北京大学
PEKING UNIVERSITY

D-SOBA: A Single-Loop Decentralized Bilevel Algorithm with Transient Complexity Analysis

Kun Yuan

Center for Machine Learning Research @ Peking University

EUSIPCO 2025

Joint work with

Boao Kong
(Peking U.)

Shuchen Zhu
(Peking U.)

Songtao Lu
(IBM Research)

Xinmeng Huang
(Upenn)

北京大学 PEKING UNIVERSITY

D-SOBA: A Single-Loop Decentralized Bilevel Algorithm with Transient Complexity Analysis

Boao Kong*, Shuchen Zhu*, Songtao Lu†, Xinmeng Huang‡, Kun Yuan§

*Center for Data Science, Peking University, Beijing, China
†Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong SAR of China
‡Graduate Group in Applied Mathematics and Computational Science, University of Pennsylvania, Philadelphia, USA
§Center for Machine Learning Research, Peking University, Beijing, China
{kongboao, shuchenzhu}@stu.pku.edu.cn, stlu@cse.cuhk.edu.hk, xinmengh@sas.upenn.edu, kunyuan@pku.edu.cn

***Abstract*—Stochastic bilevel optimization (SBO) is becoming increasingly essential in machine learning due to its versatility in handling nested structures. To address large-scale SBO, decentralized approaches have emerged as effective paradigms in which nodes communicate with immediate neighbors without a central server. However, current decentralized SBO algorithms face challenges, including expensive inner-loop updates and unclear understanding of the influence of network topology, data heterogeneity, and the nested bilevel algorithmic structures. In this paper, we introduce a single-loop decentralized SBO (D-SOBA) algorithm and establish its transient iteration complexity, which, for the first time, clarifies the joint influence of network topology and data heterogeneity on decentralized bilevel algorithms.**

***Index Terms*—bilevel optimization, decentralized optimization, transient iteration, non-asymptotic convergence analysis.**

I. Introduction

Decentralized stochastic bilevel optimization, which tackles problems with nested optimization structures across multiple computing nodes, has gained growing interest. This paper considers N nodes connected through a given graph topology. Each node i privately owns an upper-level loss function $f_i : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}$ and a lower-level loss function $g_i : \mathbb{R}^d \times \mathbb{R}^p \to \mathbb{R}$. All nodes collaboratively aim to find a solution to the following optimization problem:

$$\min_{x\in\mathbb{R}^d} \quad \Phi(x) := f(x, y^\star(x)) := \frac{1}{N}\sum_{i=1}^N f_i(x, y^\star(x)) \tag{1a}$$

$$\text{s.t.} \quad y^\star(x) := \arg\min_{y\in\mathbb{R}^p} \left\{ g(x,y) := \frac{1}{N}\sum_{i=1}^N g_i(x,y) \right\} \tag{1b}$$

where f_i and g_i are defined as:

$$\begin{aligned} f_i(x,y) &:= \mathbb{E}_{\xi_i\sim\mathcal{D}_{f_i}}[F(x,y;\xi_i)], \\ g_i(x,y) &:= \mathbb{E}_{\zeta_i\sim\mathcal{D}_{g_i}}[G(x,y;\zeta_i)]. \end{aligned} \tag{2}$$

The random variables ξ_i and ζ_i represent data samples available at node i, following local distributions $\mathcal{D}_{f_i}$ and $\mathcal{D}_{g_i}$, respectively. Throughout this paper, we assume local data distributions vary across different nodes, which may result in data heterogeneity issues during the training process.

*Equal contribution. §Corresponding author.

Limitations in existing literature. Existing works, such as [1]–[5], have developed decentralized bilevel algorithms that offer both theoretical guarantees and empirical effectiveness. However, two key limitations remain in the current literature:

- **Expensive inner-loop updates.** Existing algorithms rely on computationally costly inner-loop updates to estimate the lower-level solution $y^\star(x)$ and the Hessian inverse of $g_i(x,y)$. These updates not only increase computational complexity but also incur significant communication overhead, limiting the practicality of the algorithms.
- **Inadequate Non-Asymptotic Analysis.** While existing studies [3]–[5] show that decentralized and centralized bilevel algorithms achieve the same asymptotic convergence rate, they fail to clarify the non-asymptotic stage, where decentralization-induced slowdowns are observed due to limited iterations in practical scenarios. Current research either overlooks the influence of network topologies or neglects the impact of data heterogeneity, leaving critical questions about when and how decentralized bilevel algorithms slow down unanswered.

Contributions. To address these limitations, we propose a **D**ecentralized **S**tochastic **O**ne-loop **B**ilevel **A**lgorithm (**D-SOBA**). Our contributions are threefold. First, we establish that D-SOBA achieves linear speedup with an asymptotic gradient complexity of $\mathcal{O}(1/(N\varepsilon^2))$, surpassing the current results by at least a factor of $\log(1/\varepsilon)$. Second, we provide a *non-asymptotic*[1] convergence analysis and derive the transient iteration complexity for the D-SOBA framework, quantifying how *network topology* and *data heterogeneity* jointly influence the non-asymptotic convergence stage. Third, we prove that our algorithm achieves the same asymptotic convergence rate and transient complexity as single-level algorithms, implying that the nested structure does not introduce fundamental challenges to decentralized bilevel optimization.

All our established results as well as those of existing decentralized SBO algorithms are listed in Table I. D-SOBA

[1]Following recent conventions in decentralized optimization literature, we use 'non-asymptotic convergence' to characterize the iteration complexity required to achieve ε-approximate stationarity and use 'asymptotic convergence' to characterize the iteration complexity as $T \to +\infty$.

