北京大学
PEKING UNIVERSITY

Towards Decentralized Optimization over Digraphs: Effective metrics, lower bound, and optimal algorithms

Kun Yuan

Center for Machine Learning Research @ Peking University

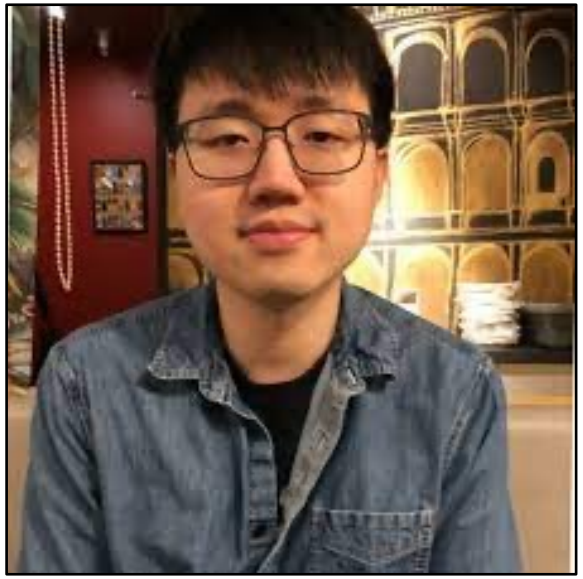
Dec. 28, 2023

Joint work with

北京大学 PEKING UNIVERSITY

Liyuan Liang (PKU)

Xinmeng Huang (UPenn)



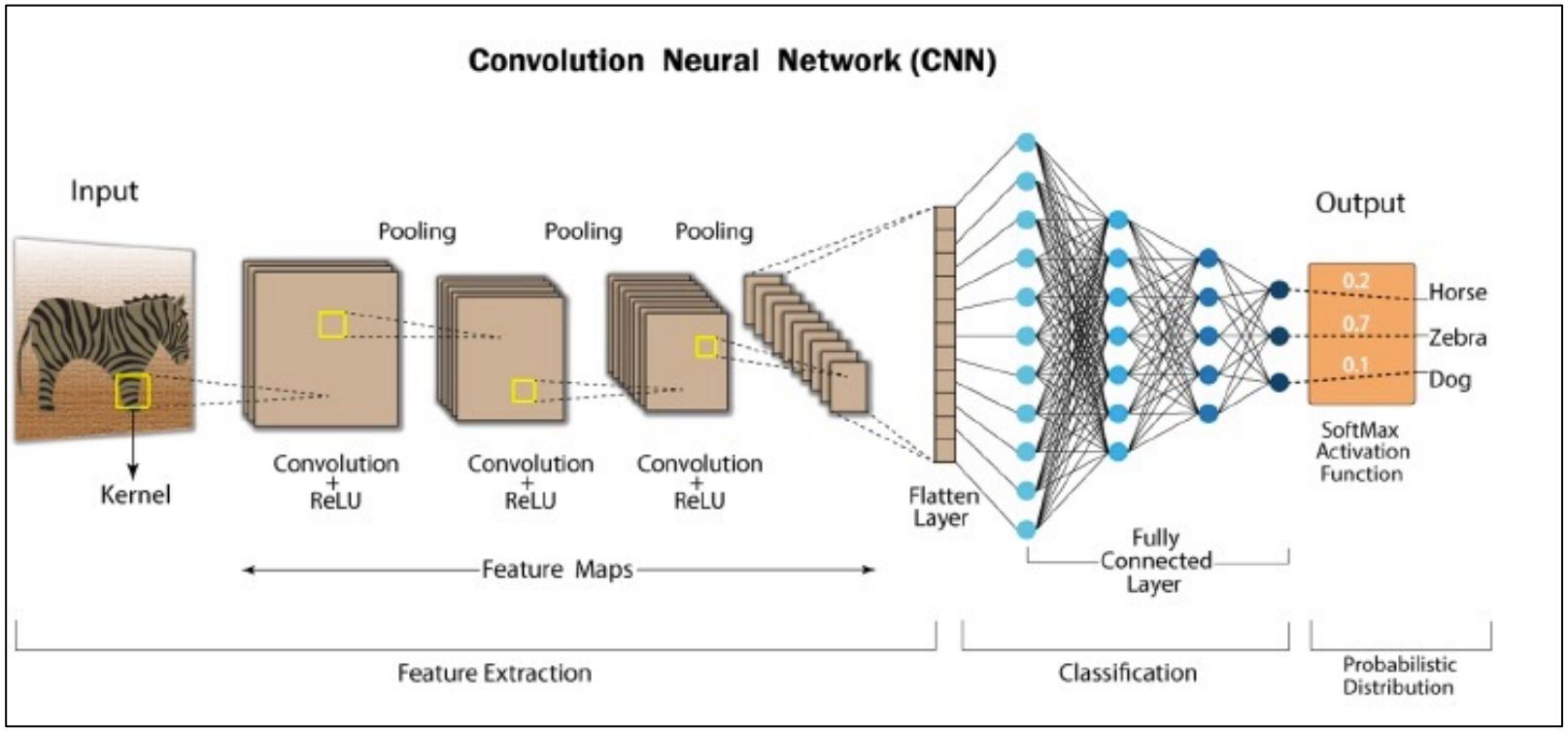
Ran Xin (ByteDance)

北京大学
PEKING UNIVERSITY

Part 01

Decentralized Optimization

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

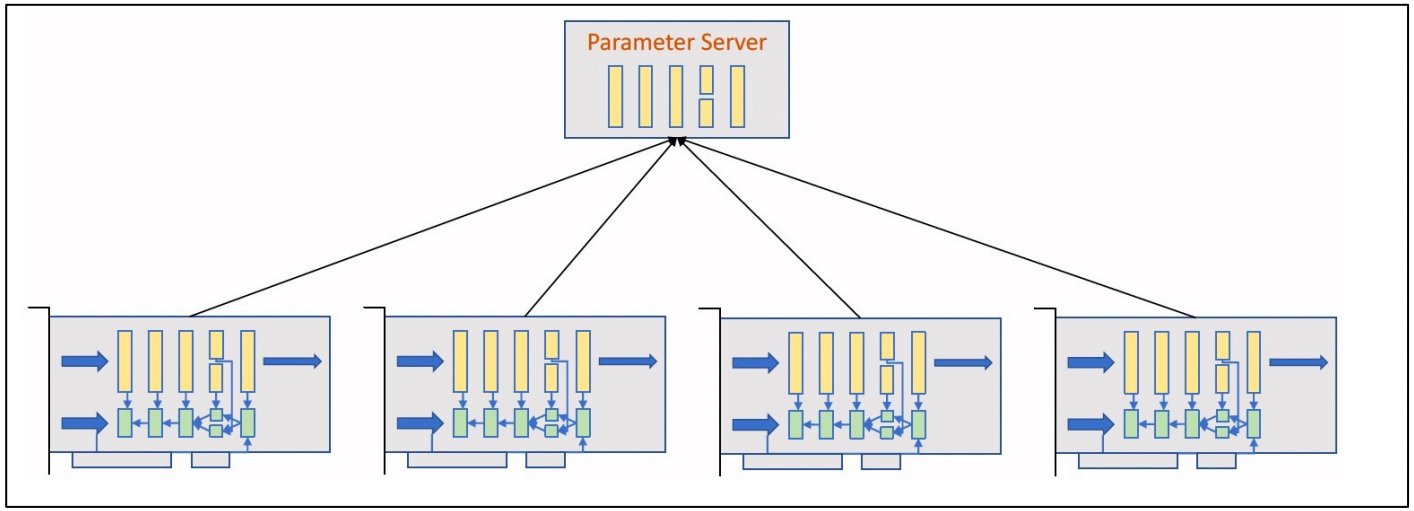
Vanilla parallel stochastic gradient descent (PSGD)

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

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

- 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

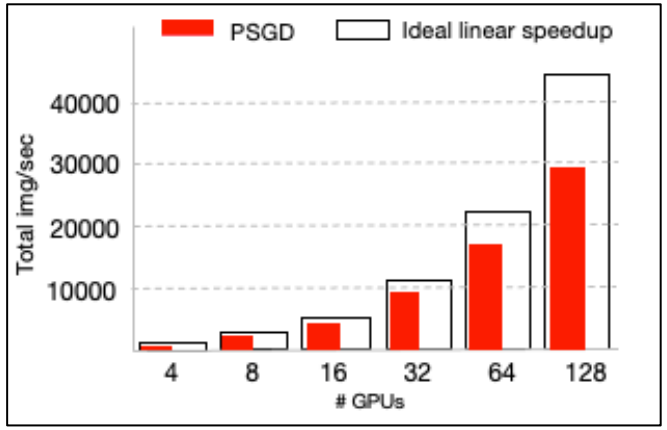
Vanilla parallel stochastic gradient descent (PSGD)



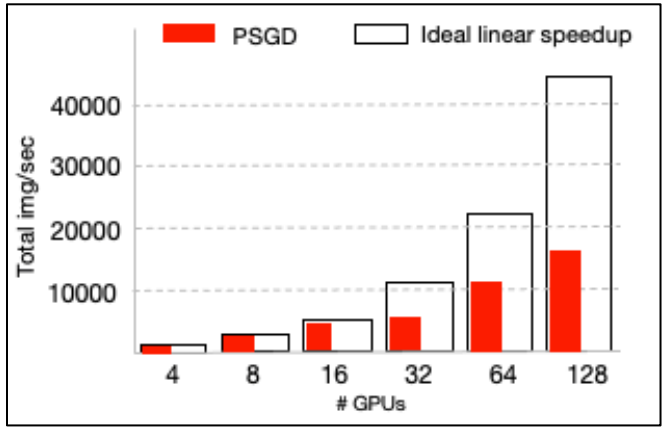
- Global average incurs $O(n)$ comm. overhead; **proportional to network size n**
- When network size n is large, PSGD suffers severe communication overhead

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**

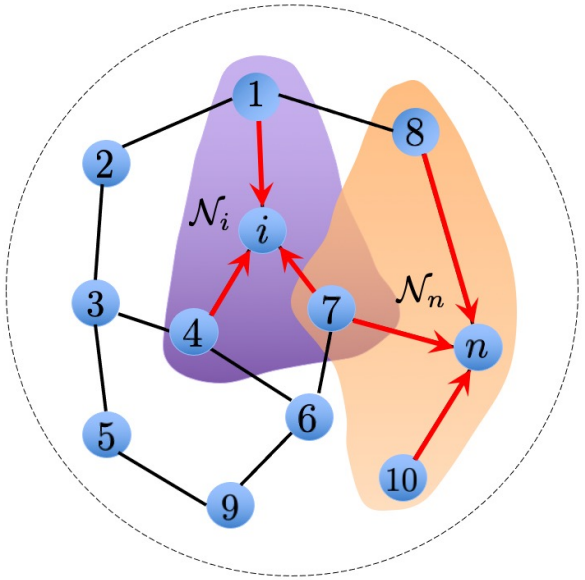
B. Ying, K. Yuan, H. Hu, Y. Chen and W. Yin, “BlueFog: Make decentralized algorithms practical for optimization and deep learning”, arXiv: 2111. 04287, 2021

