{\mathbb{E}[f(\bar{x}_0^k)-f(\bar{x}_T^k)]}{DT}}_{sub-optimality} + \underbrace{\frac{\sigma}{\sqrt{nT}}}_{noise} + \underbrace{\frac{\mathbb{E}[\sum_{t=1}^{T}\langle \bar{g}_t^k, \bar{\Delta}_t^k - u^k\rangle]}{DT}}_{regret-term} + \underbrace{\frac{\mathbb{E}[\sum_{t=1}^{T}\langle \bar{\Delta}_t^k, \tilde{\nabla}_t^k - \bar{\nabla}_t^k\rangle]}{DT}}_{discrepancy}. \tag{3}$$

We will now average above over K epochs and bound each term. It is only the sub-optimality term that will telescope. Other terms can be bounded independent of k, i.e., bounds for noise term, regret term and discrepancy term are independent of epochs. Recall that $N := KT$.

Sub-optimality Term: Let $\gamma := f(\bar{x}_0) - \inf_x f(x)$. Summing the sub-optimality term over $k \in [K]$ and dividing by K, the sum telescopes as follows due to the initialization at the start of each epoch $k \in [K]$:

$$\frac{1}{K}\sum_{k=1}^{K}\frac{\mathbb{E}[f(\bar{x}_0^k)-f(\bar{x}_T^k)]}{DT} = \frac{f(\bar{x}_0^1)-\mathbb{E}[f(\bar{x}_T^K)]}{DTK} \leq \frac{\gamma}{DN}. \tag{4}$$

Regret Term: For the regret term, we can use Theorem 5 of Shahrampour & Jadbabaie (2018) with a fixed learning rate η, where for any $k \in [K]$:

$$R_T^k(u^k) \leq \frac{4D^2}{\eta} + \eta\Big(\frac{G^2T}{2} + \frac{2TG(L+G)\sqrt{n}}{1-\rho}\Big).$$

Choosing $\eta = \frac{8D}{c_1\sqrt{T}}$ where $c_1 = 4\sqrt{\frac{G^2(1-\rho)+4G(L+G)\sqrt{n}}{2(1-\rho)}}$ gives

$$regret-term \leq \frac{D\sqrt{T}c_1}{DT} = O(T^{-1/2}). \tag{5}$$

Discrepancy Term: For this part, we remove the superscript k for simplicity as the results hold for any $k \in [K]$. First, recall that $\|\bar{\Delta}_t\| \leq D$ since the domain $\mathcal{D}$ in Algorithm 4 is bounded. Next, we will bound $\|\tilde{\nabla}_t - \bar{\nabla}_t\|$ under the assumption that local functions f^i are L_1-Lipschitz smooth. Note that due to doubly stochasticity of P we also have $\bar{x}_t = \bar{y}_t$. Therefore,

$$\tilde{\nabla}_t - \bar{\nabla}_t = \frac{1}{n}\sum_{i=1}^{n}\int_0^1 (\nabla f^i(\bar{y}_{t-1} + s\bar{\Delta}_t) - \nabla f^i(y_{t-1,i} + s\Delta_{t,i}))ds.$$

Then, we have

$$\|\tilde{\nabla}_t - \bar{\nabla}_t\| \leq \frac{L_1}{n}\sum_{i=1}^{n}\int_0^1 \left\|\bar{y}_{t-1} + s\bar{\Delta}_t - y_{t-1,i} - s\Delta_{t,i}\right\| ds \leq \frac{L_1}{n}\sum_{i=1}^{n}\|\bar{y}_{t-1} - y_{t-1,i}\| + \frac{L_1}{2n}\sum_{i=1}^{n}\left\|\bar{\Delta}_t - \Delta_{t,i}\right\|.$$

The first term can be bounded with Lemma 4 as $\|\bar{y}_{t-1} - y_{t-1,i}\| \leq \frac{D\sqrt{n}}{1-\rho}$. We can bound the second term with $\left\|\bar{\Delta}_t - \Delta_{t,i}\right\| \leq 2D$. Hence, $\|\tilde{\nabla}_t - \bar{\nabla}_t\| \leq L_1 D c_2$ where $c_2 := \frac{\sqrt{n}}{1-\rho} + 1$. The discrepancy term can then be bounded as

$$discrepancy - term \leq DL_1c_2. \tag{6}$$

Substituting (4), (5), and (6) into (3), we get

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f^i(w_{t,i}^k)\right\|\right] \leq \frac{\gamma}{DN} + \frac{\sigma}{\sqrt{nT}} + \frac{c_1}{\sqrt{T}} + DL_1c_2. \tag{7}$$