EUSIPCO 2025

Decentralized Bilevel Optimization: A Perspective from Transient Iteration Complexity

Boao Kong* KONGBOAO@STU.PKU.EDU.CN
Center for Data Science, Peking University
Beijing, China

Shuchen Zhu* SHUCHENZHU@STU.PKU.EDU.CN
Center for Data Science, Peking University
Beijing, China

Songtao Lu STLU@CSE.CUHK.EDU.HK
Department of Computer Science and Engineering
The Chinese University of Hong Kong
Hong Kong SAR of China

Xinmeng Huang XINMENGH@SAS.UPENN.EDU
Graduate Group in Applied Mathematics and Computational Science
University of Pennsylvania
Philadelphia, USA

Kun Yuan† KUNYUAN@PKU.EDU.CN
Center for Machine Learning Research, Peking University
AI for Science Institute
Beijing, China

Abstract

Stochastic bilevel optimization (SBO) is becoming increasingly essential in machine learning due to its versatility in handling nested structures. To address large-scale SBO, decentralized approaches have emerged as effective paradigms in which nodes communicate with immediate neighbors without a central server, thereby improving communication efficiency and enhancing algorithmic robustness. However, most decentralized SBO algorithms focus solely on asymptotic convergence rates, overlooking transient iteration complexity—the number of iterations required before asymptotic rates dominate, which results in limited understanding of the influence of network topology, data heterogeneity, and the nested bilevel algorithmic structures. To address this issue, this paper introduces **D-SOBA**, a **D**ecentralized **S**tochastic **O**ne-loop **B**ilevel **A**lgorithm framework. D-SOBA comprises two variants: D-SOBA-SO, which incorporates second-order Hessian and Jacobian matrices, and D-SOBA-FO, which relies entirely on first-order gradients. We provide a comprehensive non-asymptotic convergence analysis and establish the transient iteration complexity of D-SOBA. This provides the first theoretical understanding of how network topology, data heterogeneity, and nested bilevel structures influence decentralized SBO. Extensive experimental results demonstrate the efficiency and theoretical advantages of D-SOBA.

Keywords: bilevel optimization, decentralized optimization, transient iteration complexity, non-asymptotic convergence analysis.

*. Equal contribution.
†. Corresponding author.

Long Report

北京大学
PEKING UNIVERSITY

PART 01

Bilevel Optimization

Bilevel Optimization

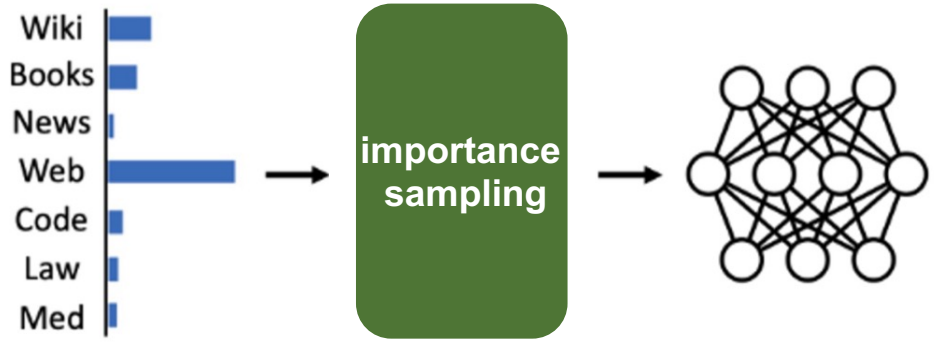
$$\min_{x \in \mathbb{R}^d} \quad \Phi(x) = f(x, y^*(x)) \qquad \textbf{(Upper Level)}$$

$$\text{s.t.} \quad y^*(x) = \arg\min_{y \in \mathbb{R}^p} g(x, y) \qquad \textbf{(Lower Level)}$$

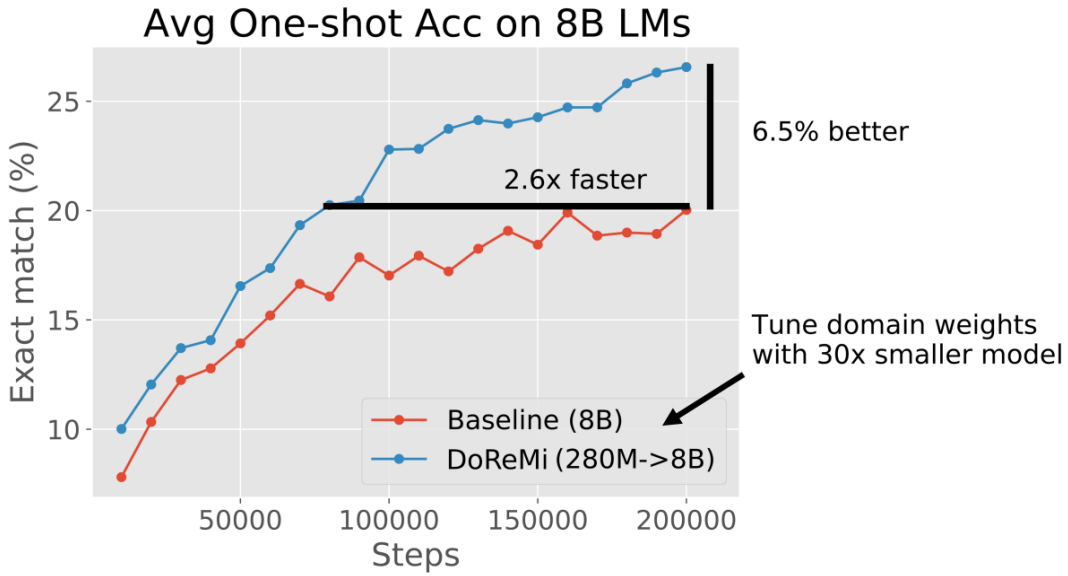
- We assume both $f(\cdot)$ and $g(\cdot)$ are smooth functions
- We further assume $f(\cdot)$ is non-convex w.r.t. x and $g(\cdot)$ is strongly-convex w.r.t. y
- If $g = -f$, the above bilevel optimization problem reduces to a minimax problem:

$$\min_x \max_y f(x, y)$$

Application: Domain reweighting

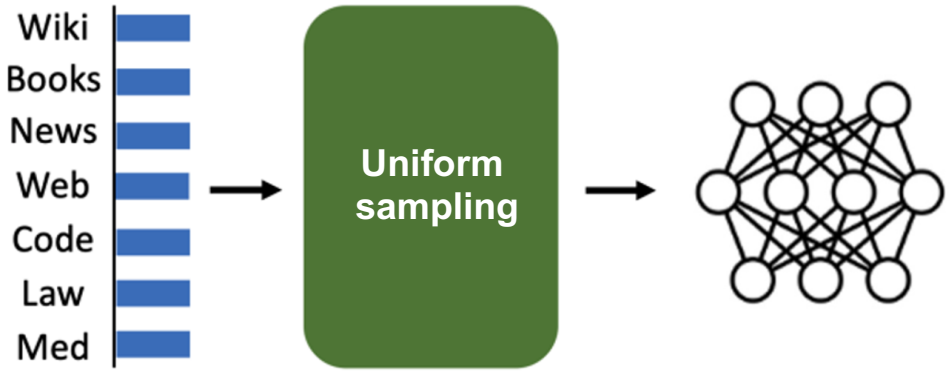


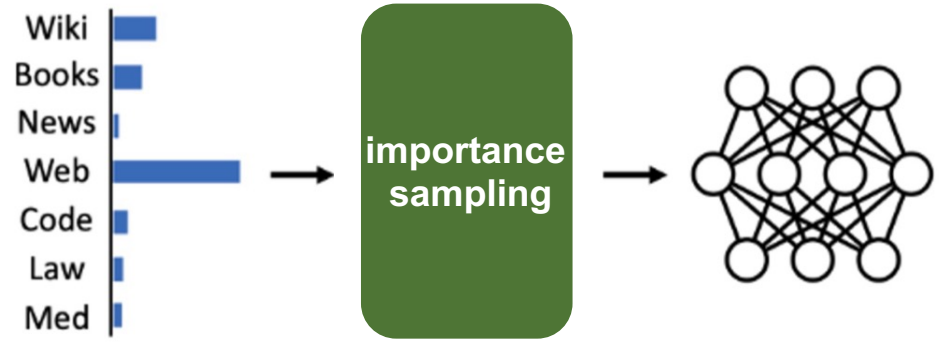
- Data for LLM comes from various sources (Wiki, books, news, code, etc.)
- The mixture proportions of pretraining data domains greatly affect LLM performance



[Xie, et.al., *DoReMi: Optimizing Data Mixtures Speeds Up Language Model Pretraining,* NeurIPS 2023]

Application: Domain reweighting





- How to decide data mixture proportions?

Bilevel Optimization!

$$\min_{p \in \Lambda} \quad \underline{L_{\text{val}}}(w^\star(p))$$

Loss on validation dataset

$$\text{s.t.} \quad w^\star(p) = \arg\min_{w} \sum_{i=1}^{m} \frac{p_i}{n_i} \sum_{j=1}^{n_i} \underline{L_{\text{trn}}}(w; \xi_j^i)$$

Loss on training dataset

[Pan, et.al., *ScaleBiO: Scalable Bilevel Optimization for LLM Data Reweighting,* 2024]

Other applications

北京大学 PEKING UNIVERSITY

- Meta-learning and hyperparameter optimization

[L. Bertinetto, et. al., *Meta-learning with differentiable closed-form solvers,* ICLR, 2019.]

[L. Franceschi, et. al., *Bilevel programming for hyperparameter optimization and meta-learning,* ICML, 2018.]

[J. Snell, et. al., *Prototypical networks for few-shot learning*, NeurIPS, 2017.]

- Reinforcement learning, adversarial learning, continue learning, imitation learning

[M. Hong, et. al., *A two-timescale stochastic algorithm framework for bilevel optimization*, SIAM J. Optim., 2023.]

[Y. Zhang, et. al., *Revisiting and advancing fast adversarial training through the lens of bilevel optimization*, ICML, 2022.]

[Z. Borsos, et. al., *Coresets via bilevel optimization for continual learning and streaming*, NeurIPS, 2020.]

[S. Arora, et. al., *Provable representation learning for imitation learning via bilevel optimization*, ICML, 2020.]

北京大学
PEKING UNIVERSITY

PART 02

Decentralized Stochastic Bilevel Optimization

Distributed Stochastic Bilevel Optimization

北京大学 PEKING UNIVERSITY

- Bilevel optimization may involve **massive** datasets, especially in LLM tasks

data reweighting

$$\min_{p\in\Lambda} \quad L_{\text{val}}(w^\star(p))$$

$$\text{s.t.} \quad w^\star(p) = \arg\min_{w} \sum_{i=1}^{m} \frac{p_i}{n_i} \sum_{j=1}^{n_i} L_{\text{trn}}(w; \xi_j^i)$$

Dataset	Sampling prop.	Epochs	Disk size
CommonCrawl	67.0%	1.10	3.3 TB
C4	15.0%	1.06	783 GB
Github	4.5%	0.64	328 GB
Wikipedia	4.5%	2.45	83 GB
Books	4.5%	2.23	85 GB
ArXiv	2.5%	1.06	92 GB
StackExchange	2.0%	1.03	78 GB

Table 1: **Pre-training data.** Data mixtures used for pre-training, for each subset we list the sampling proportion, number of epochs performed on the subset when training on 1.4T tokens, and disk size. The pre-training runs on 1T tokens have the same sampling proportion.

[LLaMA: Open and Efficient Foundation Language Models]

- Efficient distributed stochastic bilevel optimization (SBO) algorithms are in urgent need

Distributed Stochastic Bilevel Optimization

- We consider the following distributed stochastic bilevel optimization problem over N nodes

$$\min_{x\in\mathbb{R}^d} \quad \Phi(x) := f(x, y^*(x)) := \frac{1}{N}\sum_{i=1}^{N} f_i(x, y^*(x)) \qquad \textbf{(Upper Level)}$$

$$\text{s.t.} \quad y^*(x) := \arg\min_{y\in\mathbb{R}^p} \left\{ g(x,y) := \frac{1}{N}\sum_{i=1}^{N} g_i(x,y) \right\} \qquad \textbf{(Lower Level)}$$

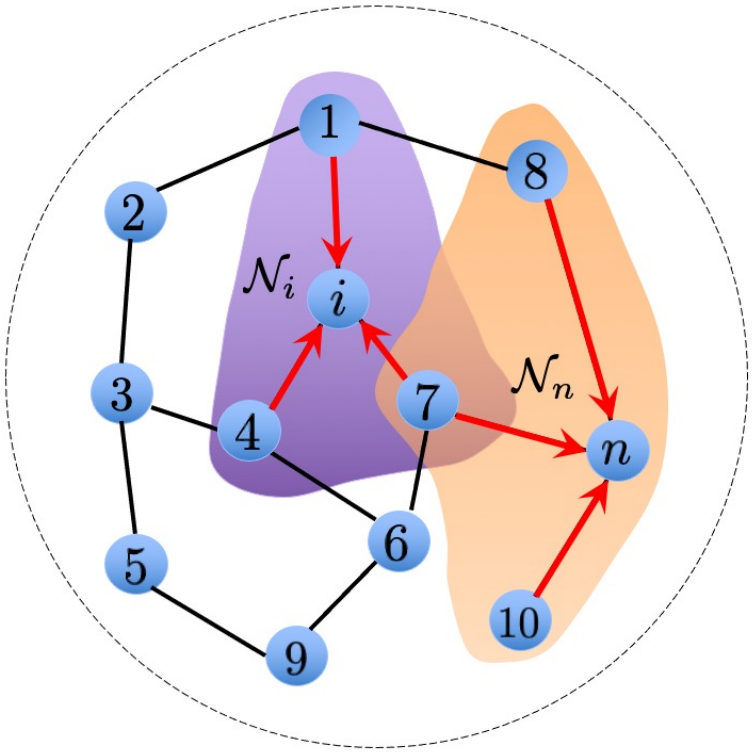
  - Each $f_i(x,y) = \mathbb{E}_{\phi\sim\mathcal{D}_{f_i}}[F_i(x,y;\phi)]$ is a local upper level stochastic cost function
  - Each $g_i(x,y) = \mathbb{E}_{\xi\sim\mathcal{D}_{g_i}}[G_i(x,y;\xi)]$ is a local lower level stochastic cost function
  - Random variables ϕ and ξ denotes the local data