Decentralized SGD (DSGD)

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

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

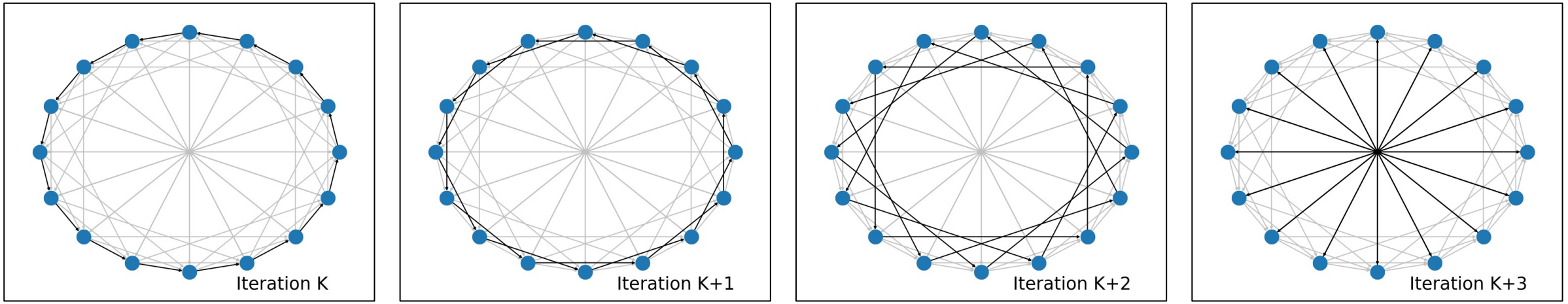
$$x_i^{(k+1)} = \sum_{j \in \mathcal{N}_i} w_{ij} x_j^{(k+\frac{1}{2})} \quad \text{(Partial averaging)}$$



- 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

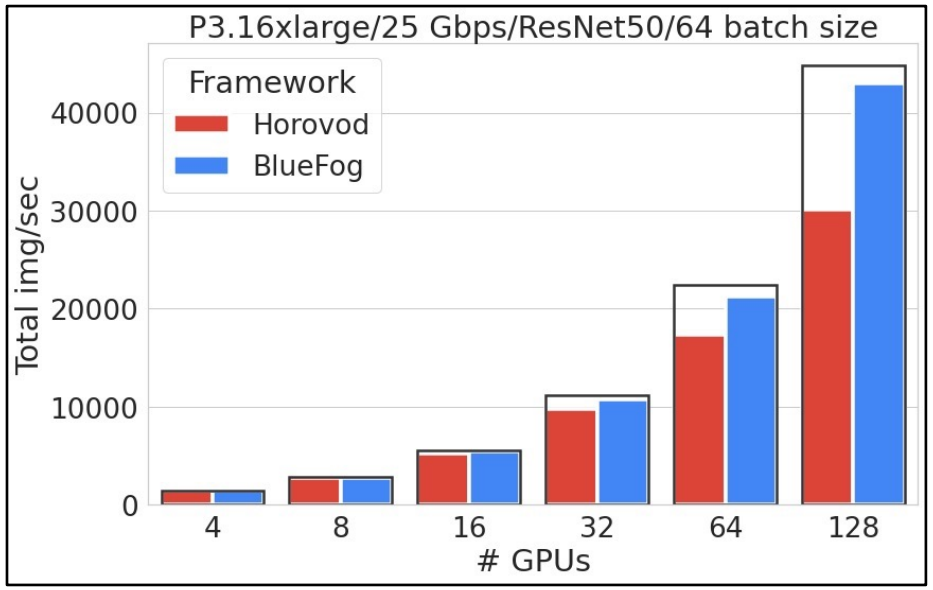
- DSGD saves more communications per iteration for larger models

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

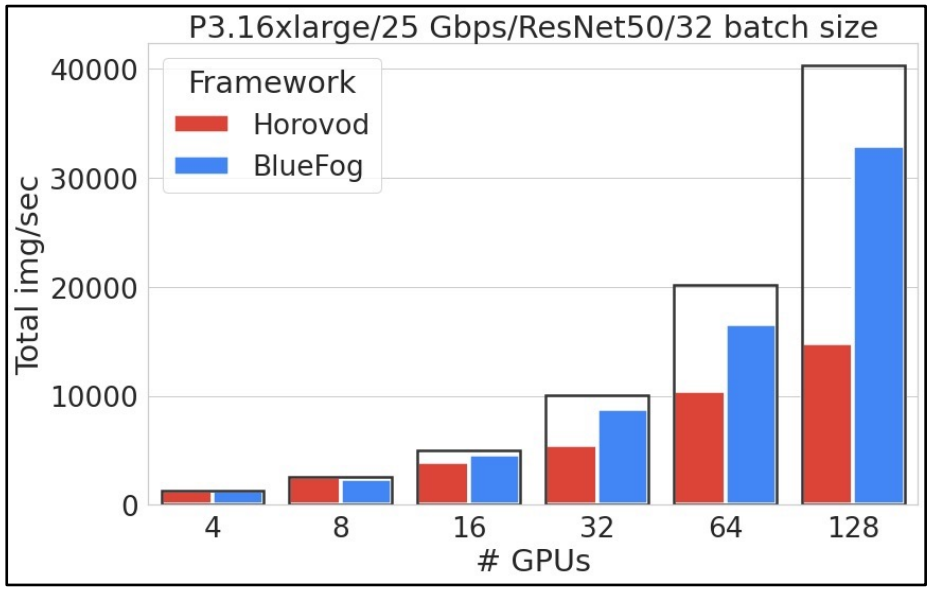
DSGD is more communication-efficient than PSGD

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

北京大学
PEKING UNIVERSITY

Part 02

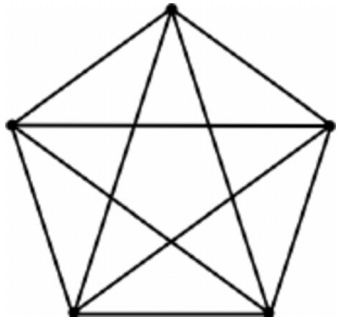
Influence of Undirected Network Topology

However, DSGD has slower convergence

- The efficient comm. comes with a cost: **slower convergence**
- Partial average $x_i^+ = \sum w_{ij} x_j$ is less effective to aggregate information than global average
- We stack all weights into a weight matrix $W = [w_{ij}]_{i=1,j=1}^{n} \in \mathbb{R}^{n \times n}$:

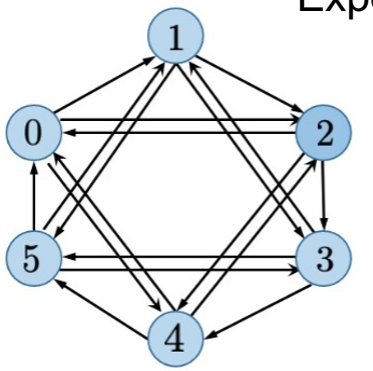
$$w_{ij} \begin{cases} > 0 & \text{if node } j \text{ can talk to } i \\ = 0 & \text{if node } j \text{ cannot talk to } i \end{cases}$$

Fully-connected graph



$$W = \frac{1}{5}\mathbb{1}\mathbb{1}^T = \begin{bmatrix} \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \end{bmatrix}$$

Exponential graph



$$W = \begin{bmatrix} \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}$$

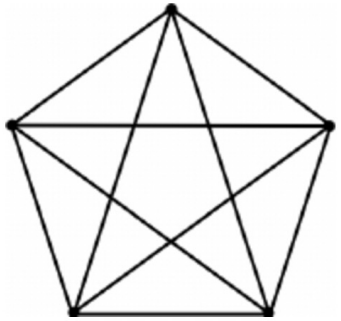
Doubly stochastic weight matrix

- Given an undirected graph, we typically associate it with a doubly stochastic weight matrix

Assumption 1. The weight matrix W is primitive and doubly-stochastic, i.e.

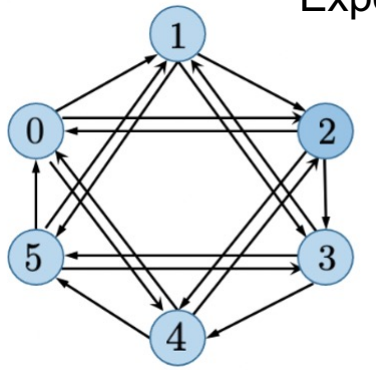
$$W\mathbf{1}_n = \mathbf{1}_n \quad \text{and} \quad \mathbf{1}_n^T W = \mathbf{1}_n^T$$

Fully-connected graph



$$W = \frac{1}{5}\mathbb{1}\mathbb{1}^T = \begin{bmatrix} \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \end{bmatrix}$$

Exponential graph



$$W = \begin{bmatrix} \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}$$

Graph spectral gap

- Weight matrix W reflects the connectivity of the underlying decentralized network

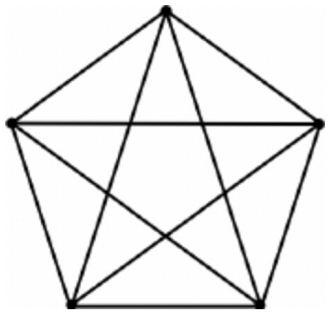
Lemma 1 Under Assumptions 1, it holds that $\|W - \mathbf{1}_n \mathbf{1}_n^T/n\| \leq \rho \in [0, 1)$

- Quantity ρ measures the gap between W and the fully-connected weight matrix $\mathbf{1}_n \mathbf{1}_n^T/n$

- For well-connected network, we have $\rho \to 0$

  For sparsely-connected network, we have $\rho \to 1$

- We define the **spectral gap** of W as $1 - \rho$



$$W = \frac{1}{5}\mathbb{1}\mathbb{1}^T = \begin{bmatrix} \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \\ \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} & \frac{1}{5} \end{bmatrix}$$

DSGD convergence rate

- Convergence comparison (non-convex and **data-homogeneous** scenario) [KLB+20]:

$$\text{P-SGD}: \quad \frac{1}{T}\sum_{k=1}^{T}\mathbb{E}\|\nabla f(\bar{x}^{(k)})\|^2 = O\left(\frac{\sigma}{\sqrt{nT}}\right)$$

$$\text{D-SGD}: \quad \frac{1}{T}\sum_{k=1}^{T}\mathbb{E}\|\nabla f(\bar{x}^{(k)})\|^2 = O\left(\frac{\sigma}{\sqrt{nT}} + \underbrace{\frac{\rho^{2/3}\sigma^{2/3}}{T^{2/3}(1-\rho)^{1/3}}}_{\text{extra overhead}}\right)$$

where σ^2 is the gradient noise, and T is the number of iterations

- D-SGD can asymptotically converge as fast as P-SGD when $T \to \infty$; the first term dominates; reach **linear speedup** asymptotically