In the left-hand side of (7), we have the average of local gradients. Using Lemma 2, we can connect this to the average of global gradients. To this end, we need r such that $\|w_{t,i}^k - \bar{w}_t^k\| \leq r$. Since $\|\bar{y}_{t-1}^k - y_{t-1,i}^k\| \leq \frac{D\sqrt{n}}{1-\rho}$, we have $\|w_{t,i}^k - \bar{w}_t^k\| \leq \|\bar{y}_{t-1}^k - y_{t-1,i}^k\| + 2D \leq D(\frac{\sqrt{n}}{1-\rho} + 2)$. Now, utilizing Lemma 2 with $r = D(\frac{\sqrt{n}}{1-\rho} + 2)$, we obtain

$$\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f(w_{t,i}^k)\right\| \leq \left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f^i(w_{t,i}^k)\right\| + DL_1\Big(\frac{2\sqrt{n}}{1-\rho} + 4\Big). \tag{8}$$

Combining (7) and (8), we obtain

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f(w_{t,i}^k)\right\|\right] \leq \frac{\gamma}{DN}+\frac{\sigma}{\sqrt{nT}}+\frac{c_1}{\sqrt{T}}+DL_1c_3,$$

where $c_3 := c_2 + \frac{2\sqrt{n}}{1-\rho} + 4 = \frac{3\sqrt{n}}{1-\rho} + 5$.

We must have $\left\|w_{t,i}^k - \bar{w}^k\right\| \leq \delta$ to bound $\left\|\nabla f(\bar{w}^k)\right\|_\delta$. We have $\left\|w_{t_1,i}^k - w_{t_2,j}^k\right\| \leq \left\|w_{t_1,i}^k - \bar{w}_{t_1}^k\right\| + \left\|\bar{w}_{t_1}^k - \bar{w}_{t_2}^k\right\| + \left\|w_{t_2,j}^k - \bar{w}_{t_2}^k\right\| \leq 2r + DT = D(\frac{2\sqrt{n}}{1-\rho} + 4 + T)$. If $T \geq 3$ and $\frac{\sqrt{n}}{1-\rho} \geq 2$, choosing $D = \frac{\delta(1-\rho)}{2T\sqrt{n}}$ guarantees that $\left\|w_{t,i}^k - \bar{w}^k\right\| \leq \delta$, and thus

$$\mathbb{E}_{k\sim\texttt{unif}[K]}\left[\left\|\nabla f(\bar{w}^k)\right\|_\delta\right] \leq \mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f(w_{t,i}^k)\right\|\right] \leq \frac{2\gamma T\sqrt{n}}{\delta N(1-\rho)}+\frac{\sigma}{\sqrt{nT}}+\frac{c_1}{\sqrt{T}}+\frac{\delta L_1(1-\rho)c_3}{2T\sqrt{n}}. \quad (9)$$

This inequality will also be used in the proof of Theorem 2 and Theorem 3. Now, we can use $\delta < 1$ and $\frac{1}{T} \leq \frac{1}{\sqrt{T}}$ to get

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f(w_{t,i}^k)\right\|\right] \leq \frac{2\gamma\sqrt{n}}{\delta N(1-\rho)}T+\frac{1}{\sqrt{T}}\left(\frac{\sigma}{\sqrt{n}}+c_1+L_1\frac{1-\rho}{2\sqrt{n}}c_3\right).$$

Choosing $T = c_4\left(\delta N\right)^{\frac{2}{3}}$ where $c_4 := \left(\frac{(1-\rho)(2\sigma+2c_1\sqrt{n}+L_1(1-\rho)c_3)}{8\gamma n}\right)^{\frac{2}{3}}$, we have

