Stochastic Decentralized Optimization of Non-Smooth Convex and Convex-Concave Problems over Time-Varying Networks

Maxim Divilkovskiy[1,2], **Alexander Gasnikov**[2,1,3]
[1]MIPT
[2]Research Center of the Artificial Intelligence Institute, Innopolis University, Innopolis, Russia
[3]ISP RAS
divilkovskii.mm@phystech.edu, avgasnikov@gmail.com

Abstract

We study non-smooth stochastic decentralized optimization problems over time-varying networks, where objective functions are distributed across nodes and network connections may intermittently appear or break. Specifically, we consider two settings: (i) stochastic non-smooth (strongly) convex optimization, and (ii) stochastic non-smooth (strongly) convex–(strongly) concave saddle point optimization. Convex problems of this type commonly arise in deep neural network training, while saddle point problems are central to machine learning tasks such as the training of generative adversarial networks (GANs). Prior works have primarily focused on the smooth setting, or time-invariant network scenarios. We extend the existing theory to the more general non-smooth and stochastic setting over time-varying networks and saddle point problems. Our analysis establishes upper bounds on both the number of stochastic oracle calls and communication rounds, matching lower bounds for both convex and saddle point optimization problems.

Extended version with supplementary material — https://arxiv.org/abs/2508.09029

1 Introduction

Distributed optimization is an important area in modern optimization. It has many applications in power control (Gan, Topcu, and Low 2013), vehicle control (Wang and Hu 2010), resource allocation (Beck et al. 2014), cooperative optimization (Nedić and Ozdaglar 2009), and, most notably, machine learning (Rabbat and Nowak 2004; Forero, Cano, and Giannakis 2010; Galakatos, Crotty, and Kraska 2018). The rapid growth in the number of model parameters created a demand for running algorithms on several nodes. Another direction is machine learning with privacy constraints (Ram, Veeravalli, and Nedić 2009), which require separating data between servers. We study the decentralized setting of distributed optimization. In this scenario, all the nodes are equal and do not differ from each other. Another property of decentralized optimization is that communication between nodes may not be precisely scheduled.

Decentralized optimization consists in optimizing a function f, which can be represented as a sum of functions: $f = \sum_{i=1}^{n} f_i$, where each function f_i is stored in a distinct node. In the decentralized time-varying setting, connections between nodes may appear or break in the process of optimization. The time-static optimization case is covered extensively in recent works (Gorbunov et al. 2022; Dvinskikh and Gasnikov 2021; Scaman et al. 2017, 2018), however the development of time-varying optimization started in the recent years. This setting poses more complex communication scheme than time-static one due to instability in connections.

In this research, we focus on the non-smooth stochastic formulation of convex minimization and convex-concave saddle point problems. Algorithms for saddle point problems are motivated by different modern machine learning approaches like GANs (Goodfellow et al. 2014; Gidel et al. 2020) and reinforcement learning (Jin and Sidford 2020; Omidshafiei et al. 2017; Wai et al. 2019). Other applications are optimal transport (Jambulapati, Sidford, and Tian 2019) and economics (Facchinei and Pang 2007). Most recent research on distributed optimization, including saddle point problems assume smoothness of the considered functions (Rogozin et al. 2024; Metelev et al. 2022). In this paper, we do not assume this restriction since without smoothness we can solve larger scope of problems.

The non-smooth setting of convex deterministic decentralized optimization over time-varying graphs was studied in (Kovalev et al. 2024). In this paper, we extend their algorithm to handle arbitrary stochastic monotone operators. Thus, our contributions include establishing the first optimal convergence rates for stochastic decentralized non-smooth convex problems, as well as for both deterministic and stochastic decentralized non-smooth saddle point problems over time-varying graphs. The contributions are summarized in Table 1.

Problem	Deterministic	Stochastic
Convex	(Kovalev et al. 2024)	Theorems 2,3 (this paper)
Saddle Point	Corollary of Theorems 2,3 (this paper)	Theorems 2,3 (this paper)

Table 1: Summary of contributions.

2 Problem Statement

We consider the following two stochastic decentralized optimization problems.

Stochastic decentralized (strongly) convex non-smooth optimization:

$$\min_{x\in\mathbb{R}^d}\left[p(x)=\frac{1}{n}\sum_{i=1}^{n}f_i(x)+\frac{r}{2}\|x\|^2\right]. \tag{1}$$

Stochastic decentralized (strongly) convex-(strongly) concave non-smooth optimization:

$$\min_{\xi\in\mathbb{R}^{d_\xi}}\max_{\zeta\in\mathbb{R}^{d_\zeta}}\left[p(\xi,\zeta)=\frac{1}{n}\sum_{i=1}^{n}f_i(\xi,\zeta)+\frac{r}{2}\|\xi\|^2-\frac{r}{2}\|\zeta\|^2\right]. \tag{2}$$

Throughout the paper, we define the solution space $\mathcal{H}$ as follows: $\mathcal{H}=\mathbb{R}^d$ for problem (1) and $\mathcal{H}=\mathbb{R}^{d_\xi+d_\zeta}$ for problem (2). We denote by $\|\cdot\|$ the standard Euclidean norm in $\mathcal{H}$, and by $\langle\cdot,\cdot\rangle$ the standard inner product in $\mathcal{H}$.

For both problems (1) and (2), we assume $r\geq 0$. When $r>0$, the problem is referred to as *strongly monotone*; when $r=0$, it is referred to as *monotone*.

Remark 1. *We also study strongly convex-strongly concave problems with different constants of strong convexity and strong concavity:*

$$p(\xi,\zeta)=\frac{1}{n}\sum_{i=1}^{n}f_i(\xi,\zeta)+\frac{r_\xi}{2}\|\xi\|^2-\frac{r_\zeta}{2}\|\zeta\|^2. \tag{3}$$

We obtain results for this case as a corollary of the symmetric case.