Decentralized Stochastic Bilevel Optimization

- Decentralized bilevel algorithms rely on **partial** averaging over a graph

$$x_i^{(t+\frac{1}{2})} = x_i^{(t)} - \gamma h_i^{(t)} \qquad \text{(Local compt.)}$$

$$x_i^{(t+1)} = \sum_{j \in \mathcal{N}_i} w_{ij} x_j^{(t+\frac{1}{2})} \qquad \text{(Partial comm.)}$$



- $h_i^{(t)}$ is an **estimation** of the gradient of upper-level loss $\nabla\Phi(x_i^{(t)})$.
- $\mathcal{N}_i$ is the set of neighbors at node i
- w_{ij} scales information from node j to node i; satisfying $\sum_{j \in \mathcal{N}_i} w_{ij} = 1$

Mixing Matrix and Spectral Gap

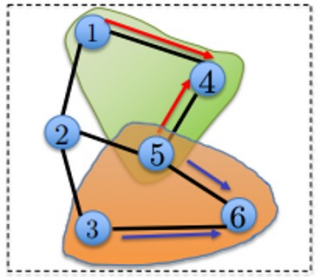
- We stack all weights into a **mixing matrix** $W = (w_{ij})_{N\times N} \in \mathbb{R}^{N\times N}$ and assume that W is **doubly-stochastic**, i.e.,

$$\mathbb{1}_N^\top W = \mathbb{1}_N^\top, \quad W\mathbb{1}_N = \mathbb{1}_N.$$

- **Network topology determines how fast** that partial average will converge to global average

- We introduce quantity ρ to gauge the graph connectivity

$$\rho = \|W - \frac{1}{n}\mathbb{1}\mathbb{1}^T\|_2 \in (0,1) \text{ where } W = [w_{ij}] \in \mathbb{R}^{n\times n}$$



$$W = \begin{pmatrix} 5/12 & 1/4 & 0 & 1/3 & 0 & 0 \\ 1/4 & 1/4 & 1/4 & 0 & 1/4 & 0 \\ 0 & 1/4 & 5/12 & 0 & 0 & 1/3 \\ 1/3 & 0 & 0 & 5/12 & 1/4 & 0 \\ 0 & 1/4 & 0 & 1/4 & 1/4 & 1/4 \\ 0 & 0 & 1/3 & 0 & 1/4 & 5/12 \end{pmatrix}$$

$$\rho = \left\|W - \mathbb{1}\mathbb{1}^\top/6\right\|_2 = 0.75$$

  - Well-connected topology has $\rho \to 0$, e.g. fully-connected topology
  - Sparsely-connected topology has $\rho \to 1$, e.g. ring has $\rho = O(1 - \frac{1}{n^2})$

Challenges to Develop Decentralized Bilevel Method

$$\min_{x\in\mathbb{R}^d} \quad \Phi(x) := f(x, y^*(x)) = \frac{1}{N}\sum_{i=1}^{N} f_i(x, y^*(x)) \qquad \text{s.t.} \quad y^*(x) := \arg\min_{y\in\mathbb{R}^p} \left\{ g(x,y) := \frac{1}{N}\sum_{i=1}^{N} g_i(x,y) \right\}$$

- The hyper-gradient $\nabla\Phi(x)$ can be computed by:

$$\nabla\Phi(x) = \nabla_1 f(x, y^*(x)) - \nabla^2_{12} g(x, y^*(x)) \left[\nabla^2_{22} g(x, y^*(x))\right]^{-1} \nabla_2 f(x, y^*(x))$$

- **Main challenge**: the Hessian inversion cannot be accessed easily

$$\left[\nabla^2_{22} g(x, y^\star(x))\right]^{-1} = \left[\frac{1}{N}\sum_{i=1}^{N} \nabla^2_{22}\, g_i(x, y^\star(x))\right]^{-1}$$

- Needs to **achieve $y^*(x)$** for each x
- **Matrix communication** is very expensive
- Cannot be easily accessed through decentralized partial averaging due to **the inversion operator**

PART 03

D-SOBA: A single-loop DSBO algorithm

SOBA: Introduce an auxiliary variable

- **Stochastic One-loop Bilevel Algorithm (SOBA)** addresses Hessian-inverse estimation [1].

- In SBO hyper-gradient, we have:

- **To remove Hessian inverse**, SOBA introduces an auxiliary variable z:

$$z^*(x) = \left[\nabla_{22}^2 g(x, y^*(x))\right]^{-1} \nabla_2 f(x, y^*(x))$$

which is the solution to the following problem:

$$\min_z \left[h(x, y^\star(x), z) = \frac{1}{2} z^\top \nabla_{22}^2 g(x, y^*(x)) z - z^\top \nabla_2 f(x, y^*(x)) \right]$$

[1] M. Dagréou, et. al., *A framework for bilevel optimization that enables stochastic and global variance reduction algorithms*. NeurIPS 2022.