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f(w_{t,i}^k)\right\|\right] \leq (\delta N)^{-\frac{1}{3}}\left(\frac{2\gamma\sqrt{n}c_4}{1-\rho}+\frac{1}{\sqrt{c_4}}\left(\frac{\sigma}{\sqrt{n}}+c_1+L_1\frac{1-\rho}{2\sqrt{n}}c_3\right)\right) = c_5(\delta N)^{-\frac{1}{3}},$$

where $c_5 := \frac{6\gamma\sqrt{n}}{1-\rho}c_4 = \frac{6\gamma\sqrt{n}}{1-\rho}\left(\frac{(1-\rho)(2\sigma+2c_1\sqrt{n}+L_1(1-\rho)c_3)}{8\gamma n}\right)^{\frac{2}{3}} = \frac{3}{2}\left(\frac{\gamma(2\sigma+2c_1\sqrt{n}+L_1(1-\rho)c_3)^2}{(1-\rho)\sqrt{n}}\right)^{\frac{1}{3}}$. In terms of the network connectivity measure $1-\rho$, $c_5 = O((1-\rho)^{-\frac{2}{3}})$. As a result, we derive

$$\mathbb{E}_{k\sim\texttt{unif}[K]}\left[\left\|\nabla f(\bar{w}^k)\right\|_\delta\right] \leq \mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f(w_{t,i}^k)\right\|\right] \leq O((\delta N)^{-\frac{1}{3}}),$$

which means that given $\{\delta, \epsilon, \rho\}$, we can find a (δ, ϵ)-stationary point in $N = \Theta\big(\delta^{-1}\epsilon^{-3}(1-\rho)^{-2}\big)$ rounds. For smooth functions $\delta = O(\epsilon)$, so the overall rate matches the optimal rate of $N = O(\epsilon^{-4})$. □

A.4. Proof of Theorem 2

Proof. Recall from Proposition 1 that $(f^i)_\delta$ and in turn f_δ have L_1-Lipschitz smooth gradients with smoothness parameter $L_1 = cL\sqrt{d}\delta^{-1}$, where L is due to Lipschitz continuity of the original functions f^i and f. Now, in Equation (9) replacing δ by $\frac{\delta}{2}$ and f by $f_{\frac{\delta}{2}}$, the smoothness parameter becomes $L_1 = \frac{2cL\sqrt{d}}{\delta}$, which yields

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f_{\frac{\delta}{2}}(w_{t,i}^k)\right\|\right] \leq \frac{4\gamma' T\sqrt{n}}{\delta N(1-\rho)}+\frac{G}{\sqrt{nT}}+\frac{c_1}{\sqrt{T}}+\frac{cL\sqrt{d}(1-\rho)c_3}{2T\sqrt{n}}, \quad (10)$$

where $\gamma' := \gamma + L$ due to the approximation error incurred in (4), and σ is also replaced by G. For the last term we can use $\frac{1}{T} \leq \frac{1}{\sqrt{T}}$ to get

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f_{\frac{\delta}{2}}(w_{t,i}^k)\right\|\right] \leq \frac{4\gamma'\sqrt{n}}{\delta N(1-\rho)}T+\frac{1}{\sqrt{T}}\left(\frac{G}{\sqrt{n}}+c_1+\frac{cL\sqrt{d}(1-\rho)c_3}{2\sqrt{n}}\right).$$

Choosing $T = c_7\left(\delta N\right)^{\frac{2}{3}}$ where $c_7 := \left(\frac{(1-\rho)(2G+2c_1\sqrt{n}+cL\sqrt{d}(1-\rho)c_3)}{16\gamma' n}\right)^{\frac{2}{3}}$, we have

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f_{\frac{\delta}{2}}(w_{t,i}^k)\right\|\right] \leq (\delta N)^{-\frac{1}{3}}\left(\frac{4\gamma'\sqrt{n}c_7}{(1-\rho)}+\frac{1}{\sqrt{c_7}}\left(\frac{G}{\sqrt{n}}+c_1+\frac{cL\sqrt{d}(1-\rho)}{2\sqrt{n}}c_3\right)\right) = c_8(\delta N)^{-\frac{1}{3}},$$

where $c_8 := \frac{12\gamma'\sqrt{n}}{1-\rho}c_7 = \frac{12\gamma'\sqrt{n}}{1-\rho}\left(\frac{(1-\rho)(2G+2c_1\sqrt{n}+cL\sqrt{d}(1-\rho)c_3)}{16\gamma' n}\right)^{\frac{2}{3}} = 3\left(\frac{\gamma'(2G+2c_1\sqrt{n}+cL\sqrt{d}(1-\rho)c_3)^2}{4(1-\rho)\sqrt{n}}\right)^{\frac{1}{3}}$. Using Proposition 2 on the left-hand side of above completes the proof. In terms of the network connectivity measure $1-\rho$, we have $c_8 = O((1-\rho)^{-\frac{2}{3}})$. □