To ensure well-posedness of the considered problems and to enable convergence analysis, we impose standard convexity and convexity-concavity conditions on the objective functions:

Assumption 1 (Convexity for the problem (1))**.** *Each function*

$$f_i(x):\mathbb{R}^d\to\mathbb{R}$$

is convex in x.

Assumption 2 (Convexity-concavity for the problem (2))**.** *Each function*

$$f_i(\xi,\zeta):\mathbb{R}^{d_\xi}\times\mathbb{R}^{d_\zeta}\to\mathbb{R}$$

is convex in ξ for each fixed ζ, and concave in ζ for each fixed ξ.

Remark 2. *Any strongly convex or strongly convex-concave problem can be brought into the form of problems* (1) *and* (3)*, respectively, by appropriately choosing the regularization parameters. In particular, any $\mu-$strongly convex function f can be rewritten as*

$$\left(f(x)-\frac{\mu}{2}\|x\|^2\right)+\frac{\mu}{2}\|x\|^2.$$

A similar transformation applies to strongly convex–strongly concave functions.

We further assume the existence of a solution for both problems.

Assumption 3 (Existence of solution)**.** *For problems* (1)*,* (2) *there exists a solution x^* such that, for some distance $R>0$,*

$$\|x^*\|\leq R. \tag{4}$$

This assumption is crucial for non-strongly monotone problems, where a solution may not exist in general. In the case of strongly monotone problems, the solution always exists and is unique. We also require this constant R for our convergence analysis.

To unify the analysis of both problems, we define operators associated with each problem.

Definition 1. *Let $x\in\mathcal{H}$ be arbitrary. For problem* (1)*, define the associated operator as*

$$\mathbf{T}_i(x)=\partial f_i(x). \tag{5}$$

For problem (2)*, define it as*

$$\mathbf{T}_i(x)=[\partial_\xi f_i(\xi,\zeta),\ -\partial_\zeta f_i(\xi,\zeta)], \tag{6}$$

where $x=(\xi,\zeta)$.

We use the notation

$$\mathbf{T}(x)=(\mathbf{T}_1(x),\ldots,\mathbf{T}_n(x)).$$

This definition allows us to treat both optimization and saddle point problems within a unified analysis. In the convex minimization case (1), each operator $\mathbf{T}_i$ coincides with the subdifferential mapping of the corresponding convex function f_i, which is a set-valued monotone operator. For the saddle point problem (2), the operator $\mathbf{T}_i$ collects the partial subdifferentials with respect to the primal variable ξ and the negative dual variable ζ, capturing the first-order stationarity condition. Further, we assume that these operators are bounded, which is equivalent to the Lipschitz continuity of the underlying functions.

Assumption 4. *Let x be arbitrary, and let $g_i\in\mathbf{T}_i(x)$, where $\mathbf{T}_i$ is defined in Definition 1. Then, for all $i\in\{1,\ldots,n\}$,*

$$\|g_i\|\leq M. \tag{7}$$

To estimate the convergence rate for these problems, we introduce gap functions for each problem. These gap functions measure how close our result to the solution x^*, which exists due to the Assumption 3.

Definition 2 (Gap function for the problem (1))**.**

$$G_{\mathrm{CVX}}(x_o)=p(x_o)-p(x^*). \tag{8}$$

Definition 3 (Gap function for the problem (2))**.**

$$G_{\mathrm{SPP}}(x_o)=p(\xi_o,\zeta^*)-p(\xi^*,\zeta_o), \tag{9}$$

where $x_o=(\xi_o,\zeta_o), x=(\xi,\zeta)$.

It is well known that problem (1) is a special case of a problem (2). With introduced gap functions, lower bound for convex optimization will also be the lower bound for saddle point optimization. The upper bound for the saddle point optimization will also be the upper bound for convex optimization. Thus, if both bounds coincide, we can conclude that these gaps are optimal and cannot be improved.

In our convergence rate analysis, we determine the number of communications and oracle calls required to ensure that the expected gap function is bounded by ε. Due to the stochastic nature of the problem, we analyze the expectation of the gap function, meaning the convergence rate guarantees $\mathbb{E}\left[G_{\mathrm{CVX}}(x_o)\right]\leq\varepsilon$ or $\mathbb{E}\left[G_{\mathrm{SPP}}(x_o)\right]\leq\varepsilon$.

3 Stochastic Decentralized Setting

The design of deterministic decentralized algorithms for time-static networks is provided in (Scaman et al. 2017) for smooth problems and in (Scaman et al. 2018) for non-smooth problems. However, their algorithms rely on a dual oracle, which may be inaccessible in practical implementations. In contrast, (Kovalev, Salim, and Richtarik 2020) proposed a reformulation of the decentralized problem as a Forward-Backward algorithm, achieving the optimal convergence rate in the time-static scenario while using only a primal oracle.

This idea was later extended to time-varying graphs, attaining optimal convergence rates with a deterministic primal oracle for both smooth (Kovalev et al. 2021) and non-smooth (Kovalev et al. 2024) settings.

In our paper, we follow this reformulation, extending the analysis to stochastic primal settings and saddle-point problems.

We start by formalizing the time-varying optimization setting. At a fixed moment of time, a communication network may be represented as a graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \dots, n\}$ is a set of nodes and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is a set of edges between these nodes. In our scenario, the connections may change over time. Therefore, we represent the time-varying communication network as a function of a continuous time parameter $\tau > 0$ as a function $\mathcal{G}(\tau) = (\mathcal{V}, \mathcal{E}(\tau))$, where $\mathcal{E}(\tau) : \mathbb{R}_+ \to 2^{\mathcal{V}\times\mathcal{V}}$ denotes the time-varying set of edges.