- But D-SGD **requires more iteration** to reach that stage due to the overhead caused by partial average

Transient iterations

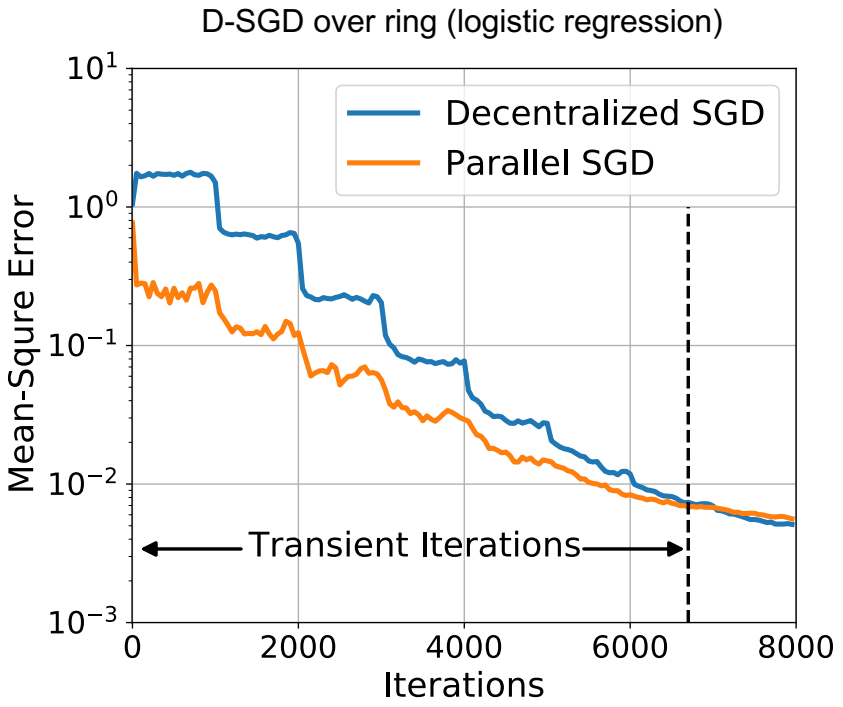
- Definition [POP21]: number of iterations before D-SGD achieves linear speedup

- D-SGD for non-convex and data-homogeneous scenario has $O(n^3(1-\rho)^{-2})$ transient iterations

$$\frac{\rho^{2/3}\sigma^{2/3}}{T^{2/3}(1-\rho)^{1/3}} \leq \frac{\sigma}{\sqrt{nT}} \quad \Longrightarrow \quad O(\frac{\rho^4 n^3}{(1-\rho)^2})$$

- Topology significantly influence the trans. stage.

- Sparse topology $\rho \to 1$ incurs longer tran. Iters.



Techniques to reduce transient iterations

- Remove the influence of data heterogeneity
  - Exact-Diffusion [YAYS20, YAH23] (also known as D2 [TLY+18])
  - Gradient tracking [KLS22, AY22]
- Develop multi-step gossip strategy to accelerate convergence
  - DeTAG [LS21]; MG-DSGD[YHC+22]; MG-Exact-Diffusion[YAH23]
- Develop sparse and effective network topologies
  - Exponential Graph [YYC+21]; EquiTopo [SLJ+22]
  - CECA-DSGD [DJY+23]

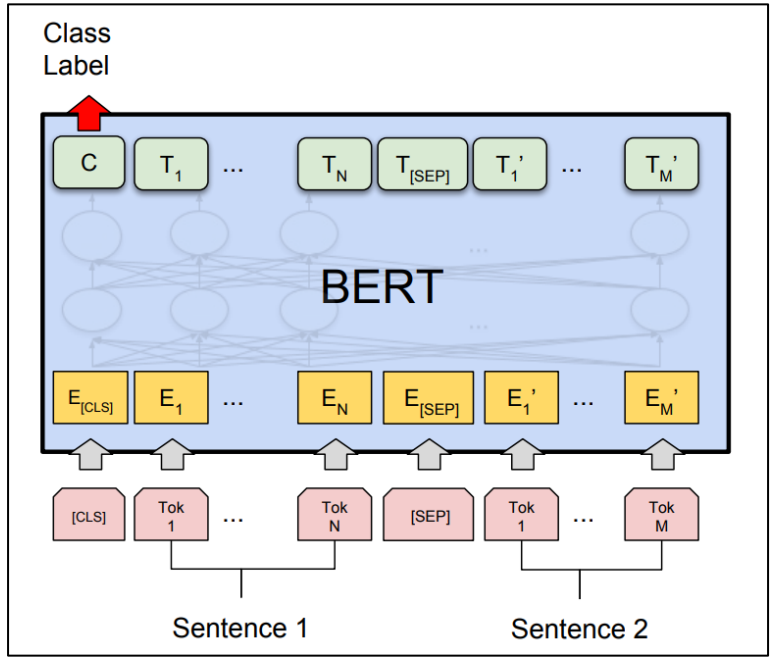
Algorithm	Tran. Iters.
DSGD	$\mathcal{O}(n^3/(1-\rho)^4)$
+ ED/D2	$\mathcal{O}(n^3/(1-\rho)^2)$
+ MG	$\mathcal{O}(n/(1-\rho))$
+ EquiTopo	$\mathcal{O}(n)$

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
Ring	76.16	11.6	76.14	6.5	76.16	3.3	75.62	1.8
one-peer exp.	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

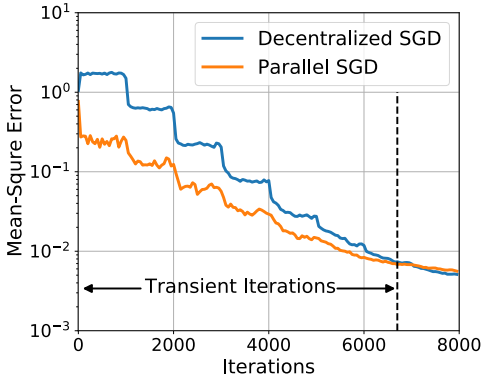
Table. Comparison in loss and training time [CYZ+21]

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

Review the research path on decentralized optimization

- We find that DSGD is efficient in communication but slow in convergence



- We identify the spectral gap metric to quantify the influence of network topology on convergence

$$\|W - \mathbf{1}_n \mathbf{1}_n^T / n\| \leq \rho$$

- We establish the transient iteration complexity

$$\mathcal{O}(n^3/(1-\rho)^4)$$

- We develop techniques to reduce transient iterations

Remove data heterogeneity | Multiple gossip | Effective graphs

The key step is to **identify an effective metric** that captures the influence of network topology

北京大学
PEKING UNIVERSITY

Part 03

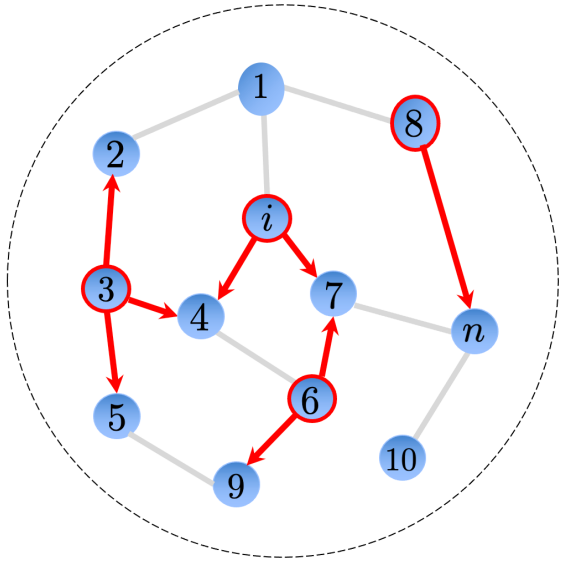
Decentralized Optimization over Digraphs

Directed network topology

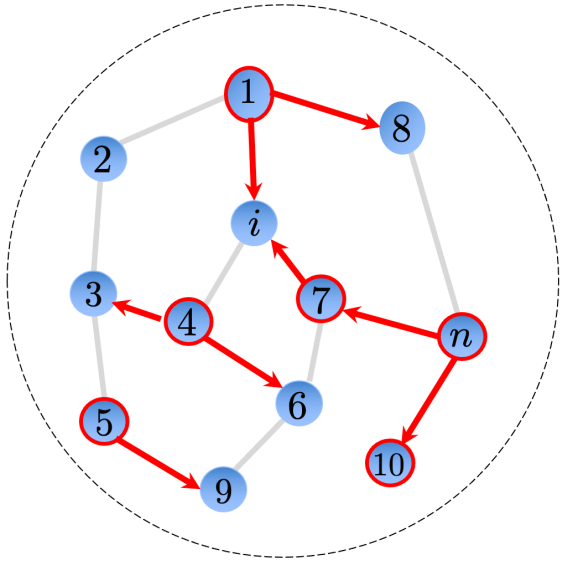
Social networks are directed

Asynchronous networks are directed

(nodes are activated at different iteration)



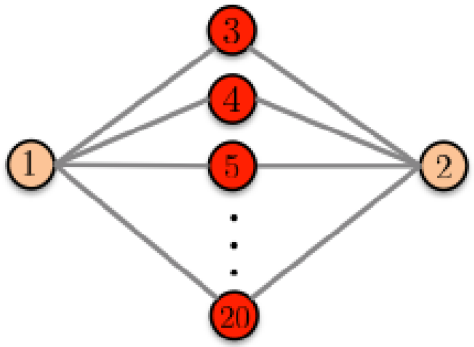
iteration k



iteration k+1

Doubly-stochastic matrices cannot be constructed easily

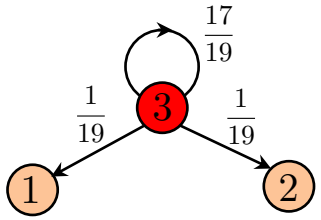
- Doubly-stochastic matrices typically cannot be constructed over digraphs
- Even for undirected networks, sometimes singly-stochastic weight matrix is more preferred



Highly-unbalanced network

Metropolis rule

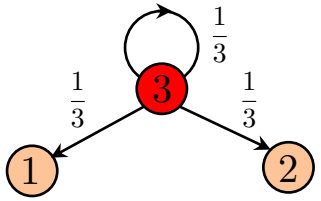
$$W = I - L/19$$



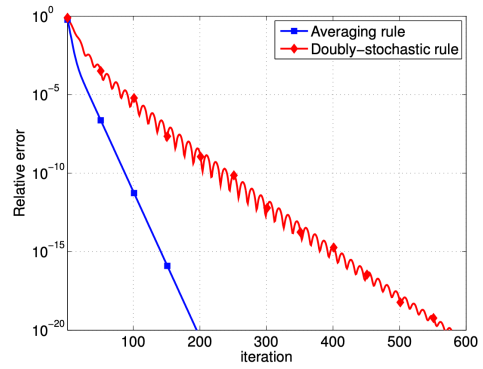
Doubly-stochastic matrix
(very close to identity)

