北京大学
PEKING UNIVERSITY

Achieving Linear Speedup with Network-Independent Learning Rates in Decentralized Stochastic Optimization

Kun Yuan

Center for Machine Learning Research @ Peking University

Dec. 12, 2023

Joint work with

北京大学
PEKING UNIVERSITY

Hao Yuan

(Peking University)

Sulaiman A. Alghunaim

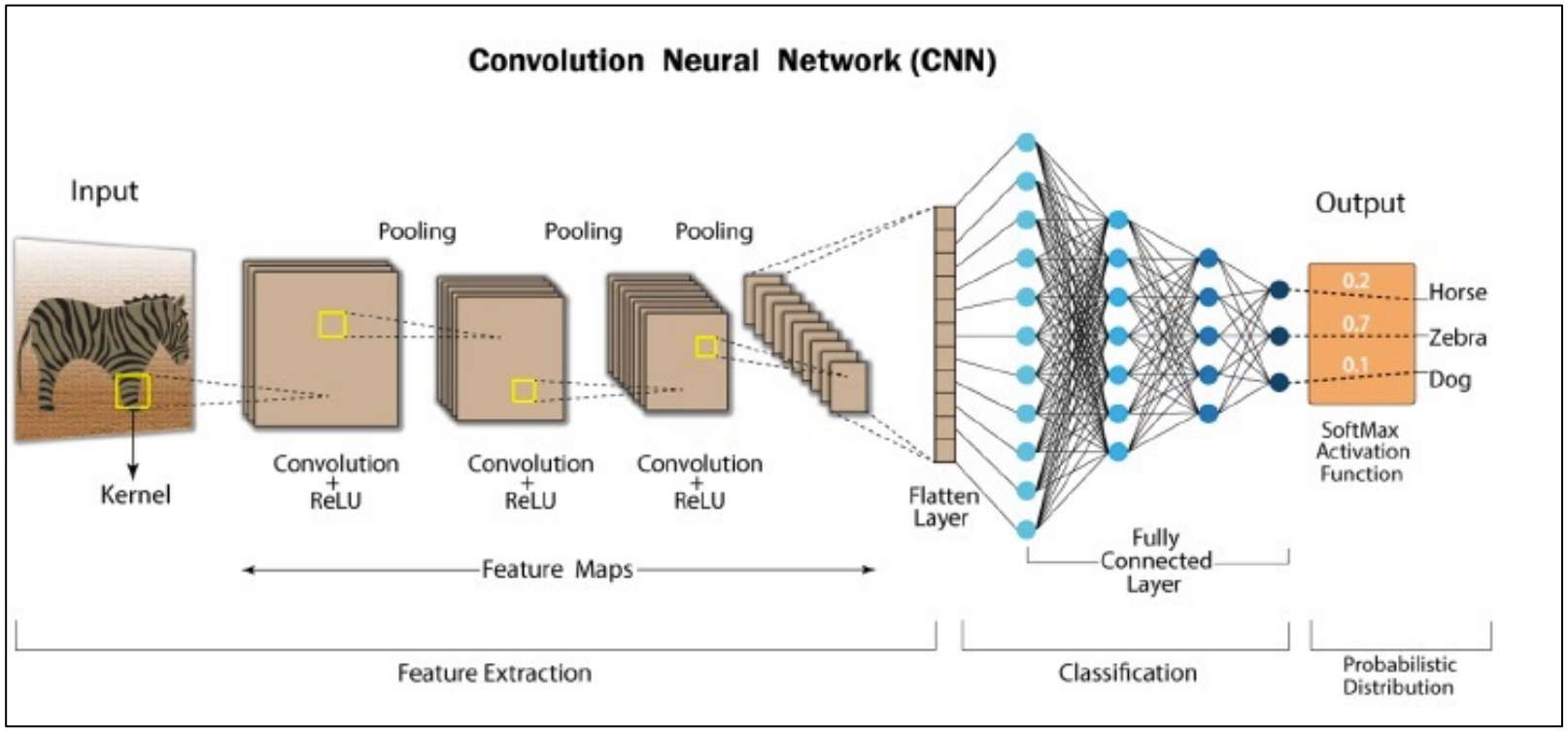
(Kuwait University)

北京大学
PEKING UNIVERSITY

Preface

Decentralized optimization in deep learning

Training deep neural network is notoriously difficult



DNN training = non-convexity + **massive dataset** + huge models

Distributed learning

北京大学 PEKING UNIVERSITY

- Training deep neural networks typically requires **massive** datasets; efficient and scalable distributed optimization algorithms are in urgent need

- A network of n nodes (devices such as GPUs) collaborate to solve the problem:

$$\min_{x\in\mathbb{R}^d} \quad f(x)=\frac{1}{n}\sum_{i=1}^{n} f_i(x), \quad \text{where} \quad f_i(x)=\mathbb{E}_{\xi_i\sim D_i} F(x;\xi_i)$$

  - Each component $f_i : \mathbb{R}^d \to \mathbb{R}$ is local and private to node i
  - Random variable ξ_i denotes the local data that follows distribution D_i
  - Each local distribution D_i is different; data heterogeneity exists

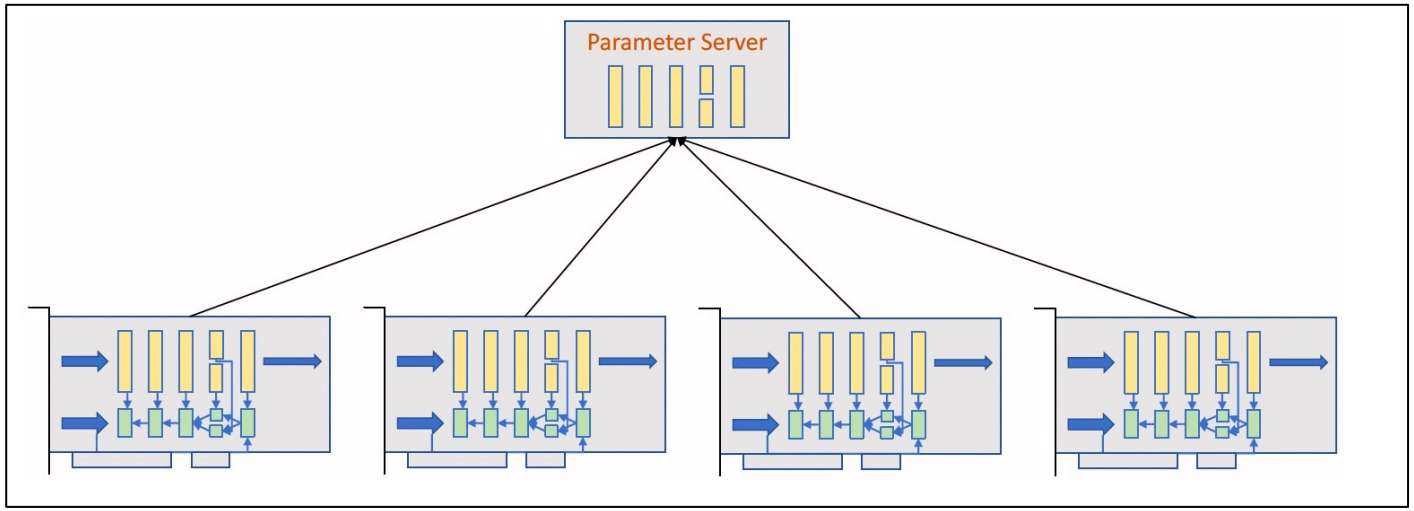
Vanilla parallel stochastic gradient descent (PSGD)