Next, we formalize the mechanism of node interaction, which is commonly modeled through *gossip matrix* multiplication. In the time-static setting, the gossip matrix remains constant throughout the execution of the algorithm and corresponds to the fixed structure of the underlying communication network. In contrast, the time-varying setting poses additional challenges, as the network topology evolves over time, making it inappropriate to associate a single fixed matrix with the entire process.

We represent the gossip matrix as a matrix-valued function

$$\mathbf{W}(\tau) : \mathbb{R}_+ \to \mathbb{R}^{n\times n},$$

which satisfies Assumption 5. See Figure 1 for the example of a time-varying graph.

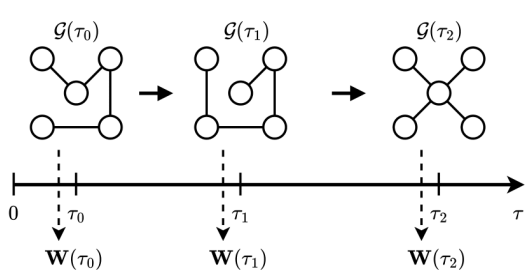

Figure 1: Illustration of a time-varying communication graph. At each moment in time, the associated gossip matrix reflects the current network configuration.

Assumption 5. *For all $\tau \geq 0$, the gossip matrix $\mathbf{W}(\tau) \in \mathbb{R}^{n\times n}$ associated with the time-varying communication network $\mathcal{G}(\mathcal{V}, \mathcal{E}(\tau))$ satisfies the following properties:*

1. $\mathbf{W}(\tau)_{ij} = 0$ *if* $i \neq j$ *and* $(j,i) \notin \mathcal{E}(\tau)$,
2. $\mathbf{W}(\tau)\mathbf{1}_n = 0$ *and* $\mathbf{W}(\tau)^\top \mathbf{1}_n = 0$.

The first condition encodes the structure of the network. The second implies that the gossip step converges to the average over the whole network. A common choice for the gossip matrix is a Laplacian matrix of a graph $\mathcal{G}(\tau)$. This definition allows us to design communication step of the form $x^{k+1} = x^k - \eta\mathbf{W}(\tau)x^k$, which is consistent with standard decentralized optimization algorithms.

For the convergence analysis, we also introduce a condition number χ for the time-varying network as follows.

Assumption 6. *There exists a constant $\chi \geq 1$ such that the following inequality holds for all $\tau \geq 0$:*

$$\|\mathbf{W}(\tau)x - x\|^2 \leq \left(1 - \frac{1}{\chi}\right)\|x\|^2 \quad \text{for all } x \in \left\{x \in \mathbb{R}^n : \sum_{i=1}^n x_i = 0\right\}. \tag{10}$$

The constant $\chi \geq 1$ quantifies the connectivity of the time-varying network. A larger value of χ indicates poorer connectivity and results in more iterations required for the algorithm to converge.

Next, we define the stochastic decentralized oracle. In the deterministic setting, this coincides with the standard definition of a decentralized first-order oracle. Stochasticity arises from allowing some deviation from the true operator value. We define a first-order oracle for both problems simultaneously by treating it as an oracle that returns a stochastic approximation of the corresponding operator.

Definition 4 (Stochastic decentralized oracle)**.** *For arbitrary x, let $\mathbf{T}_i$ be defined as in Definition 1 for the problems* (1) *and* (2)*. A random vector $\tilde{\mathbf{g}}_i(x) : \mathcal{H} \to \mathcal{H}$ is called a stochastic operator oracle associated with operator $\mathbf{T}_i(x)$ if there exists $\sigma > 0$ such that for any $x \in \mathcal{H}$ the following holds:*

$$\mathbb{E}\left[\tilde{\mathbf{g}}_i(x)\right] = \mathbf{g}_i \in \mathbf{T}_i; \tag{11}$$

$$\mathbb{E}\left[\|\tilde{\mathbf{g}}_i(x) - \mathbf{g}_i\|_2^2\right] \leq \sigma^2. \tag{12}$$

Here, $\mathbf{T}_i(x)$ is the true subdifferential, $\mathbf{g}_i(x)$ is the subgradient and $\tilde{\mathbf{g}}_i(x)$ is a stochastic subgradient.

4 Decentralized Reformulation and the Algorithm

A reformulation of the convex problem in the deterministic setting was presented in (Kovalev et al. 2024). In this paper, we follow a similar approach but generalize the algorithm to arbitrary monotone operators. Specifically, we extend the algorithm to handle any stochastic monotone operator instead of relying on subgradients of convex functions. Unlike mentioned paper, which is based on a saddle point reformulation of the convex minimization problem, we bypass this step by