SOBA: The algorithm framework

$$h(x, y^\star(x), z) = \frac{1}{2} z^\top \nabla^2_{22} g(x, y^*(x)) z - z^\top \nabla_2 f(x, y^*(x))$$

$$\min_{x \in \mathbb{R}^d} \quad \Phi(x) := f(x, y^*(x)) = \frac{1}{N} \sum_{i=1}^{N} f_i(x, y^*(x)) \qquad \text{s.t.} \quad y^*(x) := \arg\min_{y \in \mathbb{R}^p} \left\{ g(x, y) := \frac{1}{N} \sum_{i=1}^{N} g_i(x, y) \right\}$$

Exact version

$$y^\star(x) = \arg\min_y g(x, y)$$

$$z^*(x) = \arg\min_z \; h(x, y^*(x), z)$$

$$\nabla \Phi(x) = \nabla_1 f(x, y^*(x)) - \nabla^2_{12} g(x, y^*(x)) z^\star(x)$$

$$x^+ = x - \alpha \nabla \Phi(x)$$

No Hessian inversion

Gradient descent version

$$y^+ = y - \beta \nabla_2 g(x, y) \qquad (K\times)$$

$$z^+ = z - \gamma \nabla_3 h(x, y^+, z) \qquad (K\times)$$

$$\nabla \Phi(x) = \nabla_1 f(x, y^+) - \nabla^2_{12} g(x, y^+) z^+$$

$$x^+ = x - \alpha \nabla \Phi(x) \qquad (1\times)$$

No Hessian inversion

SOBA: Single-Loop!

$$y^+ = y - \beta \nabla_2 g(x, y) \qquad (1\times)$$

$$z^+ = z - \gamma \nabla_3 h(x, y^+, z) \qquad (1\times)$$

$$\nabla \Phi(x) = \nabla_1 f(x, y^+) - \nabla^2_{12} g(x, y^+) z^+$$

$$x^+ = x - \alpha \nabla \Phi(x) \qquad (1\times)$$

No Hessian inversion

M. Dagréou, et. al., *A framework for bilevel optimization that enables stochastic and global variance reduction algorithms*. NeurIPS 2022.

Centralized SOBA

$$\min_z \left[h(x, y^\star(x), z) = \frac{1}{2} z^\top \nabla_{22}^2 g(x, y^*(x)) z - z^\top \nabla_2 f(x, y^*(x)) \right]$$

北京大学 PEKING UNIVERSITY

- Recall that both the upper and lower level cost function are

$$f(x, y^*(x)) = \frac{1}{N} \sum_{i=1}^{N} f_i(x, y^*(x)) \qquad g(x, y) := \frac{1}{N} \sum_{i=1}^{N} g_i(x, y)$$

- Furthermore, the auxiliary cost function can be written into a finite-sum problem

$$h(x, y^*(x), z) = \frac{1}{N} \sum_{i=1}^{N} h_i(x, y^*(x), z)$$

$$\text{where} \quad h_i(x, y^*(x), z) := \frac{1}{2} z^\top \nabla_{22}^2 g_i(x, y^*(x)) z - z^\top \nabla_2 f_i(x, y^*(x))$$

B. Kong et. al., *Decentralized Bilevel Optimization over Graphs: Loopless Algorithmic Update and Transient Iteration Complexity*. arXiv 2024.

Centralized SOBA

SOBA: Single-Loop!

$$y^+ = y - \beta \nabla_2 g(x, y) \qquad (1\times)$$

$$z^+ = z - \gamma \nabla_3 h(x, y^+, z) \qquad (1\times)$$

$$\nabla \Phi(x) = \nabla_1 f(x, y^+) - \nabla_{12}^2 g(x, y^+) z^+$$

$$x^+ = x - \alpha \nabla \Phi(x) \qquad (1\times)$$

No Hessian inversion

Centralized SOBA

$$x^{(t+1)} = x^{(t)} - \frac{\alpha_t}{N} \sum_{i=1}^{N} D_{x,i}^{(t)}, \qquad (1\times)$$

$$y^{(t+1)} = y^{(t)} - \frac{\beta_t}{N} \sum_{i=1}^{N} D_{y,i}^{(t)}, \qquad (1\times)$$

$$z^{(t+1)} = z^{(t)} - \frac{\gamma_t}{N} \sum_{i=1}^{N} D_{z,i}^{(t)}. \qquad (1\times)$$

$$D_{x,i}^{(t)}(x, y, z) = \nabla_1 f_i(x^{(t)}, y^{(t)}) - \nabla_{12}^2 g_i(x^{(t)}, y^{(t)}) z^{(t)},$$

$$D_{y,i}^{(t)}(x, y, z) = \nabla_2 g_i(x^{(t)}, y^{(t)}), \quad D_{y,i}^{(t)}(x, y, z) = \nabla_{22}^2 g_i(x^{(t)}, y^{(t)}) z^{(t)} - \nabla_2 f_i(x^{(t)}, y^{(t)}).$$

Decentralized SOBA

北京大学 PEKING UNIVERSITY

Centralized SOBA

$$x^{(t+1)} = x^{(t)} - \frac{\alpha_t}{N}\sum_{i=1}^{N} D_{x,i}^{(t)}, \quad (1\times)$$

$$y^{(t+1)} = y^{(t)} - \frac{\beta_t}{N}\sum_{i=1}^{N} D_{y,i}^{(t)}, \quad (1\times)$$

$$z^{(t+1)} = z^{(t)} - \frac{\gamma_t}{N}\sum_{i=1}^{N} D_{z,i}^{(t)}. \quad (1\times)$$

Decentralized SOBA (D-SOBA)

$$x_i^{(t+1)} = \sum_{j\in\mathcal{N}_i} w_{ij}\left(x_j^{(t)} - \alpha_t D_{x,i}^{(t)}\right), \quad (1\times)$$

$$y_i^{(t+1)} = \sum_{j\in\mathcal{N}_i} w_{ij}\left(y_j^{(t)} - \alpha_t D_{y,i}^{(t)}\right), \quad (1\times)$$

$$z_i^{(t+1)} = \sum_{j\in\mathcal{N}_i} w_{ij}\left(z_j^{(t)} - \alpha_t D_{z,i}^{(t)}\right). \quad (1\times)$$

$$D_{x,i}^{(t)}(x,y,z) = \nabla_1 f_i(x^{(t)},y^{(t)}) - \nabla_{12}^2 g_i(x^{(t)},y^{(t)})z^{(t)},$$

$$D_{y,i}^{(t)}(x,y,z) = \nabla_2 g_i(x^{(t)},y^{(t)}), \quad D_{y,i}^{(t)}(x,y,z) = \nabla_{22}^2 g_i(x^{(t)},y^{(t)})z^{(t)} - \nabla_2 f_i(x^{(t)},y^{(t)}).$$

B. Kong et. al., *Decentralized Bilevel Optimization over Graphs: Loopless Algorithmic Update and Transient Iteration Complexity*. arXiv 2024.