$$g_i^{(k)} = \nabla F(x^{(k)}; \xi_i^{(k)}) \qquad \text{(Local compt.)}$$

$$x^{(k+1)} = x^{(k)} - \frac{\gamma}{n} \sum_{i=1}^{n} g_i^{(k)} \qquad \text{(Global comm.)}$$

- Each node i samples data $\xi_i^{(k)}$ and computes gradient $\nabla F(x^{(k)}; \xi_i^{(k)})$
- All nodes synchronize (i.e. globally average) to update model x per iteration

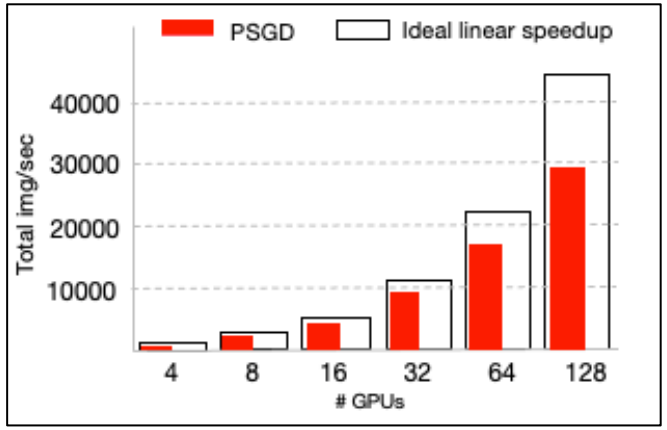
Vanilla parallel stochastic gradient descent (PSGD)



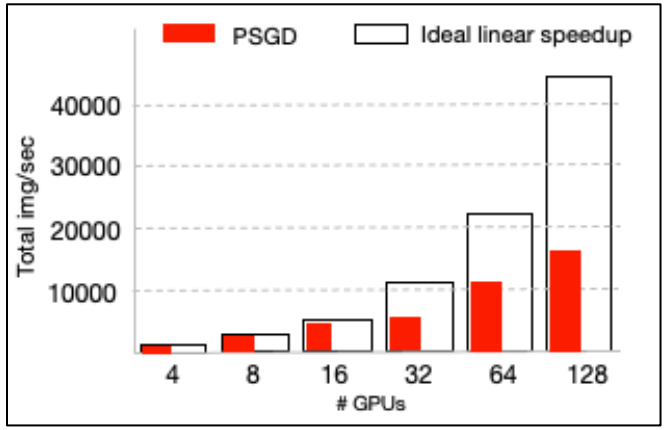
- Global average incurs $O(n)$ comm. overhead; **proportional to network size n**
- When network size n is large, PSGD suffers severe communication overhead

PSGD cannot achieve linear speedup due to comm. overhead

- PSGD cannot achieve ideal linear speedup in throughput due to comm. overhead
- Larger comm-to-compt ratio leads to worse performance in PSGD



Small comm.-to-compt. ratio



Large comm.-to-compt. ratio

- How can we accelerate PSGD? **Decentralized SGD is a promising paradime**

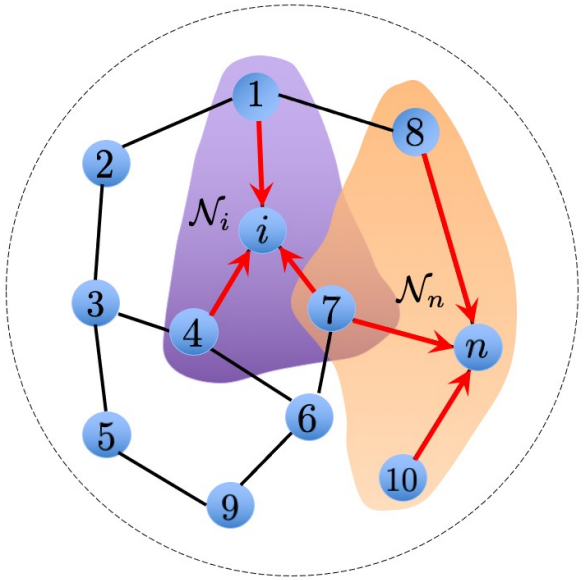
B. Ying, K. Yuan, H. Hu, Y. Chen and W. Yin, “BlueFog: Make decentralized algorithms practical for optimization and deep learning”, arXiv: 2111. 04287, 2021

Decentralized SGD (DSGD)

- To break $O(n)$ comm. overhead, we replace global average with partial average

$$x_i^{(k+\frac{1}{2})} = x_i^{(k)} - \gamma \nabla F(x_i^{(k)}; \xi_i^{(k)}) \quad \text{(Local update)}$$