Algorithm 1

input: $x^0 = x^{-1} = \tilde{x}^0 \in \mathcal{H}^n$, $y^0 = \bar{y}^0 \in \mathcal{H}^n$, $z^0 = \bar{z}^0 \in \mathcal{L}^\perp$, $m^0 \in \mathcal{H}^n$
parameters: $K,T \in \{1,2,\ldots\}$,
$\{(\alpha_k,\beta_k,\gamma_k,\sigma_k,\lambda_k,\tau_x^k,\eta_x^k,\eta_y^k,\eta_z^k,\theta_z^k)\}_{k=0}^{K-1} \subset \mathbb{R}_+^{10}$
for $k = 0,1,\ldots,K-1$ **do**
  $\underline{y}^k = \alpha_k y^k + (1-\alpha_k)\overline{y}^k, \quad \underline{z}^k = \alpha_k z^k + (1-\alpha_k)\overline{z}^k$
  $g_y^k = \nabla_y G(\underline{y}^k,\underline{z}^k), \quad g_z^k = \nabla_z G(\underline{y}^k,\underline{z}^k)$
  $\tilde{g}_z^k = (\mathbf{W}(\tau)\otimes \mathbf{I}_d) g_z^k, \quad \hat{g}_z^k = (\mathbf{W}(\tau)\otimes \mathbf{I}_d)(g_z^k + m^k)$ ▷ Decentralized communication
  $y^{k+1} = y^k - \eta_y^k(g_y^k + \hat{x}^{k+1}), \quad z^{k+1} = z^k - \eta_z^k \hat{g}_z^k, \quad \hat{x}^{k+1} = x^k + \gamma_k(\tilde{x}^k - x^{k-1})$
  $\overline{y}^{k+1} = \underline{y}^k + \alpha_k(y^{k+1} - y^k), \quad \overline{z}^{k+1} = \underline{z}^k - \theta_z^k \tilde{g}_z^k, \quad m^{k+1} = (\eta_z^k/\eta_z^{k+1})(m^k + g_z^k - \hat{g}_z^k)$
  $x^{k,0} = x^k$
  for $t = 0,1,\ldots,T-1$ **do**
    $g_x^{k,t} = (\tilde{\mathbf{g}}_1(x_1^{k,t}),\ldots,\tilde{\mathbf{g}}_n(x_n^{k,t}))$ ▷ Stochastic oracle call
    $x^{k,t+1} = x^{k,t} - \eta_x^k\left(g_x^{k,t} + \beta_k x^{k,t+1} - y^{k+1} + \tau_x^k(x^{k,t+1} - x^k)\right)$
  end for
  $x^{k+1} = \sigma_k x^{k,T} + (1-\sigma_k)\tilde{x}^{k+1}, \quad \tilde{x}^{k+1} = \frac{1}{T}\sum_{t=1}^T x^{k,t}, \quad \overline{x}^{k+1} = \alpha_k \tilde{x}^{k+1} + (1-\alpha_k)\overline{x}^k$
end for
$(x_a^K, y_a^K, z_a^K) = (\sum_{k=1}^K \lambda_k)^{-1}\sum_{k=1}^K \lambda_k(\overline{x}^k,\overline{y}^k,\overline{z}^k)$
output: $x_o^K = \frac{1}{n}\sum_{i=1}^n x_{a,i}^K \in \mathcal{H}$, where $(x_{a,1}^K,\ldots,x_{a,n}^K) = x_a^K \in \mathcal{H}^n$

directly reformulating both problems as a monotone inclusion problems.

First, we perform a standard distributed reformulation. Specifically, denote the consensus space

$$\mathcal{L} = \{(x_1,\ldots,x_n) \in \mathcal{H}^n : x_1 = \ldots = x_n\}. \tag{13}$$

In the analysis we will also need the fact that

$$\mathcal{L}^\perp = \{(x_1,\ldots,x_n) \in \mathcal{H}^n : \sum_{i=1}^n x_i = 0\}. \tag{14}$$

Hence, the problem (1) is equivalent to the following problem:

$$\min_{x\in\mathcal{H}} \left[\frac{1}{n}\sum_{i=1}^n f_i(x_i) + \frac{r}{2n}\|x\|^2\right] \text{ subject to } \quad x \in \mathcal{L}. \tag{15}$$

The problem (2) is equivalent to the following problem:

$$\min_{\xi\in\mathbb{R}^{d_\xi}} \max_{\zeta\in\mathbb{R}^{d_\zeta}} \left[\frac{1}{n}\sum_{i=1}^n f_i(\xi_i,\zeta_i) + \frac{r}{2n}\|\xi\|^2 - \frac{r}{2n}\|\zeta\|^2\right] \tag{16}$$
$$\text{subject to } (\xi,\zeta) = x \in \mathcal{L}.$$

Now, we incorporate consensus via communication into the optimization problem. Let $\mathbf{T}_i$ be defined as in Definition 1. Define

$$\mathbf{T}(x) = \begin{bmatrix}\mathbf{T}_1(x_1)\\ \vdots \\ \mathbf{T}_n(x_n)\end{bmatrix} + r_x x : \mathcal{H}^n \to (2^{\mathcal{H}})^n; \tag{17}$$

$$G(y,z) = \frac{r_{yz}}{2}\|y+z\|^2 : \mathcal{H}^n \times \mathcal{H}^n \to \mathcal{H}, \tag{18}$$

where,

$$x = (x_1,\ldots,x_n) \in \mathcal{H}^n,$$

$$\text{and} \quad r_x, r_{yz} > 0 \text{ satisfy } r_x + \frac{1}{r_{yz}} = r.$$

Let $\mathsf{E} = \mathcal{H}^n \times \mathcal{H}^n \times \mathcal{L}^\perp$ be an Euclidean space. Consider the operators

$$A(u) = \begin{bmatrix}0\\ \nabla_y G(y,z)\\ \mathbf{P}\nabla_z G(y,z)\end{bmatrix}; \quad B(u) = \begin{bmatrix}\mathbf{T}(x) - y\\ x\\ 0\end{bmatrix},$$

where

$$\mathbf{P} = (\mathbf{I}_n - (1/n)\mathbf{1}_n\mathbf{1}_n^\top)\otimes \mathbf{I}_d$$

for the problem (1) and

$$\mathbf{P} = (\mathbf{I}_n - (1/n)\mathbf{1}_n\mathbf{1}_n^\top)\otimes \mathbf{I}_{d_\xi + d_\zeta}$$

for the problem (2). $\mathbf{P}$ is the orthogonal projection matrix onto $\mathcal{L}^\perp$. Then, operator A is a monotone operator and corresponds to the gradient of a smooth convex function. Operator B is a monotone set-valued operator.