Average rule

$$w_{ij} = \begin{cases} \frac{1}{\deg(j)} & \text{if } (j,i) \in \mathcal{E} \\ 0 & \text{otherwise} \end{cases}$$



Column-stochastic matrix

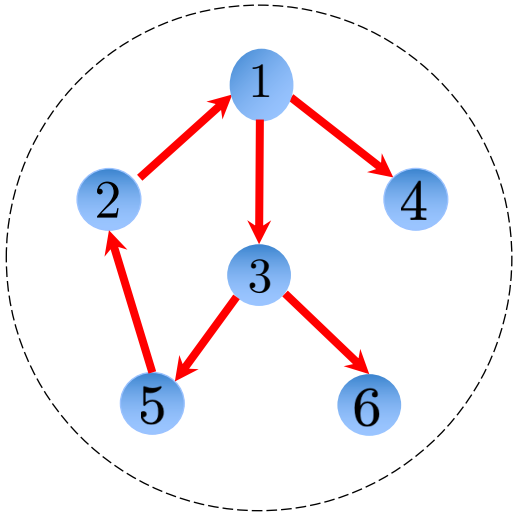


Column-stochastic performs
better than doubly-stochastic

Column-stochastic weight matrix

- This talk focuses on the **column-stochastic** weight matrix associated with a directed network
- A common way to construct the column-stochastic weight matrix

$$w_{ij} = \begin{cases} 1/(1+d_i^{\text{out}}) & \text{if directed edge } (j,i) \in \mathcal{E} \text{ or } j = i \\ 0 & \text{otherwise} \end{cases}$$



$$\begin{bmatrix} \frac{1}{3} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & \frac{1}{2} & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & 0 & \frac{1}{2} & 0 \\ 0 & 0 & \frac{1}{3} & 0 & 0 & 1 \end{bmatrix}$$

Column-stochastic weight matrix

北京大学 PEKING UNIVERSITY

Assumption 2.

The mixing matrix W is entry-wisely non-negative, primitive (i.e., all entries of W^{k_0} are positive for sufficiently large $k_0 \in \mathbb{N}_+$), and satisfies $\mathbf{1}_n^\top W = \mathbf{1}_n^\top$.

Lemma 2.

Under Assumption 2, there exists a unique equilibrium vector $\pi \in \mathbb{R}^n$ with positive entries such that $W\pi = \pi$ and $\mathbf{1}_n^\top \pi = 1$. Moreover, it holds that

$$\lim_{k\to\infty} W^k = \pi \mathbf{1}_n^\top$$

北京大学
PEKING UNIVERSITY

Part 04

Push-sum decentralized averaging

Column-stochastic matrix cannot enable global average

北京大学 PEKING UNIVERSITY

- Assume each node i maintains a local vector $z_i \in \mathbb{R}^n$
- We introduce $\mathbf{z} = [z_1^\top; z_2^\top; \cdots; z_n^\top] \in \mathbb{R}^{n \times d}$ and let $\mathbf{z}^{(0)} = \mathbf{z}$
- Column-stochastic matrix cannot enable global average

$$\mathbf{z}^{(k)} = W\mathbf{z}^{(k-1)} = W^k\mathbf{z}^{(0)} \longrightarrow \pi\mathbf{1}^\top\mathbf{z}^{(0)} \quad (\text{as } k \to \infty)$$

  which implies that $z_i^{(k)} \to \pi_i \sum_{j=1}^n z_j$ rather than $z_i^{(k)} \to (1/n)\sum_{j=1}^n z_j$

- We can correct the bias to enable global average

$$\mathbf{w}^{(k)} = \operatorname{diag}(n\pi)^{-1}\mathbf{z}^{(k)} \longrightarrow \operatorname{diag}(n\pi)^{-1}\pi\mathbf{1}_n^\top\mathbf{z} = (1/n)\mathbf{1}_n\mathbf{1}_n^\top\mathbf{z}$$

Push-sum decentralized averaging

(Bias correction) $\mathbf{w}^{(k)} = \mathrm{diag}(n\pi)^{-1}\mathbf{z}^{(k)} \longrightarrow \mathrm{diag}(n\pi)^{-1}\pi\mathbf{1}_n^\top\mathbf{z} = (1/n)\mathbf{1}_n\mathbf{1}_n^\top\mathbf{z}$

- However, the equilibrium vector π is not known in advance
- Push-sum decentralized averaging [BBT+10, TLR12, NO13]

$$\begin{aligned}\mathbf{z}^{(k+1)} &= W\mathbf{z}^{(k)} \\ v^{(k+1)} &= Wv^{(k)} \qquad \text{(starting with } v^{(0)} \text{ satisfying } \mathbf{1}_n^\top v^{(0)} = n\text{)} \\ V^{(k+1)} &= \mathrm{diag}(v^{(k+1)}) \\ \mathbf{w}^{(k+1)} &= {V^{(k+1)}}^{-1}\mathbf{z}^{(k+1)}.\end{aligned}$$

- It is guaranteed that $w_i^{(k)} \to (1/n)\sum_{j=1}^n z_j$

Push-sum decentralized optimization

北京大学 PEKING UNIVERSITY

- Many optimization algorithms over digraphs have been proposed based on push-sum averaging
  - Push-sum subgradient method [NO13]
  - Push-sum dual averaging [TLR12]
  - Push-sum EXTRA [ZY17; XK17]
  - Push-sum Gradient-tracking [NOS17]
  - Push-sum SGD [ALBR19]

Push-sum decentralized optimization

北京大学 PEKING UNIVERSITY

- Existing results show that algorithms over digraphs asymptotically converge as fast as centralized algorithms

Algorithm	Rate (A.)	Rate (F.T.)	Transient Stage
Gradient-Push [4]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.
Push-DIGing [15]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.

- How much slower decentralized optimization over digraphs are compared to centralized optimization?
  We do not know

- **It is not known how the digraphs will affect the convergence rate**

- **The transient iteration complexity is not known neither**

北京大学
PEKING UNIVERSITY

Part 05

Effective metrics to evaluate digraphs

Spectral gap

北京大学 PEKING UNIVERSITY

- In undirected graphs, the connectivity is gauged by spectral gap $1-\beta$ where

$$\beta = \|W - (1/n)\mathbf{1}_n\mathbf{1}_n^\top\|_2 \in [0,1)$$

- Inspired by this, can we use the same metric to capture the connectivity of digraphs? **No we cannot!**

$$\beta = \|W - (1/n)\mathbf{1}_n\mathbf{1}_n^\top\|_2 > 1 \text{ for digraphs}$$

- No problem. Recall that $W^k \to \pi\mathbf{1}_n^\top$, can we use the following metric? **No we cannot!**

$$\beta = \|W - \pi\mathbf{1}_n^\top\|_2 > 1 \text{ for digraphs}$$

Spectral gap

- According to [XSKK19], digraph connectivity can be gauged by **Generalized Spectral Gap** $1-\beta_\pi$

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

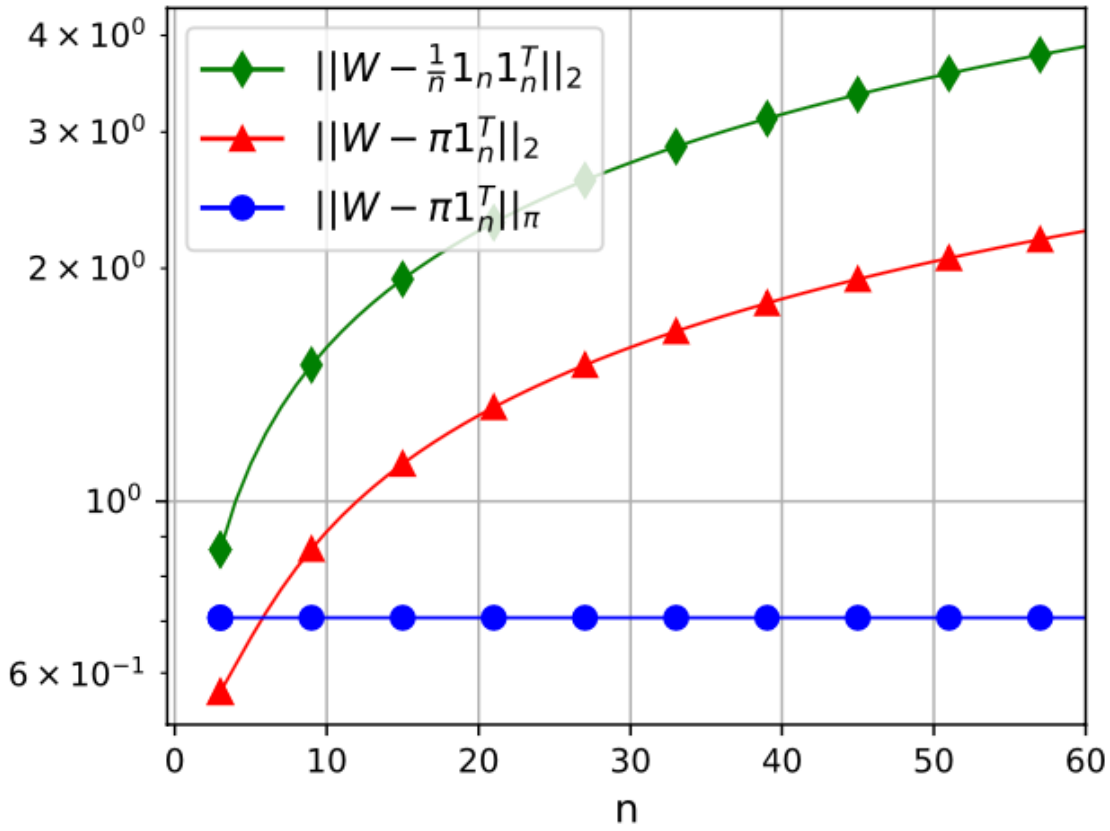
- For a vector v, its π-norm is defined as