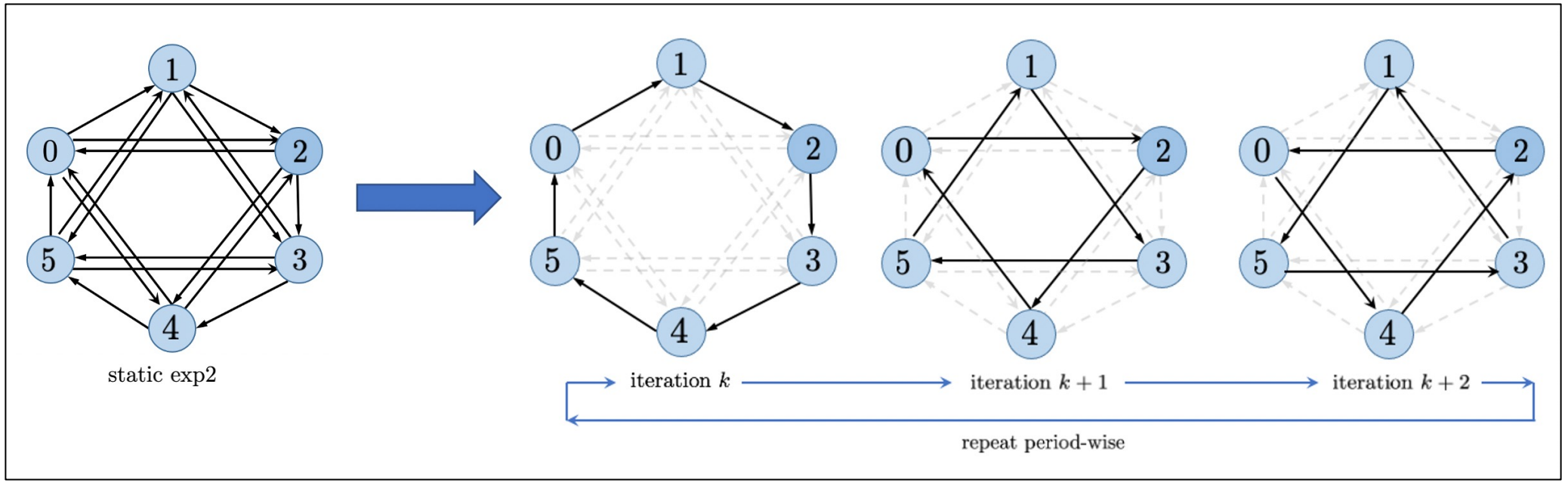
$$x_i^{(k+1)} = \sum_{j \in \mathcal{N}_i} w_{ij} x_j^{(k+\frac{1}{2})} \quad \text{(Partial averaging)}$$



- DSGD = local SGD update + partial averaging [LS08]
- $\mathcal{N}_i$ is the set of neighbors at node i ; w_{ij} scales information from j to i and satisfies $\sum_{j \in \mathcal{N}_i} w_{ij} = 1$
- Incurs $O(d_{\max})$ comm. overhead per iteration where $d_{\max} = \max_i \{|\mathcal{N}_i|\}$ is the graph maximum degree

DSGD is more communication-efficient than PSGD

- Incurs $O(1)$ comm. overhead on **sparse** topologies; much less than global average $O(n)$



One-peer exponential graph incurs O(1) comm. overhead

B. Ying, K. Yuan, Y. Chen, H. Hu, and W. Yin, "Exponential Graph is Provably Efficient for Decentralized Deep Training", NeurIPS 2021

DSGD is more communication-efficient than PSGD

北京大学 PEKING UNIVERSITY

- A real experiment on a 256-GPUs cluster [CYZ+21]

Model	Ring-Allreduce	Partial average
ResNet-50 (25.5M)	278 ms	150 ms

Table. Comparison of per-iter comm. time in terms of runtime with 256 GPUs

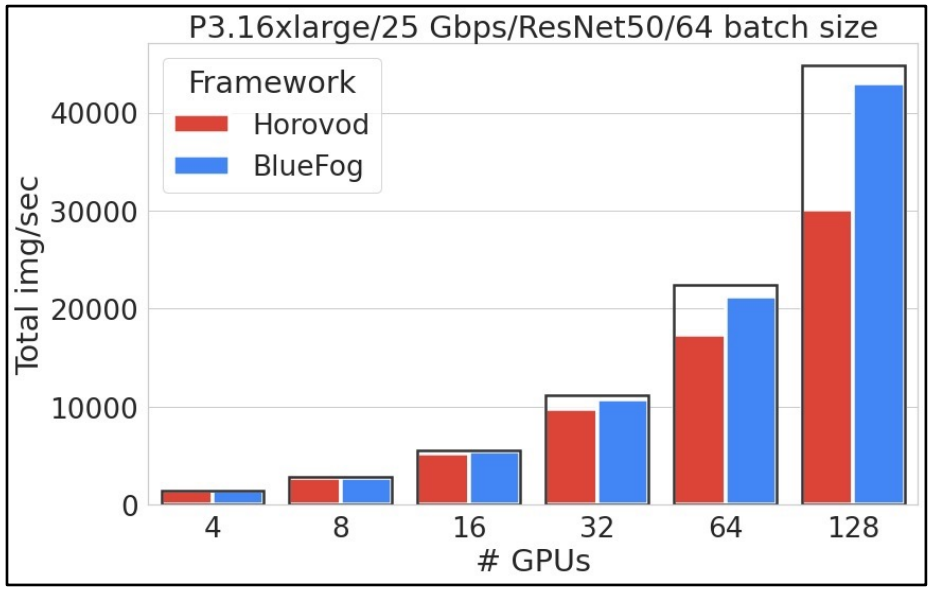
- DSGD saves more communications per iteration for larger models

[CYZ+21] Y. Chen*, K. Yuan*, Y. Zhang, P. Pan, Y. Xu, and W. Yin, ``Accelerating Gossip SGD with Periodic Global Averaging", ICML 2021

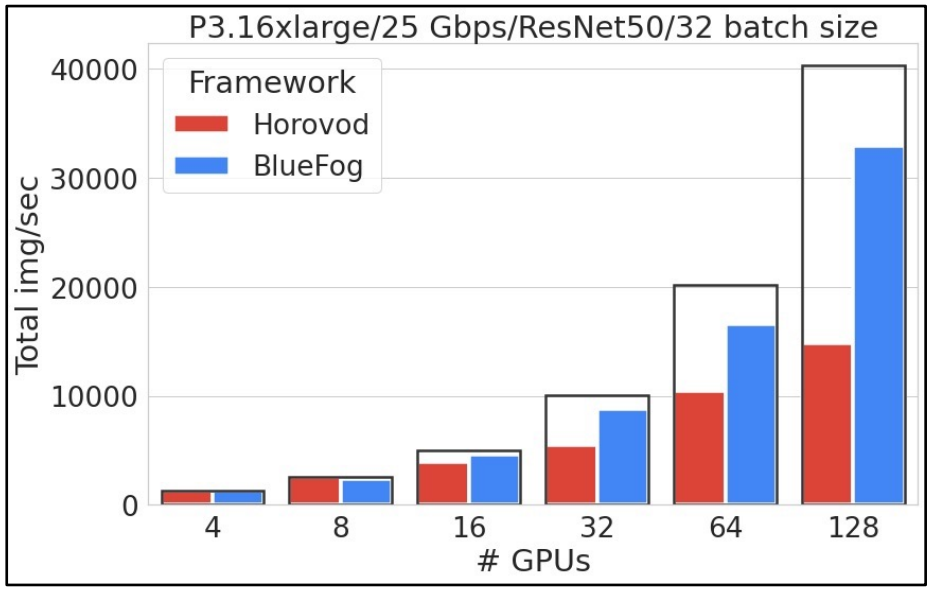
DSGD is more communication-efficient than PSGD