D-SOBA (the stochastic version)

- By **introducing the decentralized** communication to centralized SOBA, we present the following **D-SOBA algorithm**:

Algorithm 1: D-SOBA

Initialize $x^{(0)} = y^{(0)} = z^{(0)} = h^{(0)} = 0$, the values of $\alpha_t, \beta_t, \gamma_t$ are set properly

for $t = 0, 1, \cdots, T-1$ **do**

  for *each node* $i = 1, 2, \cdots, N$ *in parallel* **do**

$$
\begin{aligned}
x_i^{(t+1)} &:= \textstyle\sum_{j\in\mathcal{N}_i} w_{ij}(x_j^{(t)} - \alpha_t h_j^{(t)});\\
y_i^{(t+1)} &:= \textstyle\sum_{j\in\mathcal{N}_i} w_{ij}(y_j^{(t)} - \beta_t v_j^{(t)});\\
\hat{z}_i^{(t+1)} &:= z_i^{(t)} - \gamma_t(H_i^{(t)} z_i^{(t)} - u_{i,y}^{(t)});\\
z_i^{(t+1)} &:= \textstyle\sum_{j\in\mathcal{N}_i} w_{ij}\hat{z}_j^{(t+1)};\\
\omega_i^{(t+1)} &:= u_{i,x}^{(t)} - J_i^{(t)} z_i^{(t)};\\
h_i^{(t+1)} &:= (1-\theta_t)h_i^{(t)} + \theta_t \omega_i^{(t+1)}.
\end{aligned}
$$

$$
\begin{aligned}
&\xi_i^{(t)} \sim \mathcal{D}_{f_i},\ \zeta_i^{(t)} \sim \mathcal{D}_{g_i},\\
&u_{i,x}^{(t)} := \nabla_1 F(x_i^{(t)}, y_i^{(t)}; \xi_i^{(t)}),\\
&u_{i,y}^{(t)} := \nabla_2 F(x_i^{(t)}, y_i^{(t)}; \xi_i^{(t)}),\\
&v_i^{(t)} := \nabla_2 G(x_i^{(t)}, y_i^{(t)}; \zeta_i^{(t)}),\\
&J_i^{(t)} := \nabla_{12}^2 G(x_i^{(t)}, y_i^{(t)}; \zeta_i^{(t)}),\\
&H_i^{(t)} := \nabla_{22}^2 G(x_i^{(t)}, y_i^{(t)}; \zeta_i^{(t)}).
\end{aligned}
$$

D-SOBA Convergence: Assumptions

北京大学 PEKING UNIVERSITY

Assumption 1 (Smoothness)

Each function $\nabla f_i, \nabla g_i, \nabla^2 g_i$ are Lipschitz continuous, $g_i(x,\cdot)$ is μ_g-strongly convex for any $x \in \mathbb{R}^d$, and $\left\|\nabla_2 f_i(x, y^*(x))\right\| \leq L_f < \infty$ for all $x \in \mathbb{R}^d$.

Assumption 2 (Bounded Gradient Dissimilarity)

There exists $b > 0$ such that for any $(x,y) \in \mathbb{R}^d \times \mathbb{R}^p$:

$$\frac{1}{N}\sum_{i=1}^{N}\|\nabla_1 f_i(x,y) - \nabla_1 f(x,y)\|^2 \leq b^2, \quad \frac{1}{N}\sum_{i=1}^{N}\|\nabla_2 g_i(x,y) - \nabla_2 g(x,y)\|^2 \leq b^2, \quad \frac{1}{N}\sum_{i=1}^{N}\|\nabla_{22} g_i(x,y) - \nabla_{22} g(x,y)\|^2 \leq b^2.$$

Assumption 3 (Stochasticity)

There exists $\sigma > 0$ such that for any $(x,y) \in \mathbb{R}^d \times \mathbb{R}^p$:

$$\mathbb{E}_{\xi_i \sim \mathcal{D}_{f_i}}[\nabla_1 F(x,y;\xi_i)] = \nabla_1 f_i(x,y) \quad \mathbb{E}_{\xi_i \sim \mathcal{D}_{f_i}}\|\nabla_1 F(x,y;\xi_i) - \nabla_1 f_i(x,y)\|^2 \leq \sigma^2 \quad \mathbb{E}_{\zeta_i \sim \mathcal{D}_{g_i}}[\nabla^2_{12} G(x,y;\zeta_i)] = \nabla^2_{12} g_i(x,y) \quad \mathbb{E}_{\zeta_i \sim \mathcal{D}_{g_i}}\|\nabla^2_{12} G(x,y;\zeta_i) - \nabla^2_{12} g_i(x,y)\|^2 \leq \sigma^2$$

$$\mathbb{E}_{\zeta_i \sim \mathcal{D}_{g_i}}[\nabla_1 G(x,y;\xi_i)] = \nabla_2 g_i(x,y) \quad \mathbb{E}_{\zeta_i \sim \mathcal{D}_{g_i}}\|\nabla_2 G(x,y;\zeta_i) - \nabla_2 g_i(x,y)\|^2 \leq \sigma^2 \quad \mathbb{E}_{\zeta_i \sim \mathcal{D}_{g_i}}[\nabla^2_{22} G(x,y;\zeta_i)] = \nabla^2_{22} g_i(x,y) \quad \mathbb{E}_{\zeta_i \sim \mathcal{D}_{g_i}}\|\nabla^2_{22} G(x,y;\zeta_i) - \nabla^2_{22} g_i(x,y)\|^2 \leq \sigma^2$$

D-SOBA Convergence

北京大学 PEKING UNIVERSITY

Theorem 1

Suppose Assumption 1, 2, and 3 hold, and the mixing matrix W is doubly-stochastic. Then there exist $c_1, c_2, c_3 > 0, \alpha_t \equiv \Theta\left((N/T)^{1/2}\right)$, and $\beta_t \equiv c_1\alpha_t, \gamma_t \equiv c_2\alpha_t, \theta_t \equiv c_3\alpha_t$, such that the iteration of $\bar{x}^{(t)}$ in D-SOBA satisfies:

Clarify the influence of data heterogeneity: large hetero. hurts the rate

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\left\|\nabla\Phi(\bar{x}^{(t)})\right\|^2\right] \lesssim \frac{1}{\sqrt{NT}} + \frac{\rho^{\frac{2}{3}}}{(1-\rho)^{\frac{1}{3}}T^{\frac{2}{3}}} + \frac{\rho^{\frac{2}{3}}b^{\frac{2}{3}}}{(1-\rho)^{\frac{2}{3}}T^{\frac{2}{3}}} + \frac{\rho}{(1-\rho)T} + \frac{1}{T}.$$

Clarify the influence of network topology: $\rho \to 1$ hurts the rate

Decentralized single-level stochastic gradient descent:

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\left\|\nabla\Phi(\bar{x}^{(t)})\right\|^2\right] \lesssim \frac{1}{\sqrt{NT}} + \frac{\rho^{\frac{2}{3}}}{(1-\rho)^{\frac{1}{3}}T^{\frac{2}{3}}} + \frac{\rho^{\frac{2}{3}}b^{\frac{2}{3}}}{(1-\rho)^{\frac{2}{3}}T^{\frac{2}{3}}} + \frac{\rho}{(1-\rho)T} + \frac{1}{T}.$$

Bilevel structure does not deteriorate the order of the convergence rate!