$$\|v\|_\pi := \|\mathrm{diag}(\sqrt{\pi})^{-1} v\|_2$$

- For a vector A, its π-norm is defined as

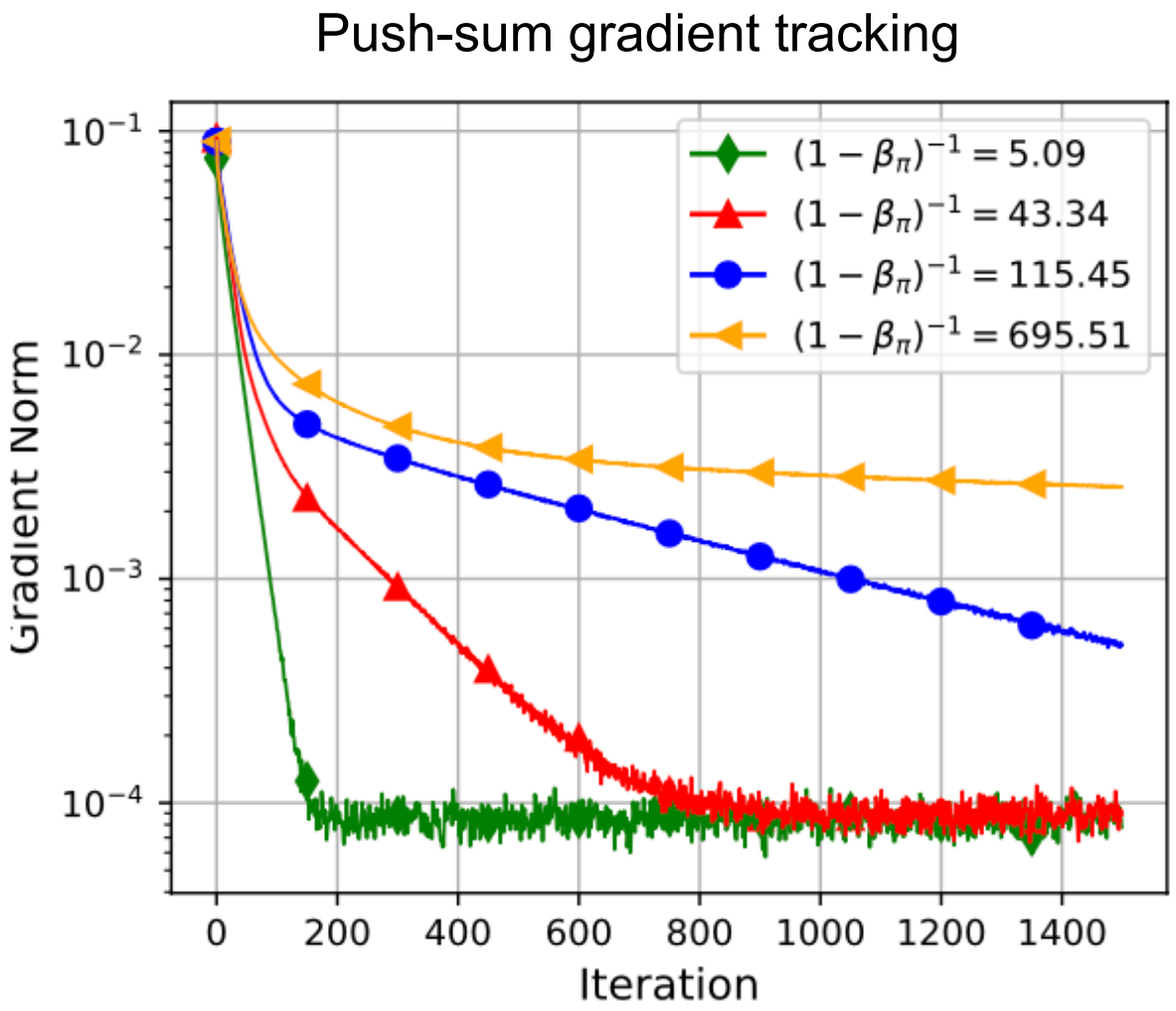
$$\|A\|_\pi := \|\mathrm{diag}(\sqrt{\pi})^{-1} A \ \mathrm{diag}(\sqrt{\pi})\|_2$$

- For undirected graphs, it holds that $\beta_\pi = \beta$



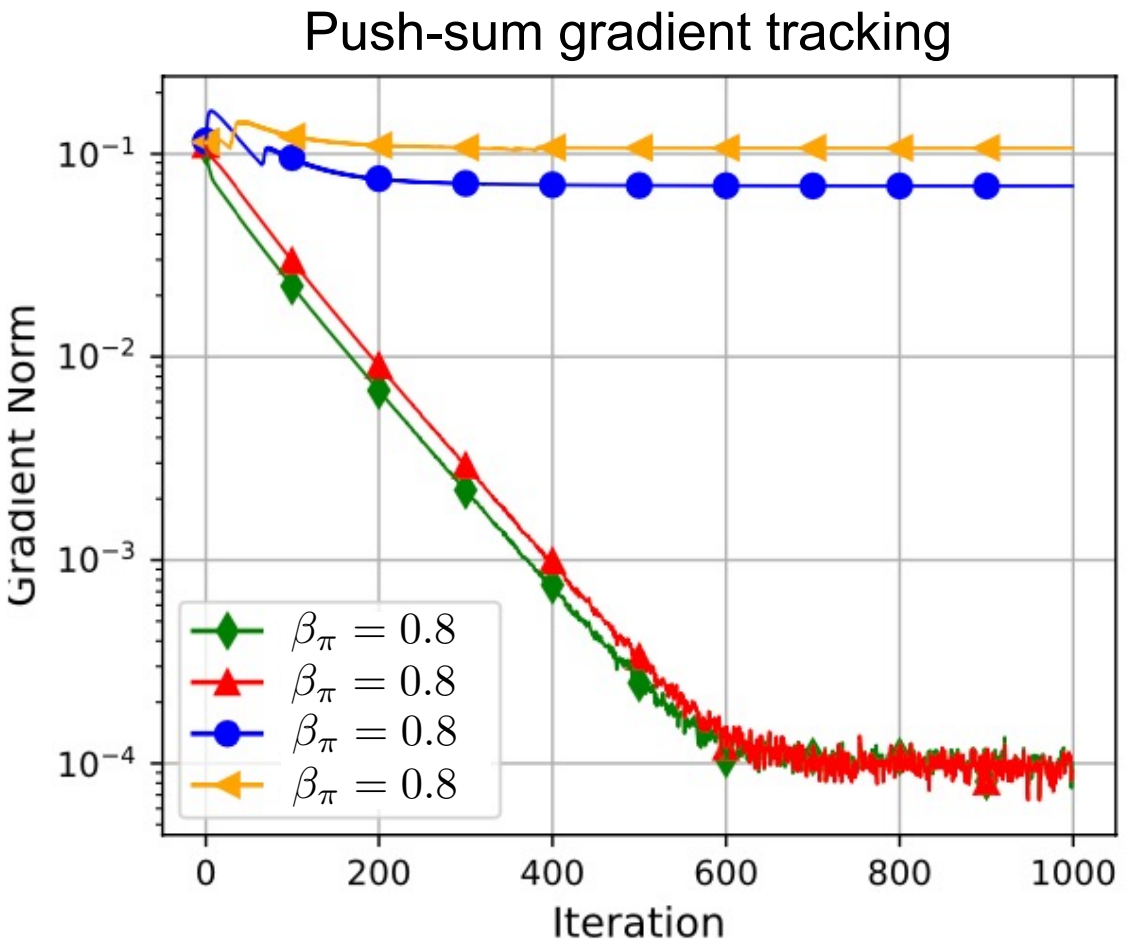
Spectral gap alone cannot precisely reflect the digraph influence

- We are done. We can use β_π to capture the influence of digraphs.



Spectral gap alone cannot precisely reflect the digraph influence

- Wait! Something strange is happening!
- Different curves vary drastically even with the same spectral gap!
- The single spectral gap alone is insufficient to capture the digraph influence



Equilibrium skewness

- Let's revisit the power iteration with column-stochastic W. Initialize $x^{(0)} = x$, we have

$$x^{(k)} = Wx^{(k-1)} = W^k x^{(0)} = W^k x \longrightarrow \pi \mathbf{1}_n^\top x \quad \text{as} \quad k \to \infty.$$

- To evaluate how fast $x^{(k)}$ converges to the global average $(1/n)\mathbf{1}_n\mathbf{1}_n^\top x$:
  - Generalized spectral gap to gauge how fast that $x^{(k)}$ approaches to $\pi \mathbf{1}_n^\top x$
  - A new metric to gauge the disagreement between $\pi \mathbf{1}_n^\top x$ and $(1/n)\mathbf{1}_n\mathbf{1}_n^\top x$

Equilibrium skewness

- Let's revisit the power iteration with column-stochastic W. Initialize $x^{(0)} = x$, we have

$$x^{(k)} = Wx^{(k-1)} = W^k x^{(0)} = W^k x \longrightarrow \pi \mathbf{1}_n^\top x \quad \text{as} \quad k \to \infty.$$

- To evaluate how fast $x^{(k)}$ converges to the global average $(1/n)\mathbf{1}_n\mathbf{1}_n^\top x$:
  - Generalized spectral gap to gauge how fast that $x^{(k)}$ approaches to $\pi \mathbf{1}_n^\top x$
  - **Equilibrium skewness** to gauge the disagreement between $\pi \mathbf{1}_n^\top x$ and $(1/n)\mathbf{1}_n\mathbf{1}_n^\top x$

$$\kappa_\pi := \max_i \pi_i / \min_i \pi_i \in [1, +\infty)$$

Revisit push-sum decentralized averaging

北京大学 PEKING UNIVERSITY

Push-sum averaging

$$\mathbf{z}^{(k+1)} = W\mathbf{z}^{(k)}$$
$$v^{(k+1)} = Wv^{(k)}$$
$$V^{(k+1)} = \mathrm{diag}(v^{(k+1)})$$
$$\mathbf{w}^{(k+1)} = {V^{(k+1)}}^{-1}\mathbf{z}^{(k+1)}.$$

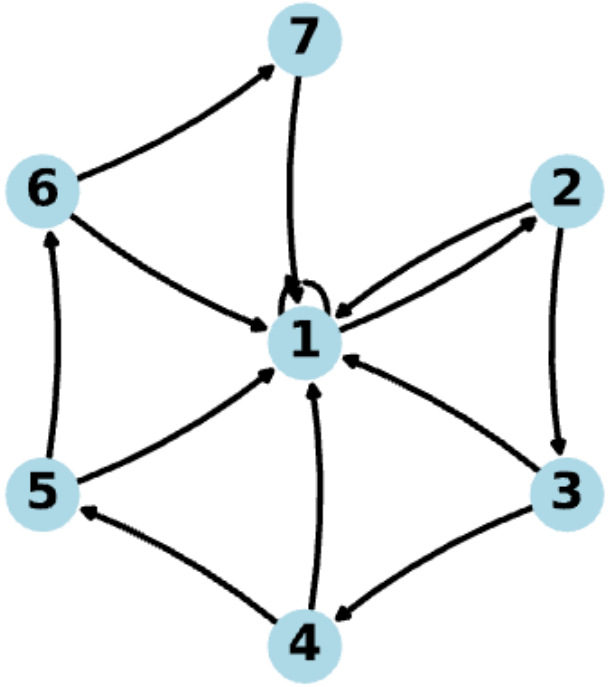
Theorem 1.