北京大学 PEKING UNIVERSITY

- DSGD (BlueFog) has **better linear speedup** than PSGD (Horovod) due to its small comm. overhead



Small comm.-to-compt. ratio



Large comm.-to-compt. ratio

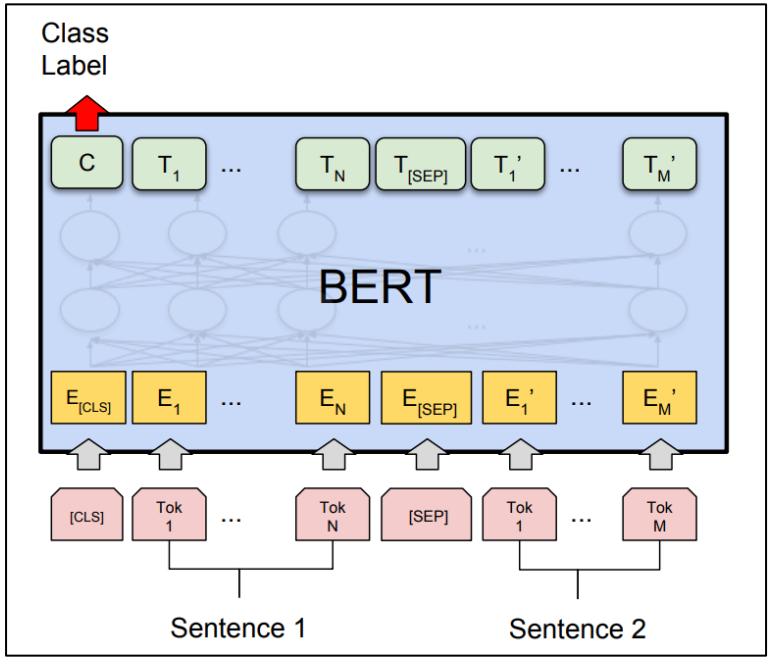
B. Ying, K. Yuan, H. Hu, Y. Chen and W. Yin, “BlueFog: Make decentralized algorithms practical for optimization and deep learning”, arXiv: 2111. 04287, 2021

DSGD achieves better linear speedup

nodes topology	4(4x8 GPUs) acc.	 time	8(8x8 GPUs) acc.	 time	16(16x8 GPUs) acc.	 time	32(32x8 GPUs) acc.	 time
P-SGD	76.32	11.6	76.47	6.3	76.46	3.7	76.25	2.2
Decentralized	76.34	11.1	76.52	5.7	76.47	2.8	76.27	1.5

DSGD shows very impressive linear speedup performance and saves more time than PSGD!

Experiments in deep learning (language modeling)



Model: BERT-Large (330M parameters)

Dataset: Wikipedia (2500M words) and BookCorpus (800M words)

Hardware: 64 GPUs

Table. Comparison in loss and training time [CYZ+21]

Method	Final Loss	Wall-clock Time (hrs)
P-SGD	1.75	59.02
D-SGD	1.77	30.4

[CYZ+21] Y. Chen*, K. Yuan*, Y. Zhang, P. Pan, Y. Xu, and W. Yin, ``Accelerating Gossip SGD with Periodic Global Averaging", ICML 2021

北京大学
PEKING UNIVERSITY

Part 01

Linear Speedup in Decentralized Optimization

Linear speedup

- Linear speedup is a fundamental benefit in decentralized stochastic optimization

- Recall the problem formulation

$$\min_{x\in\mathbb{R}^d} \quad f(x)=\frac{1}{n}\sum_{i=1}^{n} f_i(x), \quad \text{where} \quad f_i(x)=\mathbb{E}_{\xi_i\sim D_i}F(x;\xi_i).$$

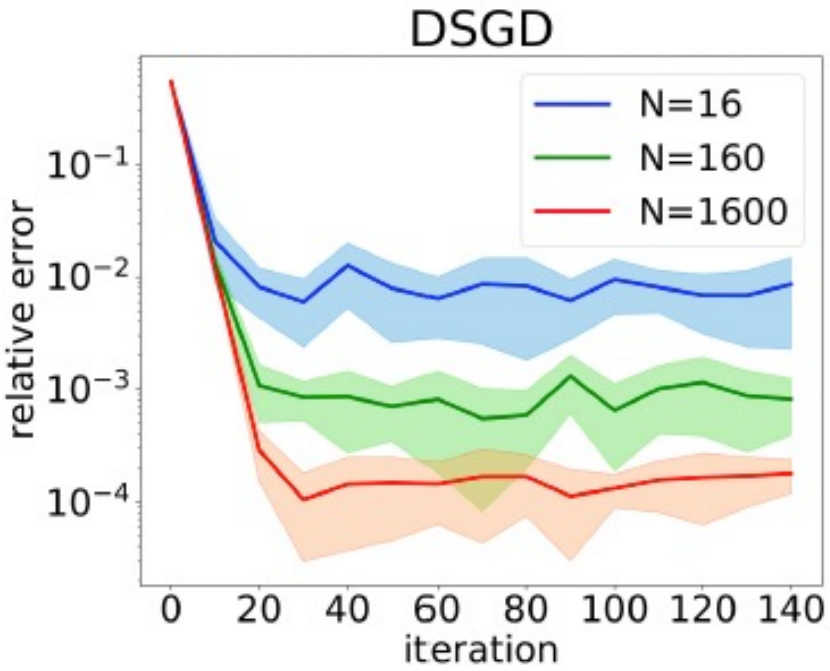
- When each $f_i(x)$ is strongly-convex, decentralized algorithms typically converge as (α is learning rate)