A.5. Proof of Theorem 3

Proof. Similar to the proof of Theorem 2, in Equation (9), we replace δ by $\frac{\delta}{2}$ and f by $f_{\frac{\delta}{2}}$, so the smoothness parameter becomes $L_1 = \frac{2cL\sqrt{d}}{\delta}$. Applying Lemma 3, we can bound σ by $\sqrt{16\sqrt{2\pi}dL^2}$ as well. Therefore, we have

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f_{\frac{\delta}{2}}(w_{t,i}^k)\right\|\right] \leq \frac{4\gamma' T\sqrt{n}}{\delta N(1-\rho)} + \frac{\sqrt{16\sqrt{2\pi}dL^2}}{\sqrt{nT}} + \frac{c_{10}}{\sqrt{T}} + \frac{cL\sqrt{d}(1-\rho)c_3}{2T\sqrt{n}},$$

where $c_{10} := 4\sqrt{\frac{c_9^2(1-\rho)+4c_9(L+c_9)\sqrt{n}}{2(1-\rho)}}$ and $c_9 := \sqrt{16\sqrt{2\pi}dL^2}$.

If we follow similar steps as in the proof of Theorem 2, we have the following result

$$\mathbb{E}\left[\frac{1}{K}\sum_{k=1}^{K}\left\|\frac{1}{nT}\sum_{t=1}^{T}\sum_{i=1}^{n}\nabla f_{\frac{\delta}{2}}(w_{t,i}^k)\right\|\right] \leq c_{11}(\delta N)^{-\frac{1}{3}},$$

where $c_{11} := 3\left(\frac{\gamma'(2\sqrt{16\sqrt{2\pi}dL^2}+2c_{10}\sqrt{n}+cL\sqrt{d}(1-\rho)c_3)^2}{4(1-\rho)\sqrt{n}}\right)^{\frac{1}{3}}$. Using Proposition 2 on the left-hand side of above completes the proof. In terms of the network connectivity measure $1-\rho$ and ambient dimension d, we have $c_{11} = O(d^{\frac{1}{3}}(1-\rho)^{-\frac{2}{3}})$. □

B. Constant Terms

The smoothness parameter of f_δ is $\kappa\frac{d!!}{(d-1)!!}\frac{L}{\delta}$, where $\kappa = \frac{2}{\pi}$ if d is even, and $\kappa = 1$ otherwise. Thus, the geometric constant $c := \kappa\frac{1}{\sqrt{d}}\frac{d!!}{(d-1)!!}$. We note that $\lim_{d\to\infty} c = \lim_{d\to\infty}\kappa\frac{1}{\sqrt{d}}\frac{d!!}{(d-1)!!} = \sqrt{\frac{\pi}{2}}$ (Yousefian et al., 2012). Here, we summarize the constant terms used throughout the proofs:

$$\begin{aligned}
c_1 &= 4\sqrt{\frac{G^2(1-\rho)+4G(L+G)\sqrt{n}}{2(1-\rho)}}\\
c_3 &= \frac{3\sqrt{n}}{1-\rho}+5\\
c_5 &= \frac{3}{2}\left(\frac{\gamma(2\sigma+2c_1\sqrt{n}+L_1(1-\rho)c_3)^2}{(1-\rho)\sqrt{n}}\right)^{\frac{1}{3}}\\
c_8 &= 3\left(\frac{\gamma'(2G+2c_1\sqrt{n}+cL\sqrt{d}(1-\rho)c_3)^2}{4(1-\rho)\sqrt{n}}\right)^{\frac{1}{3}}\\
c_9 &= \sqrt{16\sqrt{2\pi}dL^2}\\
c_{10} &= 4\sqrt{\frac{c_9^2(1-\rho)+4c_9(L+c_9)\sqrt{n}}{2(1-\rho)}}\\
c_{11} &= 3\left(\frac{\gamma'(2\sqrt{16\sqrt{2\pi}dL^2}+2c_{10}\sqrt{n}+cL\sqrt{d}(1-\rho)c_3)^2}{4(1-\rho)\sqrt{n}}\right)^{\frac{1}{3}}
\end{aligned}$$

C. Numerical Experiments Results

In this section, we present the plots of test accuracy comparisons (Figs. 5-6), described in our experiments.