Assume $v^{(0)}$ is initialized such that $\mathbf{1}_n^\top v^{(0)} = n$, Push -sum decentralized averaging converges as follows

$$\|\mathbf{w}^{(k)} - \bar{\mathbf{z}}\|_F \leq \kappa_\pi^{3/2}\beta_\pi^k\|\mathbf{z}^{(0)}\|_F$$

The **first** rate reflecting the influence of both κ_π and β_π

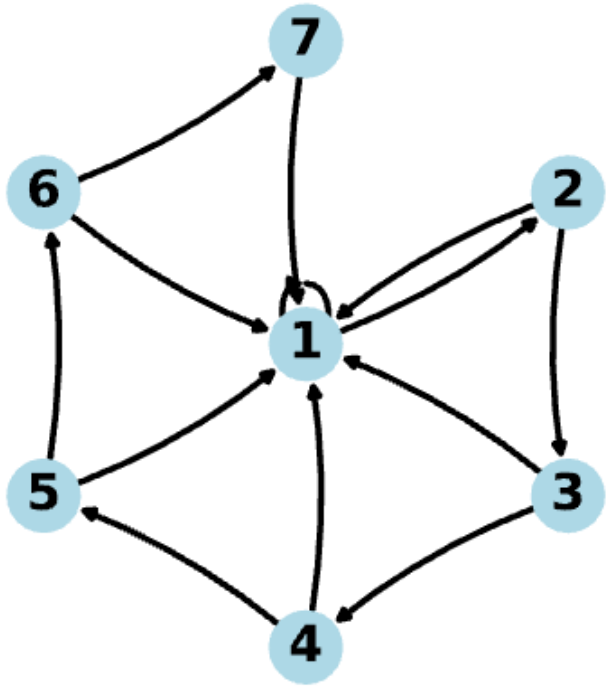
The influence of equilibrium skewness is substantial



$$W = \begin{bmatrix} 1/2 & 1/2 & \cdots & 1/2 & 1 \\ 1/2 & 0 & & & \\ & \ddots & \ddots & & \\ & & 1/2 & 0 & \\ & & & 1/2 & 0 \end{bmatrix} \in \mathbb{R}^{n \times n}.$$

The probability that a message flows from node 1 to node n decays **exponentially** fast

The influence of equilibrium skewness is substantial

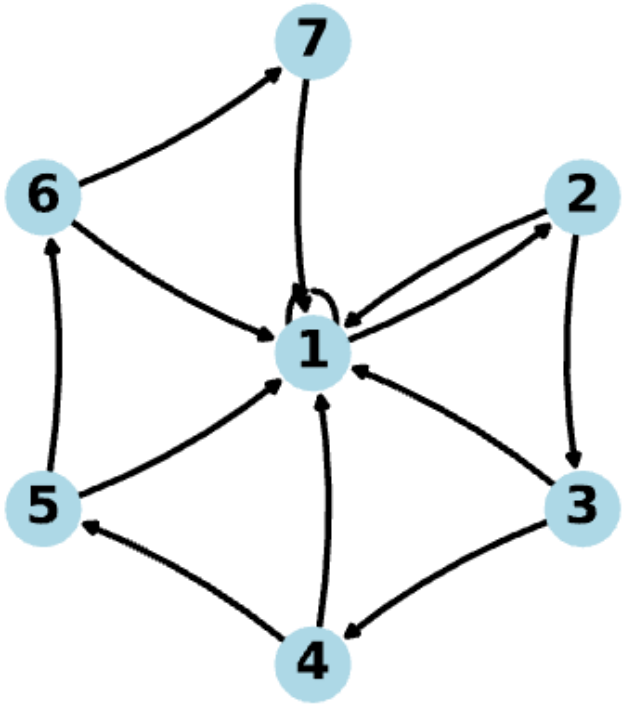


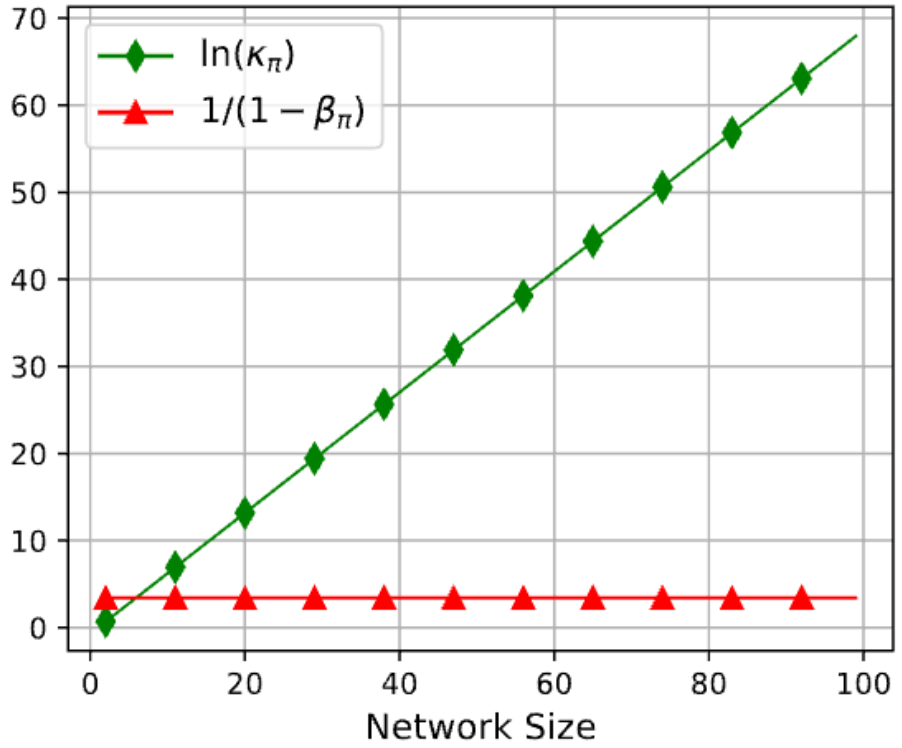
Theorem 2.

For any $n \geq 1$, there exists a matrix $W \in \mathbb{R}^{n \times n}$ such that

$$\beta_\pi = \frac{\sqrt{2}}{2} \quad \text{and} \quad \kappa_\pi = 2^{n-1}$$

The influence of equilibrium skewness is substantial



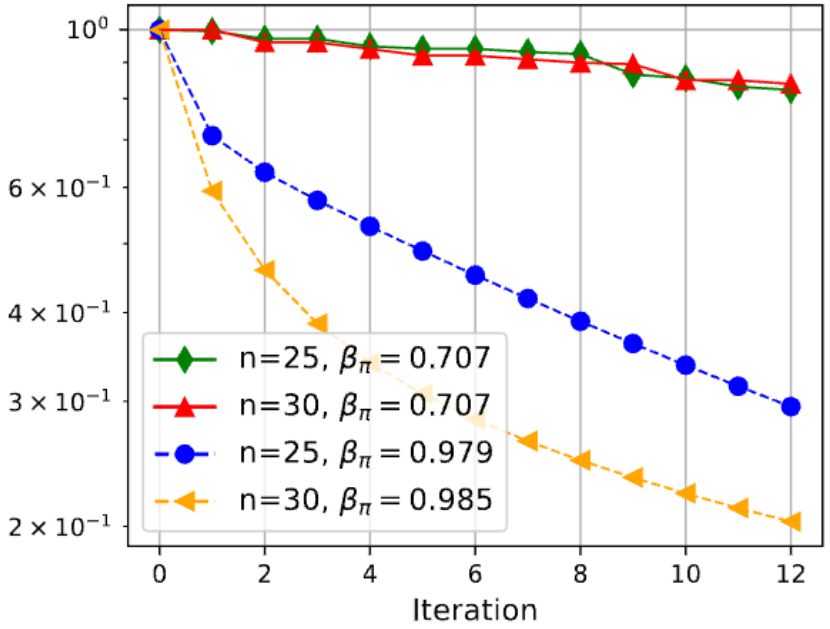


- β_π and κ_π is orthogonal
- The influence of κ_π can be highly non-trivial !

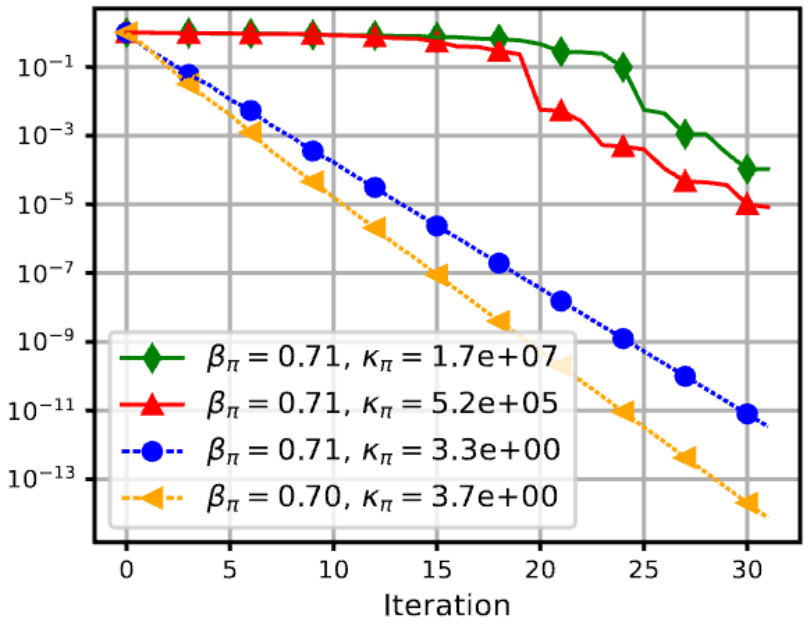
Push-sum decentralized averaging: simulation

- Metrics β_π and κ_π together precisely reflects the influence of network on push-sum averaging

$\kappa_\pi = \text{const. and } \beta_\pi \to 1$



$\beta_\pi = \text{const. and } \kappa_\pi \to \infty$



- Both metrics are indispensible

Push-sum gradient tracking

- Recall the distributed stochastic optimization problem

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

- Given a column-stochastic matrix W, the push-sum gradient tracking [NOS17] is

$$\begin{aligned}
\mathbf{x}^{(k+1)} &= W(\mathbf{x}^{(k)}-\gamma\mathbf{y}^{(k)})\\
v^{(k+1)} &= Wv^{(k)}\\
V^{(k+1)} &= \mathrm{diag}(v^{(k+1)})\\
\mathbf{w}^{(k+1)} &= {V^{(k+1)}}^{-1}\mathbf{x}^{(k+1)}\\
\mathbf{y}^{(k+1)} &= W(\mathbf{y}^{(k)}+\nabla F(\mathbf{w}^{(k+1)};\boldsymbol{\xi}^{(k+1)})-\nabla F(\mathbf{w}^{(k)};\boldsymbol{\xi}^{(k)}))
\end{aligned}$$