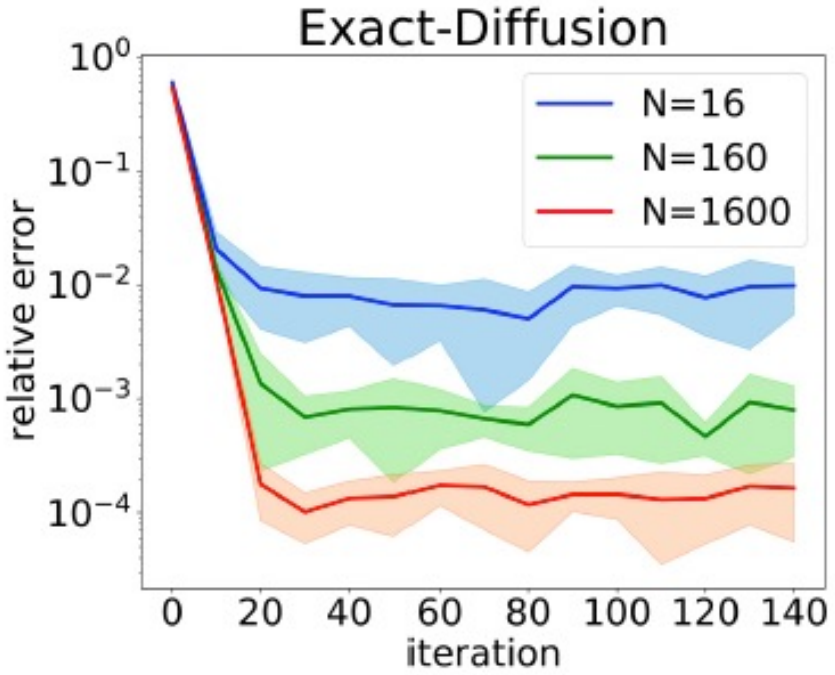
$$\limsup_{k\to\infty} \quad \mathbb{E}\|\mathbf{x}^k-\mathbf{x}^\star\|^2 = O\left(\frac{\alpha\sigma^2}{n}+\alpha^2\right)$$

The accuracy improves linearly with the number of nodes; **Linear speedup!**

Linear speedup

- Numerical experiments in logistic regression





Linear speedup

北京大学 PEKING UNIVERSITY

- However, to achieve linear speedup, a **network-dependent learning rate** has to be used

METHODS	LEARNING RATE	CONVERGENT ACC.
D-SGD [12]	$O(\frac{1-\lambda}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$
ED/NIDS [15]	$O(\frac{1}{L})$	N.A.
ED/NIDS [17], [24]	$O(\frac{1-\lambda}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$
GT [20], [31]	$O(\frac{(1-\lambda)^2}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$
GT [22], [23], [26]	$O(\frac{1-\lambda}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$

where λ measures the connectivity of the network; **unknown to each node**

- Open question: **Can we achieve linear speedup using network-independent learning rate?**

北京大学
PEKING UNIVERSITY

Part 02

Exact-Diffusion

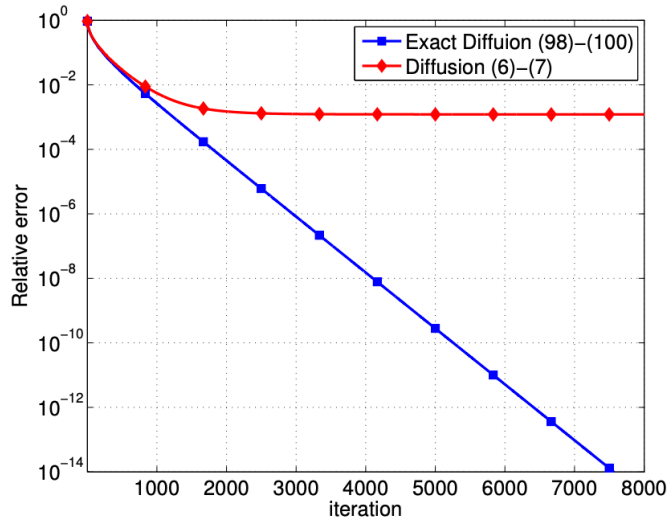
Exact-Diffusion

- Recall the optimization problem

$$\min_{x\in\mathbb{R}^d} \quad f(x)=\frac{1}{n}\sum_{i=1}^{n} f_i(x), \quad \text{where} \quad f_i(x)=\mathbb{E}_{\xi_i\sim D_i} F(x;\xi_i).$$

- Exact diffusion [YYZ+2017] (also known as D2 [TLY+2018]) is a popular decentralized algorithm

$$\begin{aligned}
\boldsymbol{\psi}^{k+1} &= \mathbf{x}^k - \alpha\nabla\mathbf{F}(\mathbf{x}^k;\xi^k), && \text{(local update)}\\
\boldsymbol{\phi}^{k+1} &= \boldsymbol{\psi}^{k+1} + \mathbf{x}^k - \boldsymbol{\psi}^k, && \text{(bias correction)}\\
\mathbf{x}^{k+1} &= \mathbf{W}\boldsymbol{\phi}^{k+1}. && \text{(communication)}
\end{aligned}$$



- Exact diffusion can correct the influence of data heterogeneity

Exact-Diffusion: Primal-dual reformulation

- To analyze the convergence, we transform Exact-Diffusion to the primal-dual form

$$\begin{aligned}\mathbf{v}^{k+1} &= \mathbf{x}^k - \alpha\nabla\mathbf{F}(\mathbf{x}^k;\xi^k) - \mathbf{B}\mathbf{y}^k,\\ \mathbf{x}^{k+1} &= \mathbf{W}\mathbf{v}^{k+1},\\ \mathbf{y}^{k+1} &= \mathbf{y}^k + \mathbf{B}\mathbf{v}^{k+1}.\end{aligned}$$

- $\mathbf{W}$ is the weight matrix associated with the network topology, and $\mathbf{B} = (\mathbf{I} - \mathbf{W})^{1/2}$
- By removing the dual variable $\mathbf{y}$, the above update reduces to Exact-Diffusion in the last page

Assumptions

北京大学 PEKING UNIVERSITY

Assumption 1 (Weight matrix)

The weight matrix W is assumed to be primitive, positive semidefinite, and doubly stochastic.

Assumption 2 (Smoothness and convexity)

Each function $f_i(x)$ is L-smooth and μ-strongly-convex for some $L \geq \mu > 0$

Assumption 3 (Gradient noise)

For all nodes indices $i = 1, \ldots, n$ and iterations $k = 0, 1, \ldots,$ we assume each stochastic gradient $\nabla F_i(x_i^k; \xi_i^k)$ is unbiased and has bounded variance.

Our first trial: Convergence analysis I

北京大学 PEKING UNIVERSITY

- Reference [LSY17] proves the convergence of Exact-Diffusion in the **deterministic** scenario with network-independent learning rate We first follow the analysis in [LSY17], which directly analyzes
- It is very natural to extend [LSY17] to **stochastic** scenario, which directly bounds $\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2$

Theorem 1 (Exact-Diffusion Convergence I)

Under Assumptions 1–3, if $\alpha \leq \frac{1}{L}$, it then holds that

$$\mathbb{E}\|\mathbf{x}^{k+1} - \mathbf{x}^\star\|^2 + \mathbb{E}\|\mathbf{y}^{k+1} - \mathbf{y}^\star\|^2 \leq \rho\left(\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 + \mathbb{E}\|\mathbf{y}^k - \mathbf{y}^\star\|^2\right) + n\alpha^2\sigma^2$$

where $\rho = \max\{\lambda, (1-\mu\alpha)^2\} \in (0,1)$.