Consider the following monotone inclusion problem:

$$\text{find } u \in \mathsf{E} \text{ such that } 0 \in A(u) + B(u). \tag{19}$$

This problem can be solved using the Forward-Backward Algorithm with Nesterov acceleration (see (Kovalev, Salim, and Richtarik 2020) for a similar approach). The acceleration relies on the fact that the operator A is a gradient of a smooth function. Since the operator $\mathbf{T}$ is not a gradient of a smooth function we have to place it into the operator B. This reformulation is a key to the presented algorithm. The following theorem establishes the equivalence of this reformulation.

Theorem 1. *Problem* (19) *is equivalent to problems* (1) *and* (2) *with the corresponding definitions of* $\mathbf{T}_i$ *for each of the problems.*

The proof of this theorem is provided in the Supplementary Materials.

Multiplication by the matrix $\mathbf{P}$ corresponds to projecting onto the consensus space. This, in turn, means averaging values across the entire network, which is challenging in the time-varying setting due to changing connectivity. To address this, the algorithm replaces global averaging via $\mathbf{P}$ with local averaging using a matrix $\mathbf{W}$, where each multiplication by this matrix corresponds to averaging over the immediate neighbors of each node in the network graph.

As mentioned earlier, a similar reformulation was introduced in (Kovalev et al. 2021). However, in their setting, the functions were smooth, and thus the operator $\mathbf{T}$ corresponded to the gradient of a smooth function. When incorporated into the operator A, convergence could be accelerated using Nesterov's acceleration. Based on this, they proposed the ADOM algorithm, which achieves the optimal convergence rate for their setting.

In our case, the operator $\mathbf{T}$ is incorporated into B, while iterations over operator A are accelerated using Nesterov's method. The iterations involving operator B cannot be further accelerated, either for convex or saddle-point problems. This aligns with classical results in non-smooth, non-distributed optimization.

With this setup, we show that Algorithm 1 converges to the desired solution. The algorithm introduces an additional input variable m, corresponding to the error-feedback mechanism. The y and z updates are accelerated using Nesterov's acceleration.

The inner loop over T corresponds to gradient descent for problem (1), and to gradient descent–ascent for problem (2). The algorithm requires K decentralized communication rounds and $K \times T$ stochastic oracle calls.

5 Optimal Convergence Rate

In this section we assume that Assumptions 3 to 6 hold. For the problem (1) Assumption 1 holds, for the problem (2) Assumption 2 holds.

In (Kovalev et al. 2024), the authors provide a lower bound for the deterministic non-smooth decentralized convex minimization problem over time-varying networks. By using their analysis and combining it with the classical lower bound for non-smooth convex optimization in the non-distributed setting (Bubeck 2015), we obtain the following lower bounds.

For concise formulation, we denote the optimality gap G as

$$G := \begin{cases} G_{\mathrm{CVX}}, & \text{for the convex minimization problem (1),} \\ G_{\mathrm{SPP}}, & \text{for the saddle-point problem (2).} \end{cases}$$

Proposition 1 (Lower bound for problems (1) and (2) in the strongly monotone case)**.** *Let $r > 0$. Then, for arbitrary $\varepsilon > 0$ there exists an optimization problem and a time-varying network such that Algorithm 1 requires at least*

$$\Omega\left(\frac{\chi M}{\sqrt{r\varepsilon}}\right) \textit{ decentralized communications}$$

and

$$\Omega\left(\frac{(M+\sigma)^2}{r\varepsilon}\right) \textit{ oracle calls}$$

to achieve $\mathbb{E}\left[G(x_o^K)\right] \leq \varepsilon$.

Proposition 2 (Lower bound for problems (1) and (2) in the monotone case)**.** *Let $r = 0$. Then, for arbitrary $\varepsilon > 0$ there exists an optimization problem and a time-varying network such that Algorithm 1 requires at least*

$$\Omega\left(\frac{\chi M R}{\varepsilon}\right) \textit{ decentralized communications}$$

and

$$\Omega\left(\frac{(M+\sigma)^2 R^2}{\varepsilon^2}\right) \textit{ oracle calls}$$

to achieve $\mathbb{E}\left[G(x_o^K)\right] \leq \varepsilon$.

These lower bounds match the corresponding lower bounds for the non-distributed setting as well as for deterministic decentralized non-smooth convex optimization.

We now present the following theorems on the convergence rate of Algorithm 1.

Theorem 2 (Upper bound for problems (1) and (2) in the strongly monotone case)**.** *Let $r > 0$. Then, for arbitrary $\varepsilon > 0$ Algorithm 1, requires*

$$\mathcal{O}\left(\frac{\chi M}{\sqrt{r\varepsilon}}\right) \textit{ decentralized communications}$$

and

$$\mathcal{O}\left(\frac{(M+\sigma)^2}{r\varepsilon}\right) \textit{ oracle calls}$$

to achieve $\mathbb{E}\left[G(x_o^K)\right] \leq \varepsilon$.

Theorem 3 (Upper bound for problems (1) and (2) in the monotone case)**.** *Let $r = 0$. Then, for arbitrary $\varepsilon > 0$ Algorithm 1, requires*

$$\mathcal{O}\left(\frac{\chi M R}{\varepsilon}\right) \textit{ decentralized communications}$$

and

$$\mathcal{O}\left(\frac{(M+\sigma)^2 R^2}{\varepsilon^2}\right) \textit{ oracle calls}$$

to achieve $\mathbb{E}\left[G(x_o^K)\right] \leq \varepsilon$.