Push-sum gradient tracking: convergence

北京大学 PEKING UNIVERSITY

Theorem 3

Suppose each $f_i(x)$ is L-smooth, each local stochastic gradient $\nabla F(x;\xi_i)$ is unbiased and has bounded variance. Given a column-stochastic weight matrix W, the convergence rate of Push-DIGing satisfies

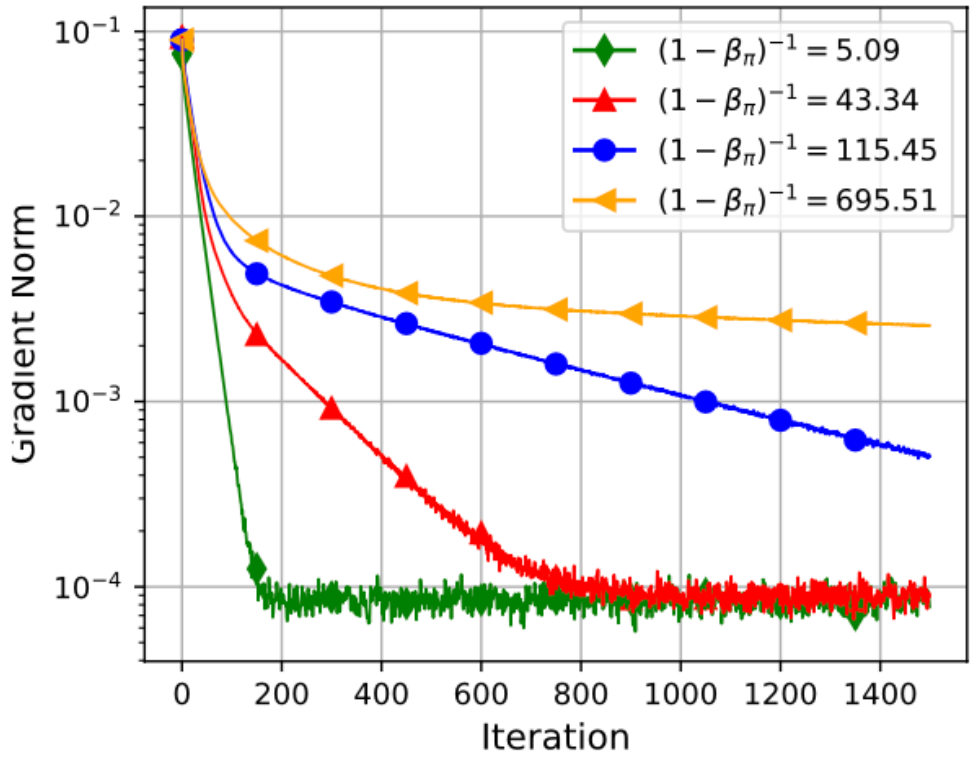
$$\frac{1}{K}\sum_{i=0}^{K-1}\mathbb{E}[\|\nabla f(\bar{x}^{(k)}\|_2^2] \lesssim \frac{\sigma}{\sqrt{nK}} + \frac{\beta_\pi^{\frac{2}{3}}\kappa_\pi^{\frac{5}{3}}\sigma^{\frac{2}{3}}}{(1-\beta_\pi)K^{\frac{2}{3}}} + \frac{\beta_\pi\kappa_\pi^3(1+\kappa_\pi\beta_\pi)}{(1-\beta_\pi)^2K} + \frac{1}{K}.$$

The **first** rate that clarifies the influence of digraphs on decentralized algorithms

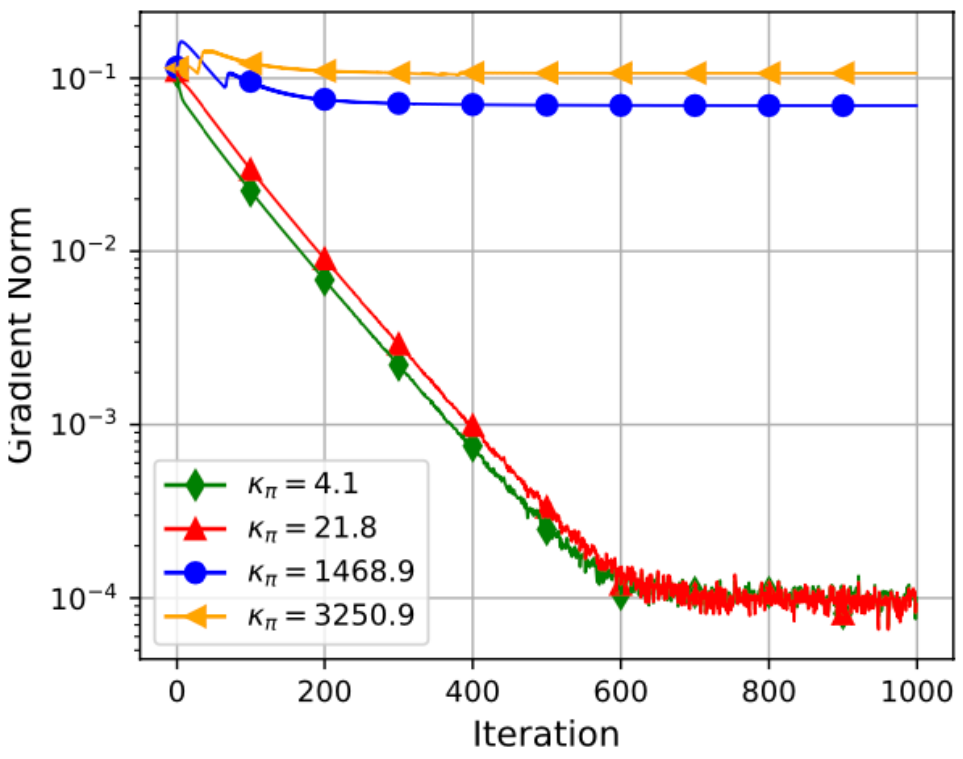
Push-sum gradient tracking: simulation

- Logistic regression with non-convex regularizer

$\kappa_\pi = \text{const. and } \beta_\pi \to 1$



$\beta_\pi = \text{const. and } \kappa_\pi \to \infty$



Push-sum gradient tracking: comparison with existing works

Algorithm	Rate (A.)	Rate (F.T.)	Transient Stage
Gradient-Push [4]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.
Push-DIGing [15]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.
Push-DIGing (Ours)	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{\beta_\pi \kappa_\pi^3 (1+\kappa_\pi \beta_\pi)}{(1-\beta_\pi)^2 K}$	$\frac{n\kappa_\pi^8}{(1-\beta_\pi)^4}$

The influence of κ_π is substantial, especially when $\kappa_\pi = O(2^n)$

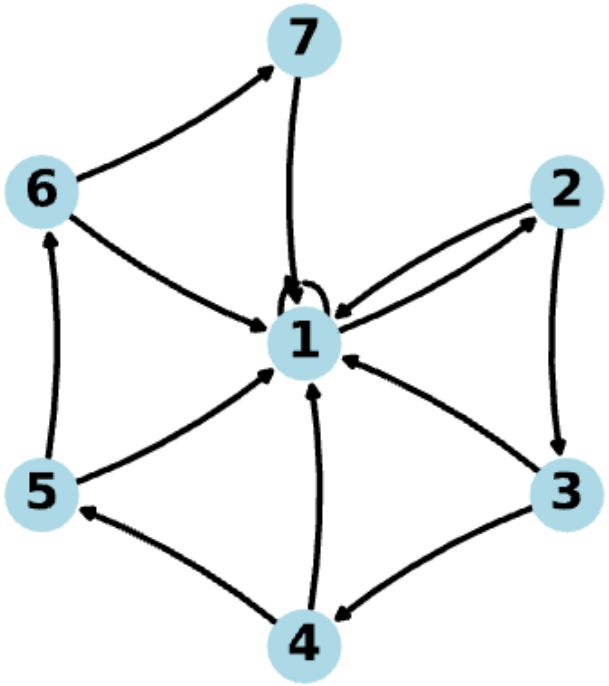
What is the **best-possible dependence** on κ_π and β_π ?

北京大学
PEKING UNIVERSITY

Part 05

Lower bound over digraphs

A challenging digraph



For this graph, we prove that

$$\pi \propto \left(2^{n-1}, 2^{n-2}, \ldots, 1\right)^\top \text{ and } \kappa_\pi = 2^{n-1}$$

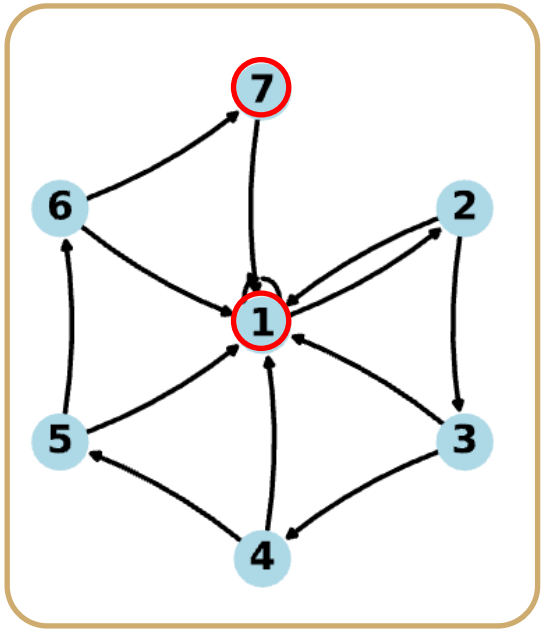
Also, the spectral gap is constant

$$\beta_\pi = 1/\sqrt{2} \quad \forall n$$

Lower bound of communication rounds: core idea

- For non-convex centralized deterministic optimization, the lower bound rate is $\Omega(L\Delta/K)$

- For the digraph below, each effective message passing requires n hops.



require n rounds per iteration

Communication lower bound

$$\Omega(\frac{nL\Delta}{K})$$

$$n = \frac{1+\ln(\kappa_\pi)}{1-\beta_\pi}$$

Communication lower bound

$$\Omega(\frac{(1+\ln(\kappa_\pi))L\Delta}{1-\beta_\pi})$$

Convergence lower bound over digraphs

北京大学 PEKING UNIVERSITY

Theorem 4

For any given $L \geq 0$, $n \geq 2$, $\sigma \geq 0$, and $\tilde{\beta} \in [1/\sqrt{2}, 1-1/n]$, there exists a set of loss functions $\{f_i\}_{i=1}^n \in \mathcal{F}_{\Delta,L}$, a set of stochastic gradient oracles in $\mathcal{O}_{\sigma^2}$, and a column-stochastic matrix $W \in \mathbb{R}^{n\times n}$ with $\beta_\pi = \tilde{\beta}$ and $\ln(\kappa_\pi) = \Omega(n(1-\beta_\pi))$, such that the convergence of any $A \in \mathcal{A}_W$ starting form $x_i^{(0)} = x^{(0)}, \forall\, 1 \leq i \leq n$ with K iterations is lower bounded by