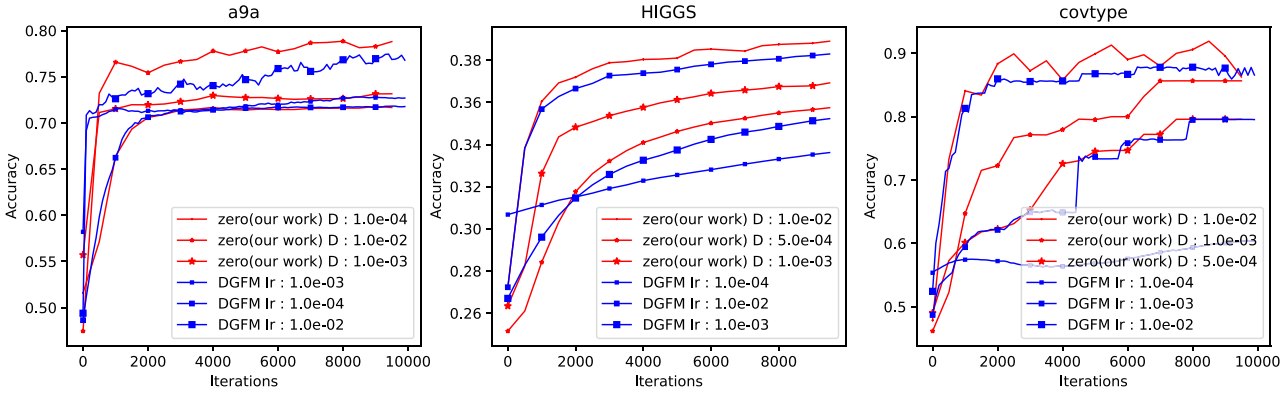

Figure 5. Evaluation of the test accuracy of our algorithm and DGFM in the zero-order setting.

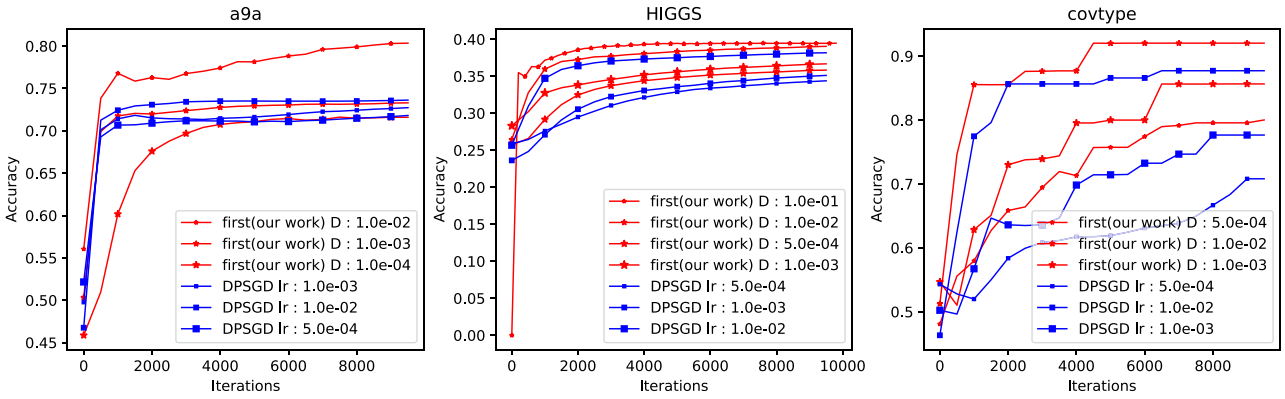

Figure 6. Evaluation of the test accuracy of our algorithm and DPSGD in the first-order setting.