The proofs of these theorems are provided in the Supplementary Materials.

These upper bounds match the corresponding lower bounds in Propositions 1 and 2, thus establishing the optimality of these convergence rates.

In the case of saddle-point problems, the strong convexity and strong concavity constants may differ. The following result addresses this asymmetric setting.

Corollary 1 (Complexity for saddle point problems with different constants of strong convexity and strong concavity)**.** *Consider the problem of type* (3)*. Let* $r_\xi > 0$*,* $r_\zeta > 0$*. Then, for arbitrary* $\varepsilon > 0$*, Algorithm 1 requires*

$$\mathcal{O}\left(\frac{\chi M}{\sqrt{\varepsilon}}\sqrt{\frac{1}{r_\xi}+\frac{1}{r_\zeta}}\right) \text{ decentralized communications}$$

and

$$\mathcal{O}\left(\frac{(M+\sigma)^2}{\varepsilon}\left(\frac{1}{r_\xi}+\frac{1}{r_\zeta}\right)\right) \text{ oracle calls,}$$

to achieve $\mathbb{E}\left[G_{\mathrm{SPP}}(x_o^K)\right] \le \varepsilon$*.*

The proof of this corollary is provided in the Supplementary Materials.

6 Experiments

We conduct experiments on synthetic random graphs constructed using the Erdős–Rényi model. We consider both time-static and time-varying settings. Our evaluation focuses on two metrics: the Euclidean distance to the objective,

$$\left\|x_o^K - x^*\right\|,$$

and the saddle point residual,

$$f(\xi_o^K, \zeta^*) - f(\xi^*, \zeta_o^K).$$

We begin with the random graph setup. In the time-static scenario, the graph remains fixed throughout the algorithm's execution. In contrast, the time-varying scenario involves randomly removing and adding edges at each iteration. Starting graph is shown in Figure 2.

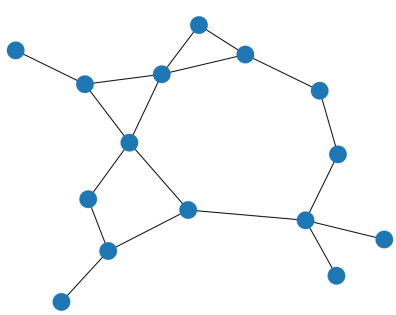

Figure 2: Example of a random graph with 15 nodes.

We test our method on a simple non-smooth saddle-point problem of the form:

$$\min_{\xi\in\mathbb{R}^{d_\xi}} \max_{\zeta\in\mathbb{R}^{d_\zeta}} p(\xi,\zeta) = \sum_{i=1}^{n} f_i(\xi,\zeta) + \frac{r}{2}\|\xi\| - \frac{r}{2}\|\zeta\|,$$

where $r = 10^{-3}$ and

$$f_i(\xi,\zeta) = \|\xi - c_{\xi,i}\|_1 - \|\zeta - c_{\zeta,i}\|_1,$$

with $c_{\xi,i}$ and $c_{\zeta,i}$ some fixed constants.

These functions are convex in ξ and concave in ζ. The subgradients with respect to ξ and ζ are computed separately.

Subgradient with respect to ξ:

$$\partial_\xi f_i(\xi,\zeta) = \partial\|\xi - c_{\xi,i}\|_1 = \left\{ v \in \mathbb{R}^{d_\xi} \;\middle|\; v_j \in \begin{cases} \{\mathrm{sign}(\xi_j - c_{\xi,i,j})\}, & \xi_j \neq c_{\xi,i,j}, \\ [-1,1], & \xi_j = c_{\xi,i,j} \end{cases} \right\}.$$

Subgradient with respect to ζ:

$$\partial_\zeta f_i(\xi,\zeta) = -\partial\|\zeta - c_{\zeta,i}\|_1 = \left\{ -v \in \mathbb{R}^{d_\zeta} \;\middle|\; v_j \in \begin{cases} \{\mathrm{sign}(\zeta_j - c_{\zeta,i,j})\}, & \zeta_j \neq c_{\zeta,i,j}, \\ [-1,1], & \zeta_j = c_{\zeta,i,j} \end{cases} \right\}.$$

Then, we can write operator $\mathbf{T}_i$ from the Definition 1 of $f_i(\xi,\zeta)$ as a pair:

$$\mathbf{T}_i(\xi,\zeta) = \left(\partial_\xi f_i(\xi,\zeta),\ -\partial_\zeta f_i(\xi,\zeta)\right).$$

We evaluate the performance for $K \in \{1,\dots,30\}$, with a fixed parameter $T = 10$.

Figure 3 presents the performance of the method under the time-varying graph setting. As observed, the randomness in the graph structure introduces some fluctuations in both performance metrics. Conversely, Figure 4 illustrates the results under the time-static setting, where the graph remains fixed. In this case, we observe a more stable and monotonic decrease in error across iterations.

7 Conclusion

In this paper, we investigate non-smooth stochastic decentralized optimization over time-varying networks, addressing both convex minimization and convex-concave saddle point problems. Our research extends previous work that focused primarily on smooth settings or time-static networks; thus we provide a more general result. We study both monotone and strongly monotone scenarios. We consider both monotone (weakly convex/concave) and strongly monotone problems, as well as the asymmetric case where the strong convexity and strong concavity parameters differ.

We generalize the deterministic algorithm from (Kovalev et al. 2024) to handle arbitrary stochastic monotone operators, making it applicable to a broader class of real-world problems. Our main theorems establish optimal convergence rates (matching theoretical lower bounds) for both convex minimization and saddle point problems. Moreover, these rates are optimal for the deterministic case and the non-distributed case as well.