The first result to show that decentralized bilevel method can match with single-level method

Comparison with existing algorithms

北京大学 PEKING UNIVERSITY

State-of-the-art complexity

Algorithm	Single loop	Asymptotic rate$^{\diamond}$	Asymptotic grad. complexity†	Transient complexity‡	Assumption$^{\triangleleft}$
DSBO[11]	✗	$\frac{1}{\sqrt{T}}$	$\frac{1}{\epsilon^3}$	N. A.	LC f_i
MA-DSBO[12]	✗	$\frac{1}{\sqrt{T}}$	$\frac{1}{\epsilon^2}\log(\frac{1}{\epsilon})$	N. A.	LC f_i
SLAM[40]	✗	$\frac{1}{\sqrt{NT}}$	$\frac{1}{N\epsilon^2}\log(\frac{1}{\epsilon})$	N. A.	LC f_i
Gossip DSBO[62]	✗	$\frac{1}{\sqrt{NT}}$	$\frac{1}{N\epsilon^2}\log(\frac{1}{\epsilon})$	$\frac{N^3G^4}{(1-\rho)^4}$ $^{\triangleright}$	BG ∇f_i
MDBO[22, Thm 1]*	✗	$\frac{1}{(1-\rho)\sqrt{T}}$	$\frac{1}{(1-\rho)^2\epsilon^2}\log(\frac{1}{\epsilon})$	N. A.	BG ∇f_i
MDBO[22, Thm 2]*	✗	$\frac{1}{\sqrt{NT}}$	$\frac{1}{N\epsilon^2}\log(\frac{1}{\epsilon})$	$\frac{N^3}{(1-\rho)^8}$	BG $\nabla f_i, \nabla g_i$
D-SOBA (ours)	✔	$\boldsymbol{\frac{1}{\sqrt{NT}}}$	$\boldsymbol{\frac{1}{N\epsilon^2}}$	$\boldsymbol{\max\left\{\frac{\rho^4N^3}{(1-\rho)^2}, \frac{\rho^4N^3b^2}{(1-\rho)^4}\right\}}$	**BGD** $\nabla f_i, \nabla g_i$
Single-level DSGD [14]	✔	$\frac{1}{\sqrt{NT}}$	$\frac{1}{N\epsilon^2}$	$\max\left\{\frac{\rho^4N^3}{(1-\rho)^2}, \frac{\rho^4N^3b^2}{(1-\rho)^4}\right\}$	BGD ∇f_i

D-SOBA Experiment 1: Hyper-parameter tuning in logistic regression

北京大学 PEKING UNIVERSITY

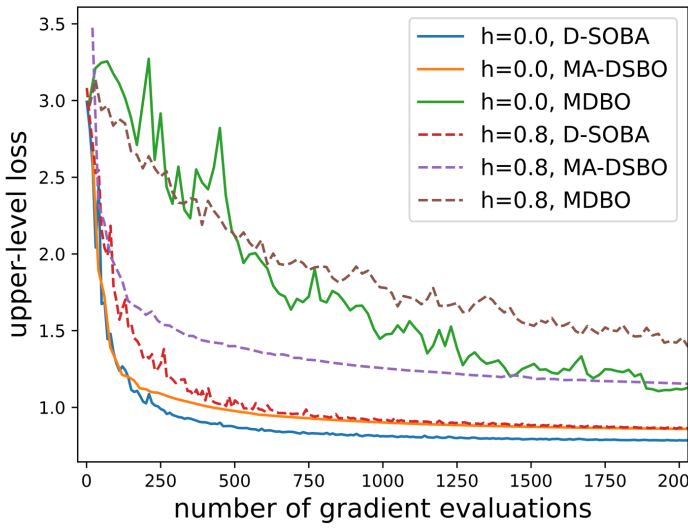
- Loss function on the i-th node:

$$f_i(x,y) = \frac{1}{|\mathcal{D}_{val}^{(i)}|} \sum_{(\xi,\zeta)\in\mathcal{D}_{val}^{(i)}} L(\xi^\top y, \zeta),$$

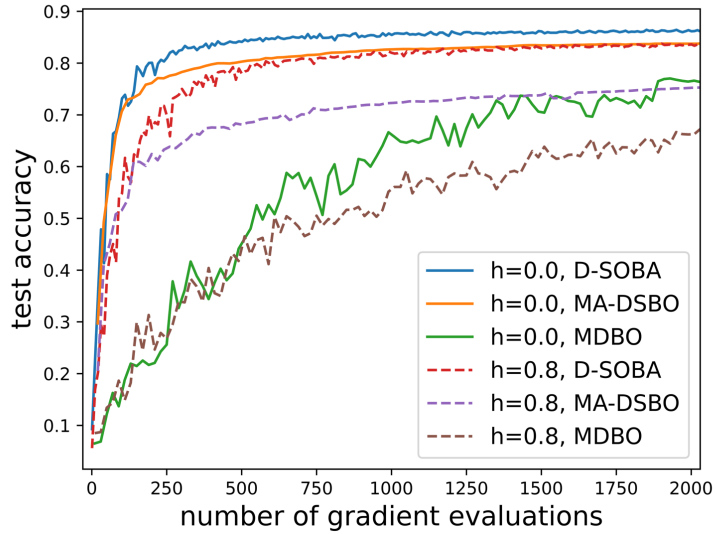
$$g_i(x,y) = \frac{1}{|\mathcal{D}_{tr}^{(i)}|} \sum_{(\xi,\zeta)\in\mathcal{D}_{tr}^{(i)}} L(\xi^\top y, \zeta) + \frac{1}{cp}\sum_{i=1}^{c}\sum_{j=1}^{p} e^{x_j} y_{ij}^2$$

- N=20, two different heterogeneity.
- Compare with MA-DSBO and MDBO.

upper-level loss



test accuracy



D-SOBA achieves a **faster convergence** and **higher test-accuracy** than the other two algorithms.