- The above result implies $\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\|x_i^k - x^\star\|^2 = C\rho^k + \frac{\alpha\sigma^2}{1-\lambda}$; **no linear speedup**

Z. Li, W. Shi, and M. Yan, “A decentralized proximal-gradient method with network independent step-sizes and separated convergence rates,” IEEE TSP 2019

Our second trial: Convergence analysis II

- Most linear speedup works study optimality error and consensus error separately.

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 \le 2\underbrace{\mathbb{E}\|\mathbf{x}^k - \bar{\mathbf{x}}^k\|^2}_{\text{consensus error}} + 2\underbrace{\mathbb{E}\|\bar{\mathbf{x}}^k - \mathbf{x}^\star\|^2}_{\text{optimality error}}$$

- Following this idea, we prove

Lemma 1 (Optimality and Consensus error)

Under Assumptions 1–3, if $\alpha \le \frac{1}{4L}$, then we have

$$\mathbb{E}\|\bar{x}^{k+1} - x^\star\|^2 \le (1-\mu\alpha)\mathbb{E}\|\bar{x}^k - x^\star\|^2 + \frac{3L\alpha}{n}\mathcal{E}^k + \frac{\alpha^2\sigma^2}{n} \qquad \text{(Optimality error)}$$

$$\mathcal{E}^{k+1} \le \sqrt{\lambda}\,\mathcal{E}^k + \frac{\alpha^2\lambda^2 n\sigma^2}{\lambda} + \frac{\alpha^2\lambda^2 L^2}{\lambda(1-\sqrt{\lambda})}\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2. \qquad \text{(Consensus error)}$$

- The term $\mathcal{E}^k$ relates to the consensus error $\mathbb{E}\|\mathbf{x}^k - \bar{\mathbf{x}}^k\|^2$ and $\mathbb{E}\|\mathbf{y}^k - \bar{\mathbf{y}}^k\|^2$

Our second trial: Convergence analysis II

- An additional term $\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2$ exists in the evolution of the consensus error

$$\mathcal{E}^{k+1} \leq \sqrt{\lambda}\ \mathcal{E}^k + \frac{\alpha^2\lambda^2 n\sigma^2}{\underline{\lambda}} + \frac{\alpha^2\lambda^2 L^2}{\underline{\lambda}(1-\sqrt{\lambda})}\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 \qquad \text{(consensus error)}$$

- W can bound $\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2$ by

$$\begin{aligned}\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 &\leq 2\mathbb{E}\|\mathbf{x}^k - \bar{\mathbf{x}}^k\|^2 + 2\mathbb{E}\|\bar{\mathbf{x}}^k - \mathbf{x}^\star\|^2 \\ &\leq \mathcal{E}^k + 2n\mathbb{E}\|\bar{x}^k - x^\star\|^2\end{aligned}$$

- Substituting the above inequality to consensus error and setting $\sqrt{\lambda} + c_2\alpha^2 \leq (1+\sqrt{\lambda})/2 := \bar{\lambda}$

$$\begin{aligned}\mathcal{E}^{k+1} &\leq (\sqrt{\lambda} + c_1\alpha^2)\ \mathcal{E}^k + c_2\alpha^2\mathbb{E}\|\bar{x}^k - x^\star\|^2 + \frac{\alpha^2\lambda^2 n\sigma^2}{\underline{\lambda}} \\ &\leq \bar{\lambda}\mathcal{E}^k + c_2\alpha^2\mathbb{E}\|\bar{x}^k - x^\star\|^2 + c_3 n\alpha^2\end{aligned}$$

Our second trial: Convergence analysis II

- As a result, when α is **network-dependent** such that $\sqrt{\lambda}+c_2\alpha^2 \le (1+\sqrt{\lambda})/2$, we have

$$\mathcal{E}^{k+1} \le \bar{\lambda}\mathcal{E}^k + c_2\alpha^2\mathbb{E}\|\bar{x}^k - x^\star\|^2 + c_3 n\alpha^2 \qquad \text{(consensus error)}$$

- Combining with the optimality error, we achieve the following linear dynamics

$$\underbrace{\begin{bmatrix} \mathbb{E}\|\bar{x}^{k+1}-x^\star\|^2 \\ \mathcal{E}^{k+1}\end{bmatrix}}_{\mathbf{z}^{k+1}} \le \underbrace{\begin{bmatrix} 1-\mu\alpha & \frac{L\alpha}{n} \\ c_2\alpha^2 & \bar{\lambda}\end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} \mathbb{E}\|\bar{x}^{k}-x^\star\|^2 \\ \mathcal{E}^{k}\end{bmatrix}}_{\mathbf{z}^{k}} + \underbrace{\begin{bmatrix} \frac{\alpha^2\sigma^2}{n} \\ c_3 n\alpha^2\end{bmatrix}}_{\mathbf{b}}$$

- Using this dynamic, we can prove linear speedup with network-dependent learning rate $\alpha = O(\frac{1-\lambda}{L})$

$$\frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\|x_i^k - x^\star\|^2 = O\Big(\frac{\alpha\sigma^2}{n}+\alpha^2\Big)$$

A brief summary

- In analysis I, when we examine $\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2$ directly, we prove network-independent learning rate but not linear speedup

$$\limsup_{k\to\infty} \quad \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\|x_i^k - x^\star\|^2 = O\left(\alpha\sigma^2 + \alpha^2\right) \quad \text{with} \quad \alpha = O(1/L)$$

- In analysis II, when we examine $\mathbb{E}\|\bar{x}^k - x^\star\|^2$ and $\mathbb{E}\|\mathbf{x}^k - \bar{\mathbf{x}}^k\|^2$ separately, we prove linear speedup convergence but with network-dependent learning rate

$$\limsup_{k\to\infty} \quad \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}\|x_i^k - x^\star\|^2 = O\left(\frac{\alpha\sigma^2}{n} + \alpha^2\right) \quad \text{with} \quad \alpha = O(\frac{1-\lambda}{L})$$

- Seems **impossible** to achieve both linear speedup and network-independent learning rate

PART 03

Achieving linear speedup with network-independent learning rate

A novel analysis

北京大学 PEKING UNIVERSITY

- In analysis I, we prove the convergence with network-independent learning rate

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 = C\rho^k + \frac{n\alpha\sigma^2}{1-\lambda} \qquad \text{(General error)}$$