A possible future direction may be studying the case of asymmetric oracle cost. In saddle point problems, the computational cost of querying the convex (primal) and concave (dual) oracles may differ (e.g., in GANs). Extending our analysis could result in a better convergence rate with this assumption.

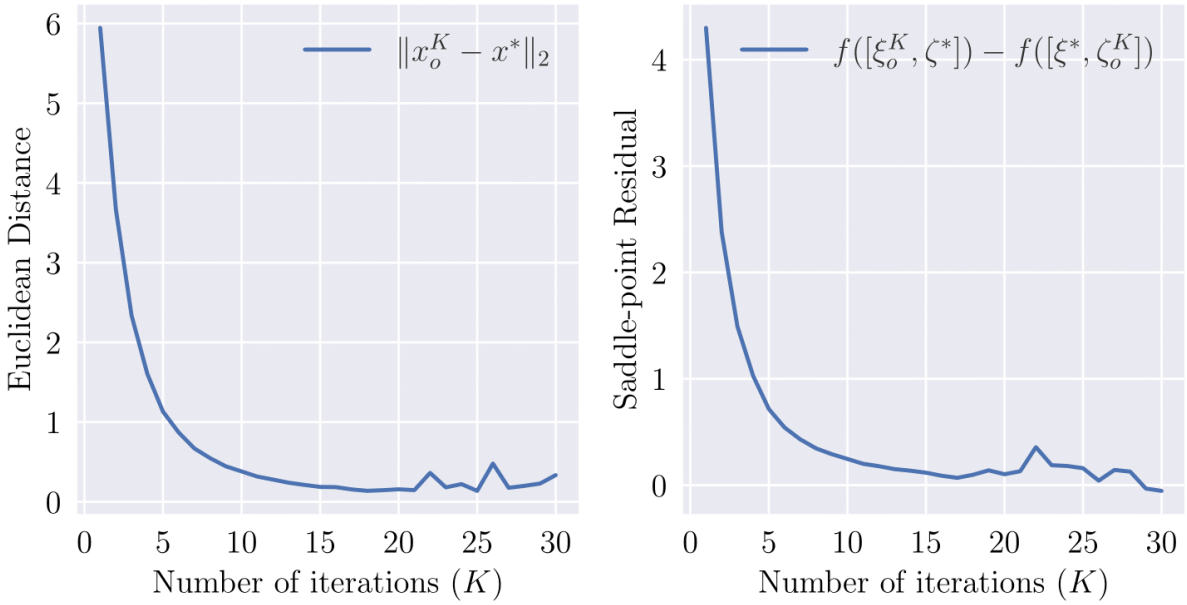

Figure 3: Performance over time-varying graphs.

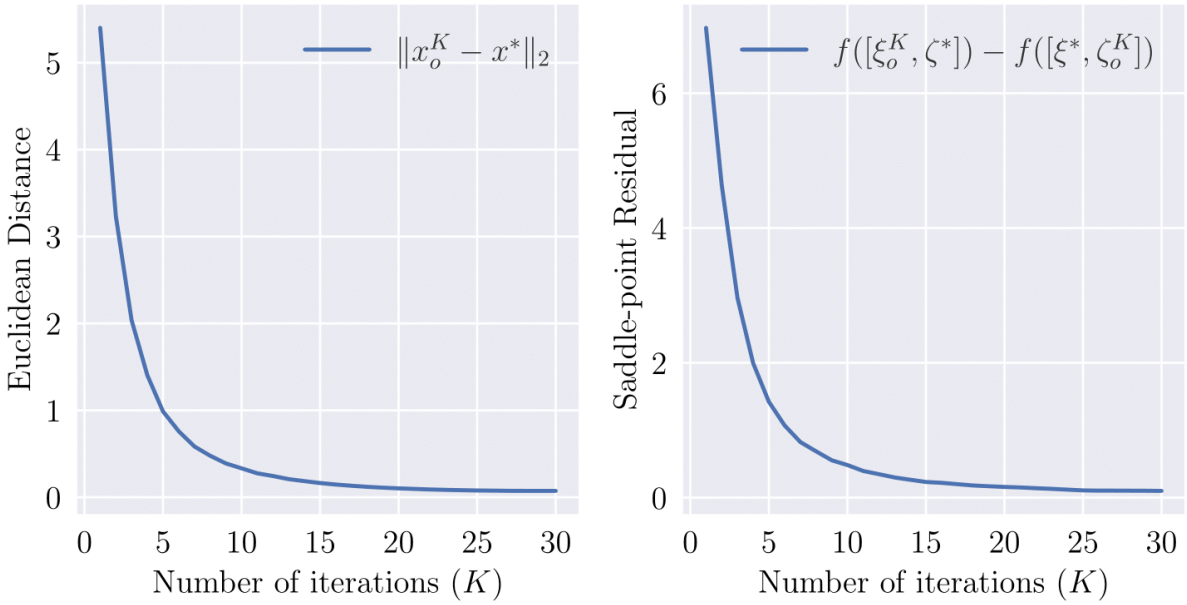

Figure 4: Performance over time-static graphs.

Acknowledgements

The study was supported by the Ministry of Economic Development of the Russian Federation (agreement No. 139-10-2025-034 dd. 19.06.2025, IGK 000000C313925P4D0002)

References

Beck, A.; Nedić, A.; Ozdaglar, A.; and Teboulle, M. 2014. An $O(1/k)$ Gradient Method for Network Resource Allocation Problems. *IEEE Transactions on Control of Network Systems*, 1(1): 64–73.

Bubeck, S. 2015. Convex Optimization: Algorithms and Complexity. arXiv:1405.4980.