D-SOBA Experiment 2: Data hyper-cleaning for FashionMNIST dataset

北京大学 PEKING UNIVERSITY

- Loss function on the i-th node:

$$f_i(x,y) = \frac{1}{\left|\mathcal{D}_{val}^{(i)}\right|} \sum_{(\xi_e,\zeta_e)\in D_{val}^{(i)}} L(\phi(\xi_e;y),\zeta_e),$$

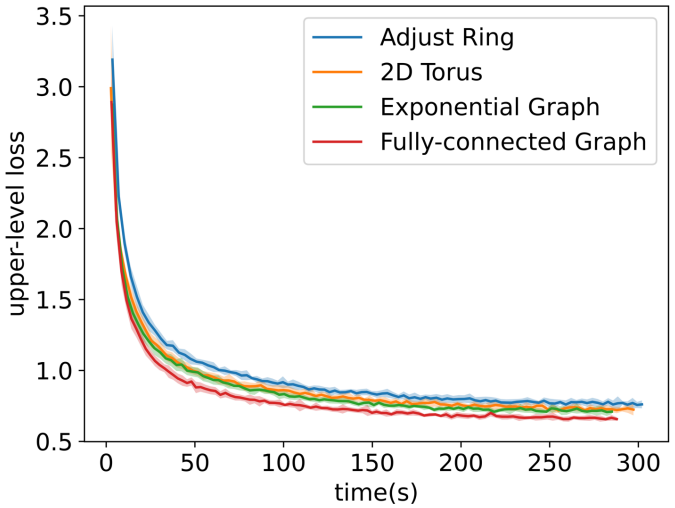
$$g_i(x,y) = \frac{1}{\left|\mathcal{D}_{tr}^{(i)}\right|} \sum_{(\xi_e,\zeta_e)\in \mathcal{D}_{tr}^{(i)}} \sigma(x_e) L(\phi(\xi_e;y),\zeta_e) + C\|y\|^2$$

  - ϕ: MLP with a 300-dim hidden layer and ReLU activation.

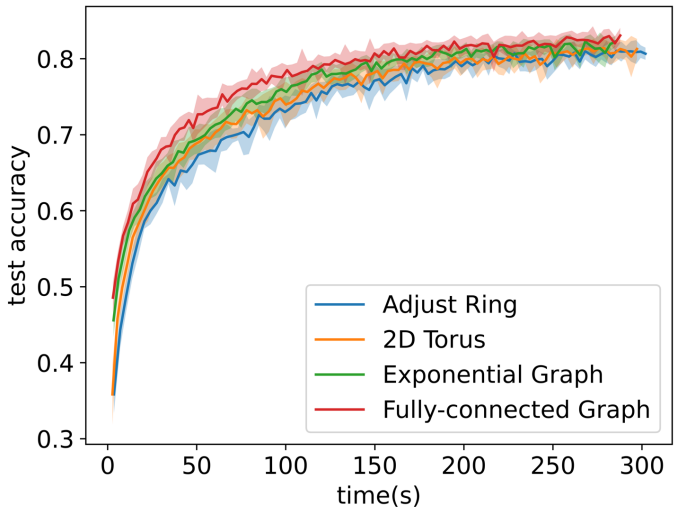
- N=10, non-i.i.d FashionMNIST dataset.

$$\rho_{\text{fully-connected}} < \rho_{\text{exp}} < \rho_{\text{torus}} < \rho_{\text{adjusted-ring}}$$

upper-level loss



test accuracy



Sparser topology leads to worse convergence performance.

Summary

We present D-SOBA: A single-loop algorithm for decentralized SBO problem.

- D-SOBA achieves **loopless** Hessian-inverse estimation.
- We present the non-asymptotic convergence analysis for D-SOBA
- D-SOBA has the same transient complexity with **single-level D-SGD**.

References:

[1] Boao Kong, Shuchen Zhu, Songtao Lu, Xinmeng Huang, Kun Yuan, *D-SOBA: A Single-Loop Decentralized Bilevel Algorithm with Transient Complexity Analysis,* EUSIPCO 2025

[2] Boao Kong, Shuchen Zhu, Songtao Lu, Xinmeng Huang, Kun Yuan, *Decentralized Bilevel Optimization: A Perspective from Transient Iteration Complexity,* arXiv 2402.03167v3, 2024.

北京大学
PEKING UNIVERSITY

The end of the story? Not Yet !

D-SOBA suffers from several drawbacks

北京大学 PEKING UNIVERSITY

- **Stringent assumption**

Assumption 2 (Bounded Gradient Dissimilarity)

There exists $b > 0$ such that for any $(x, y) \in \mathbb{R}^d \times \mathbb{R}^p$:

$$\frac{1}{N}\sum_{i=1}^{N}\|\nabla_1 f_i(x,y) - \nabla_1 f(x,y)\|^2 \le b^2, \quad \frac{1}{N}\sum_{i=1}^{N}\|\nabla_2 g_i(x,y) - \nabla_2 g(x,y)\|^2 \le b^2, \quad \frac{1}{N}\sum_{i=1}^{N}\|\nabla_{22} g_i(x,y) - \nabla_{22} g(x,y)\|^2 \le b^2.$$

It does not hold even for simple quadratic functions $f_i(x,y) = x^\top A_i x + y^\top A_i y$

- Gradient dissimilarity/data heterogeneity **amplifies the influence of network topology**

$$\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\left[\left\|\nabla\Phi(\bar{x}^{(t)})\right\|^2\right] \lesssim \frac{1}{\sqrt{NT}} + \frac{\rho^{\frac{2}{3}}}{(1-\rho)^{\frac{1}{3}}T^{\frac{2}{3}}} + \boxed{\frac{\rho^{\frac{2}{3}}b^{\frac{2}{3}}}{(1-\rho)^{\frac{2}{3}}T^{\frac{2}{3}}}} + \frac{\rho}{(1-\rho)T} + \frac{1}{T}.$$

Amplifies the influence of ρ

Open questions

北京大学 PEKING UNIVERSITY

- Can we remove the influence of data heterogeneity?
- Can heterogeneity-correction techniques, originally designed for single-level optimization, be effectively applied to bilevel optimization problems? Which technique yields the best results?
- Can we use different decentralized algorithms across different optimization levels? What combination of the algorithms lead to best performance?
- Can we use different network topologies across different optimization levels? How to choose the network topology for each level?

Our new results

北京大学 PEKING UNIVERSITY

All these questions have been answered in our new paper:

Shuchen Zhu, Boao Kong, Songtao Lu, Xinmeng Huang, Kun Yuan, ***SPARKLE: A Unified Single-Loop Primal-Dual Framework for Decentralized Bilevel Optimization***, arXiv 2411.14166, 2024.

Thank you!