- In analysis II, we prove the convergence

$$\mathbb{E}\|\bar{x}^{k+1} - x^\star\|^2 \le (1-\mu\alpha)\mathbb{E}\|\bar{x}^k - x^\star\|^2 + \frac{3L\alpha}{n}\mathcal{E}^k + \frac{\alpha^2\sigma^2}{n} \qquad \text{(Optimality error)}$$

$$\mathcal{E}^{k+1} \le \sqrt{\lambda}\,\mathcal{E}^k + \frac{\alpha^2\lambda^2 n\sigma^2}{\lambda} + \frac{\alpha^2\lambda^2 L^2}{\lambda(1-\sqrt{\lambda})}\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 \qquad \text{(Consensus error)}$$

- Previously, the following bound of $\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2$ leads to network-dependent learning rate

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 \le \mathcal{E}^k + 2n\mathbb{E}\|\bar{x}^k - x^\star\|^2$$

A novel analysis

北京大学 PEKING UNIVERSITY

- New idea: substituting the general error term to the consensus term

$$\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 = C\rho^k + \frac{n\alpha\sigma^2}{1-\lambda} \qquad \text{(General error)}$$

$$\mathcal{E}^{k+1} \leq \sqrt{\lambda}\,\mathcal{E}^k + \frac{\alpha^2\lambda^2 n\sigma^2}{\lambda} + \frac{\alpha^2\lambda^2 L^2}{\lambda(1-\sqrt{\lambda})}\mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 \qquad \text{(Consensus error)}$$

- This leads to

$$\mathbb{E}\|\mathbf{x}^k - \bar{\mathbf{x}}^k\|^2 \leq \mathcal{E}^{k+1} \leq \sqrt{\lambda}\,\mathcal{E}^k + O(\alpha^2 n\sigma^2) + O(\rho^k)$$

$$\leq O\left(\frac{\alpha^2 n\sigma^2}{1-\lambda}\right) + \underbrace{O(\rho^k + \lambda^k)}_{\text{exponentially decay}} \qquad \text{with} \quad \alpha = O(1/L)$$

A novel analysis

- Recall the optimality and consensus error with $\alpha = O(1/L)$

$$\mathbb{E}\|\bar{x}^{k+1} - x^\star\|^2 \leq (1-\mu\alpha)\mathbb{E}\|\bar{x}^k - x^\star\|^2 + \frac{3L\alpha}{n}\mathcal{E}^k + \frac{\alpha^2\sigma^2}{n} \qquad \text{(Optimality error)}$$

$$\mathcal{E}^k = O(\frac{\alpha^2 n\sigma^2}{1-\lambda}) + O(\rho^k + \lambda^k) \qquad \text{(Consensus error)}$$

- Substituting the consensus error into the optimality error, with $\alpha = O(1/L)$ we have

$$\begin{aligned}
\mathbb{E}\|\bar{x}^{k+1} - x^\star\|^2 &\leq (1-\mu\alpha)\mathbb{E}\|\bar{x}^k - x^\star\|^2 + \frac{\alpha^2\sigma^2}{n} + O(\rho^k + \lambda^k) \\
&\leq \frac{\alpha\sigma^2}{n} + O(\rho^k + \lambda^k + (1-\mu\alpha)^k) \\
&\leq \frac{\alpha\sigma^2}{n} + O(\rho^k) \qquad (\text{ Since } \rho = \max\{\lambda, 1-\mu\alpha\})
\end{aligned}$$

Main theorem

北京大学 PEKING UNIVERSITY

Theorem 2 (Linear speedup with network-independent learning rate)

Under Assumptions 1–3, if $\alpha \leq \frac{1}{4L}$, Exact-Diffusion converges as follows

$$\limsup_{k\to\infty} \mathbb{E}\|\mathbf{x}^k - \mathbf{x}^\star\|^2 = O\left(\frac{\alpha\sigma^2}{n} + \alpha^2\right)$$

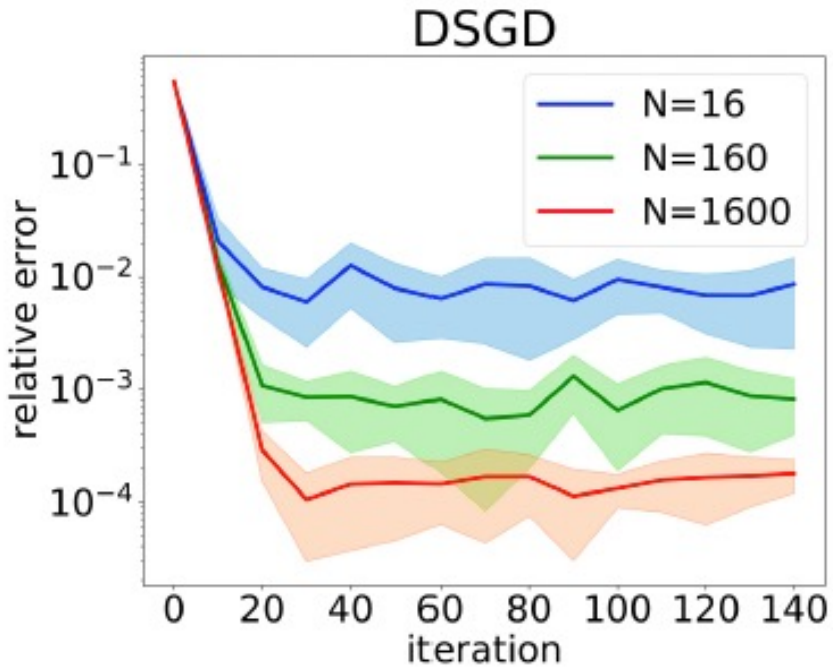
Following our analysis, DSGD can also achieve linear speedup with network-independent learning rate