Dvinskikh, D.; and Gasnikov, A. 2021. Decentralized and Parallel Primal and Dual Accelerated Methods for Stochastic Convex Programming Problems. arXiv:1904.09015.

Facchinei, F.; and Pang, J. 2007. *Finite-Dimensional Variational Inequalities and Complementarity Problems*. Springer Series in Operations Research and Financial Engineering. Springer New York. ISBN 9780387218151.

Forero, P. A.; Cano, A.; and Giannakis, G. B. 2010. Consensus-Based Distributed Support Vector Machines. *Journal of Machine Learning Research*, 11(55): 1663–1707.

Galakatos, A.; Crotty, A.; and Kraska, T. 2018. *Distributed Machine Learning*, 1196–1201. New York, NY: Springer New York. ISBN 978-1-4614-8265-9.

Gan, L.; Topcu, U.; and Low, S. 2013. Optimal decentralized protocol for electric vehicle charging. *IEEE Transactions on Power Systems*, 28: 940–951.

Gidel, G.; Berard, H.; Vignoud, G.; Vincent, P.; and Lacoste-Julien, S. 2020. A Variational Inequality Perspective on Generative Adversarial Networks. arXiv:1802.10551.

Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative Adversarial Networks. arXiv:1406.2661.

Gorbunov, E.; Rogozin, A.; Beznosikov, A.; Dvinskikh, D.; and Gasnikov, A. 2022. *Recent Theoretical Advances in Decentralized Distributed Convex Optimization*, 253–325. Springer International Publishing. ISBN 9783031008320.

Jambulapati, A.; Sidford, A.; and Tian, K. 2019. A Direct $\tilde{O}(1/\epsilon)$ Iteration Parallel Algorithm for Optimal Transport. arXiv:1906.00618.

Jin, Y.; and Sidford, A. 2020. Efficiently Solving MDPs with Stochastic Mirror Descent. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 4890–4900. PMLR.

Kovalev, D.; Borodich, E.; Gasnikov, A.; and Feoktistov, D. 2024. Lower Bounds and Optimal Algorithms for Non-Smooth Convex Decentralized Optimization over Time-Varying Networks. *arXiv preprint arXiv:2405.18031*.

Kovalev, D.; Gasanov, E.; Gasnikov, A.; and Richtarik, P. 2021. Lower Bounds and Optimal Algorithms for Smooth and Strongly Convex Decentralized Optimization Over Time-Varying Networks. In Ranzato, M.; Beygelzimer, A.; Dauphin, Y.; Liang, P.; and Vaughan, J. W., eds., *Advances in Neural Information Processing Systems*, volume 34, 22325–22335. Curran Associates, Inc.

Kovalev, D.; Salim, A.; and Richtarik, P. 2020. Optimal and Practical Algorithms for Smooth and Strongly Convex Decentralized Optimization. In Larochelle, H.; Ranzato, M.; Hadsell, R.; Balcan, M.; and Lin, H., eds., *Advances in Neural Information Processing Systems*, volume 33, 18342–18352. Curran Associates, Inc.

Metelev, D.; Rogozin, A.; Gasnikov, A.; and Kovalev, D. 2022. Decentralized Saddle-Point Problems with Different Constants of Strong Convexity and Strong Concavity. arXiv:2206.00090.

Nedić, A.; and Ozdaglar, A. 2009. volume 9780521762229, 340–386. Cambridge University Press. ISBN 9780521762229. Publisher Copyright: © Cambridge University Press 2010.

Omidshafiei, S.; Pazis, J.; Amato, C.; How, J. P.; and Vian, J. 2017. Deep Decentralized Multi-task Multi-Agent Reinforcement Learning under Partial Observability. arXiv:1703.06182.

Rabbat, M.; and Nowak, R. 2004. Distributed optimization in sensor networks. IPSN '04, 20–27. New York, NY, USA: Association for Computing Machinery. ISBN 1581138466.

Ram, S.; Veeravalli, V.; and Nedić, A. 2009. Distributed non-autonomous power control through distributed convex optimization. In *IEEE INFOCOM 2009 - The 28th Conference on Computer Communications*, Proceedings - IEEE INFOCOM, 3001–3005. ISBN 9781424435135. 28th Conference on Computer Communications, IEEE INFOCOM 2009 ; Conference date: 19-04-2009 Through 25-04-2009.

Rogozin, A.; Beznosikov, A.; Dvinskikh, D.; Kovalev, D.; Dvurechensky, P.; and Gasnikov, A. 2024. Decentralized saddle point problems via non-Euclidean mirror prox. *Optimization Methods and Software*, 1–26.

Scaman, K.; Bach, F.; Bubeck, S.; Lee, Y. T.; and Massoulié, L. 2017. Optimal Algorithms for Smooth and Strongly Convex Distributed Optimization in Networks. In Precup, D.; and Teh, Y. W., eds., *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, 3027–3036. PMLR.

Scaman, K.; Bach, F.; Bubeck, S.; Massoulié, L.; and Lee, Y. T. 2018. Optimal Algorithms for Non-Smooth Distributed Optimization in Networks. In Bengio, S.; Wallach, H.; Larochelle, H.; Grauman, K.; Cesa-Bianchi, N.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Wai, H.-T.; Yang, Z.; Wang, Z.; and Hong, M. 2019. Multi-Agent Reinforcement Learning via Double Averaging Primal-Dual Optimization. arXiv:1806.00877.

Wang, J.; and Hu, X. 2010. Distributed Consensus in Multi-vehicle Cooperative Control: Theory and Applications (Ren, W. and Beard, R.W.; 2008) [Book Shelf]. *IEEE Control Systems Magazine - IEEE CONTROL SYST MAG*, 30: 85–86.