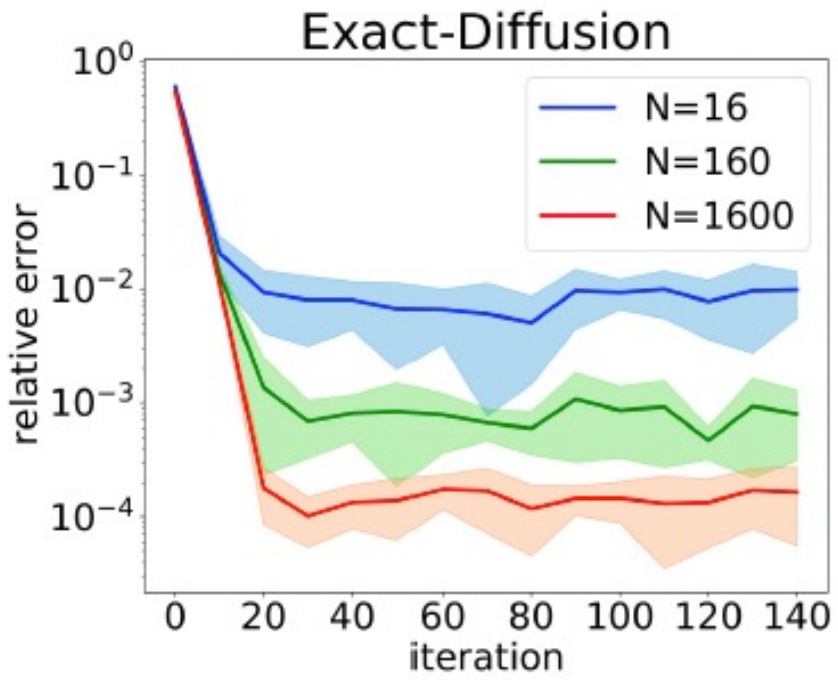
Comparison with existing results

METHODS	LEARNING RATE	CONVERGENT ACC.
D-SGD [12]	$O(\frac{1-\lambda}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$
ED/NIDS [15]	$O(\frac{1}{L})$	N.A.
ED/NIDS [17], [24]	$O(\frac{1-\lambda}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$
GT [20], [31]	$O(\frac{(1-\lambda)^2}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$
GT [22], [23], [26]	$O(\frac{1-\lambda}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$
D-SGD (Thm. 3)	$O(\frac{1}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$
ED/NIDS (Thm. 2)	$O(\frac{1}{L})$	$O(\frac{\alpha\sigma^2}{n})+O(\alpha^2)$

Numerical studies

- Numerical experiments in logistic regression
- Ring topology with difference size; Different size has different λ values
- We use the same learning rate no matter how λ varies





北京大学
PEKING UNIVERSITY

Towards Better Understanding the Influence of Directed Networks on Decentralized Stochastic Optimization

Liyuan Liang Xinmeng Huang Ran Xin Kun Yuan

https://arxiv.org/abs/2312.04928

Spectral gap is not enough; Needs new effective metrics

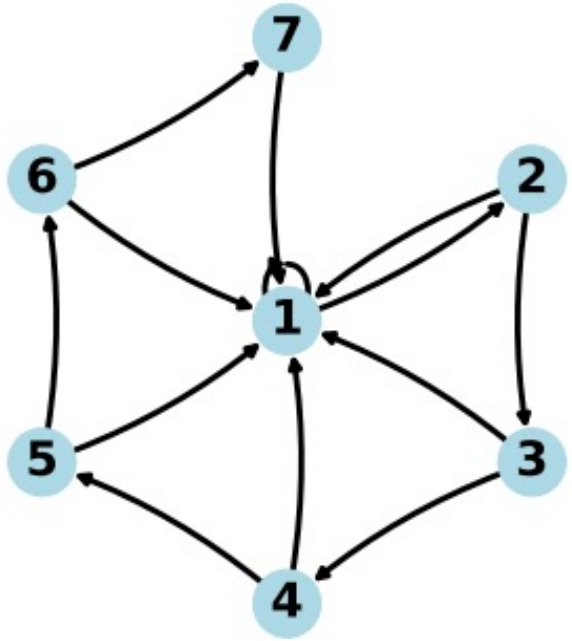
北京大学 PEKING UNIVERSITY

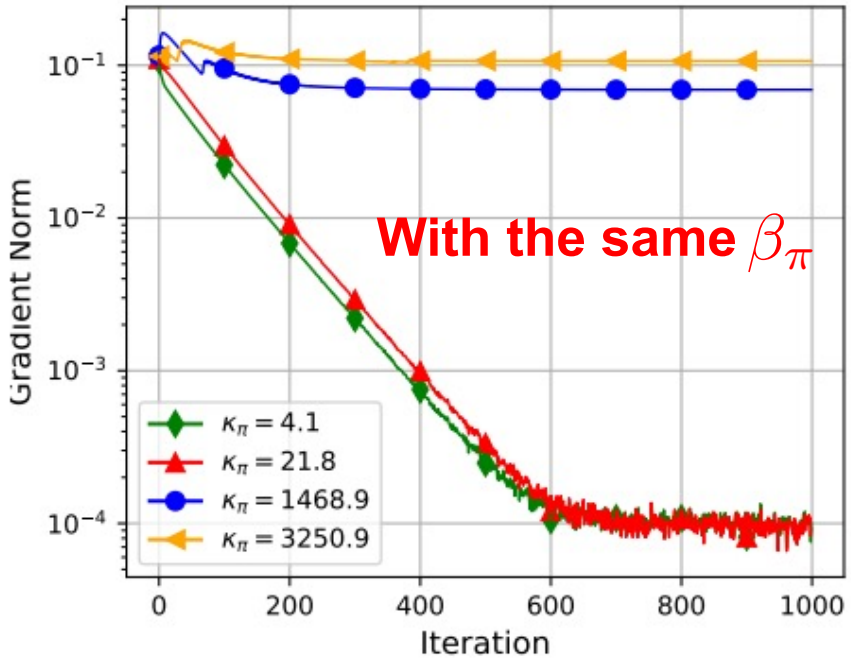
- Spectral gap is NOT enough to gauge the influence of directed networks

$$\beta_\pi = \|W - \pi \mathbf{1}^T\|_\pi \in [0, 1)$$

Needs new metric!

where π is the right Perron eigenvector $W\pi = \pi, \quad \mathbf{1}^T\pi = 1$





Lower bound and optimal algorithms

北京大学 PEKING UNIVERSITY

- Convergence **Lower bound** for non-convex decentralized stochastic optimization with column stochastic W

$$\mathbb{E}[\|\nabla f(x^{(K)})\|_2^2] = \Omega\left(\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{(1+\ln(\kappa_\pi))L\Delta}{(1-\beta_\pi)K}\right)$$

- We developed **optimal** decentralized push-sum algorithms to attain such lower bound

Algorithm	Rate (A.)	Rate (F.T.)	Transient Stage
Gradient-Push [3]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.
Push-DIGing [23]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.
Push-DIGing Theorem 2	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{\beta_\pi\kappa_\pi^3(1+\kappa_\pi\beta_\pi)}{(1-\beta_\pi)^2K}$	$\frac{n\kappa_\pi^8}{(1-\beta_\pi)^4}$
MG-Push-DIGing Theorem 3	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{(1+\ln(\kappa_\pi))L\Delta}{(1-\beta_\pi)K}$	$\frac{n(1+\ln(\kappa_\pi))^2}{(1-\beta_\pi)^2}$
Lower Bound Theorem 1	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{(1+\ln(\kappa_\pi))L\Delta}{(1-\beta_\pi)K}$	$\frac{n(1+\ln(\kappa_\pi))^2}{(1-\beta_\pi)^2}$

北京大学
PEKING UNIVERSITY

Thank you!

Kun Yuan homepage: https://kunyuan827.github.io/

We have openings for PostDocs