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

Big gap between lower and upper bound

北京大学 PEKING UNIVERSITY

Algorithm	Rate (A.)	Rate (F.T.)	Transient Stage
Gradient-Push [4]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.
Push-DIGing [15]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.
Push-DIGing (Ours)	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{\beta_\pi \kappa_\pi^3 (1+\kappa_\pi \beta_\pi)}{(1-\beta_\pi)^2 K}$	$\frac{n\kappa_\pi^8}{(1-\beta_\pi)^4}$
Lower Bound (Ours)	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{(1+\ln(\kappa_\pi))L\Delta}{(1-\beta_\pi)K}$	$\frac{n(1+\ln(\kappa_\pi))^2}{(1-\beta_\pi)^2}$

北京大学
PEKING UNIVERSITY

Part 06

Optimal decentralized algorithm over digraphs

Recall push-sum gradient tracking

$$
\begin{aligned}
\mathbf{x}^{(k+1)} &= W(\mathbf{x}^{(k)} - \gamma \mathbf{y}^{(k)}) \\
v^{(k+1)} &= W v^{(k)} \\
V^{(k+1)} &= \mathrm{diag}(v^{(k+1)}) \\
\mathbf{w}^{(k+1)} &= {V^{(k+1)}}^{-1} \mathbf{x}^{(k+1)} \\
\mathbf{y}^{(k+1)} &= W(\mathbf{y}^{(k)} + \nabla F(\mathbf{w}^{(k+1)}; \boldsymbol{\xi}^{(k+1)}) - \nabla F(\mathbf{w}^{(k)}; \boldsymbol{\xi}^{(k)}))
\end{aligned}
$$

Multiple-Gossip Push-sum Gradient tracking

Improve it with **multiple-gossip** and **mini-batch gradient**

$$
\begin{aligned}
\mathbf{x}^{(k+1)} &= W^R(\mathbf{x}^{(k)} - \gamma \mathbf{y}^{(k)}) \\
v^{(k+1)} &= W^R v^{(k)} \\
V^{(k+1)} &= \mathrm{diag}(v^{(k+1)}) \\
\mathbf{w}^{(k+1)} &= {V^{(k+1)}}^{-1} \mathbf{x}^{(k+1)} \\
\mathbf{y}^{(k+1)} &= W^R(\mathbf{y}^{(k)} + g_R^{(k+1)} - g_R^{(k)})
\end{aligned}
$$

where $g_R^{(k)} = \frac{1}{R}\sum_{r=1}^{R} \nabla F(\mathbf{w}^{(k)}; \boldsymbol{\xi}^{(k,r)})$ is the mini-batch stochastic gradient

Multiple-Gossip Push-sum Gradient tracking

北京大学 PEKING UNIVERSITY

Theorem 5

Suppose each $f_i(x)$ is L-smooth, each local stochastic gradient $\nabla F(x;\xi_i)$ is unbiased and has bounded variance, and the weight matrix W is column-stochastic, by setting $R = \frac{(1+\sqrt{\ln(\kappa_\pi)})^2}{1-\beta_\pi}$, $T = KR$, the convergence of MG-Push-DIGing satisfies

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}[\|\nabla f(\bar{x}^{(k)})\|_2^2] = \widetilde{\mathcal{O}}\left(\frac{\sigma\sqrt{L\Delta}}{\sqrt{nT}} + \frac{(1+\ln(\kappa_\pi))L\Delta}{(1-\beta_\pi)T}\right),$$

where $\widetilde{\mathcal{O}}(\cdot)$ absorbs logarithmic factors independent of κ_π and β_π.

Lower bound and upper bound are nearly-matched

Algorithm	Rate (A.)	Rate (F.T.)	Transient Stage
Gradient-Push [4]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.
Push-DIGing [15]	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	N.A.	N.A.
Push-DIGing (Ours)	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{\beta_\pi \kappa_\pi^3 (1+\kappa_\pi \beta_\pi)}{(1-\beta_\pi)^2 K}$	$\frac{n\kappa_\pi^8}{(1-\beta_\pi)^4}$
MG-Push-DIGing (Ours)	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{(1+\ln(\kappa_\pi))L\Delta}{(1-\beta_\pi)K}$	$\frac{n(1+\ln(\kappa_\pi))^2}{(1-\beta_\pi)^2}$
Lower Bound (Ours)	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}}$	$\frac{\sigma\sqrt{L\Delta}}{\sqrt{nK}} + \frac{(1+\ln(\kappa_\pi))L\Delta}{(1-\beta_\pi)K}$	$\frac{n(1+\ln(\kappa_\pi))^2}{(1-\beta_\pi)^2}$

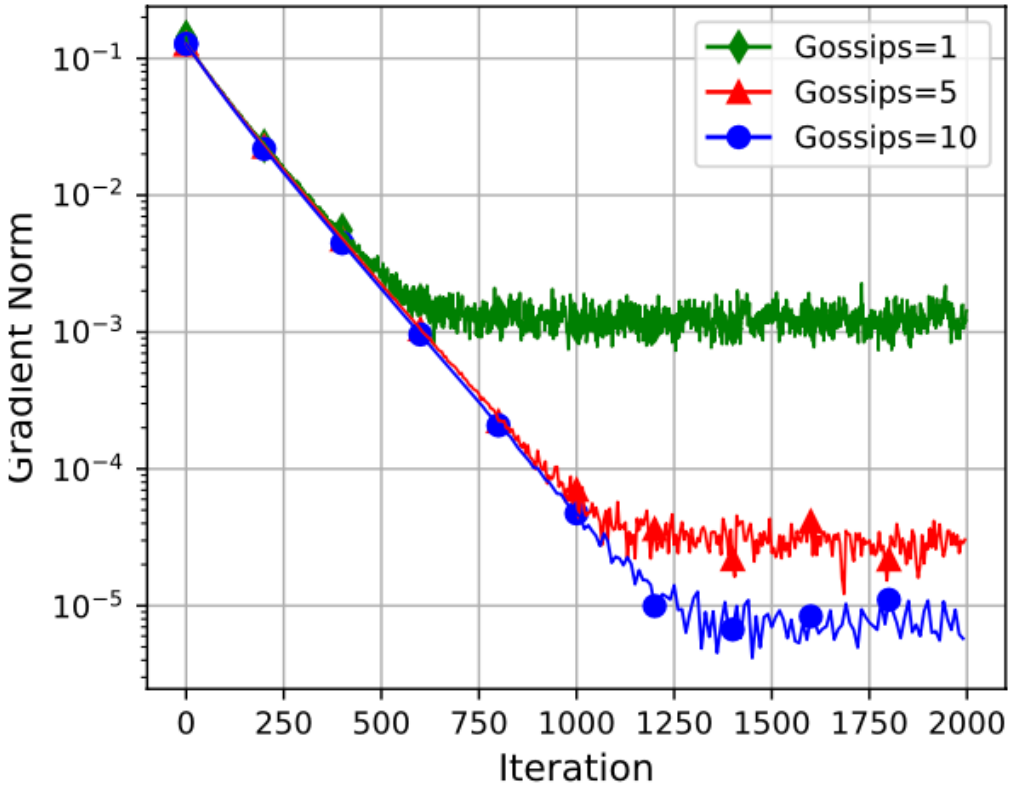
Our lower bound is nearly-tight

Our developed MG-Push-DIGing algorithm is nearly optimal

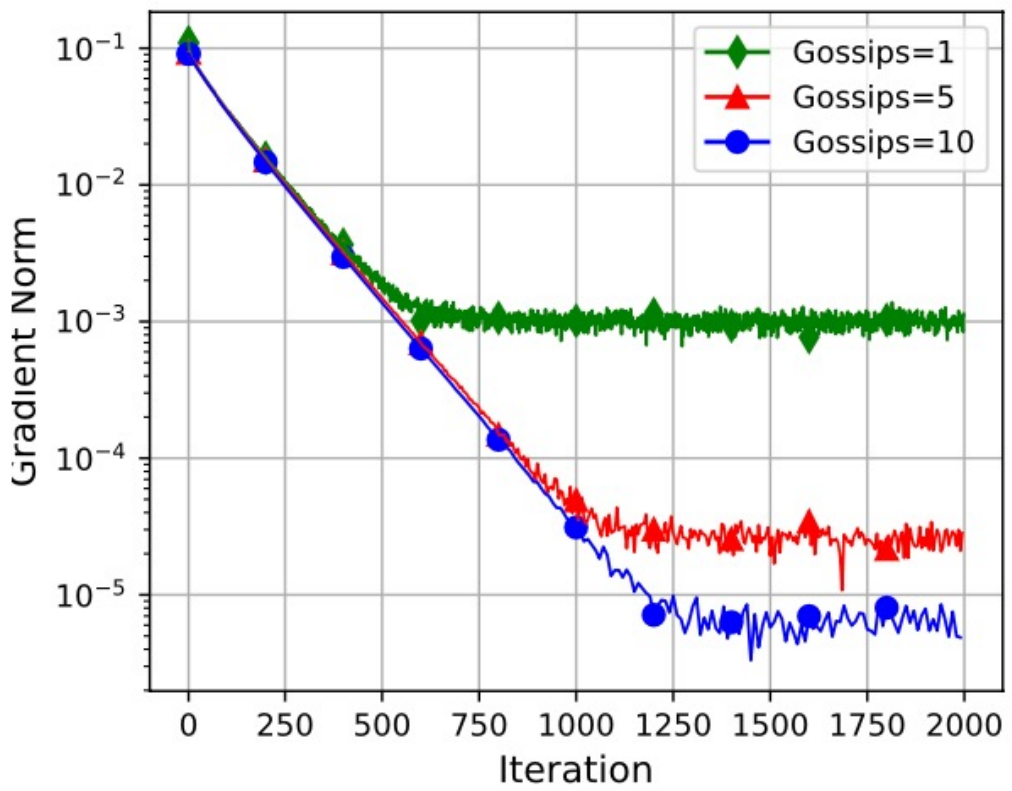
Simulations

- Logistic regression with non-convex regularizer. Multiple Gossip improves both κ_π and $(1-\beta_\pi)^{-1}$

$\kappa_\pi = 4804.49,\ (1-\beta_\pi)^{-1} = 2.34.$



$\kappa_\pi = 6.33,\ (1-\beta_\pi)^{-1} = 51.24.$



Summary

北京大学 PEKING UNIVERSITY

- The influence of diagraphs on decentralized algorithms is known in previous work
- We identify two effective metrics (spectral gap and equilibrium skewness) that can jointly capture the influence of digraphs
- The two metrics are orthogonal to each other; both of them are indispensable
- We establish the lower bound and develop a nearly-optimal algorithm to attain the lower bound

Future work

北京大学 PEKING UNIVERSITY

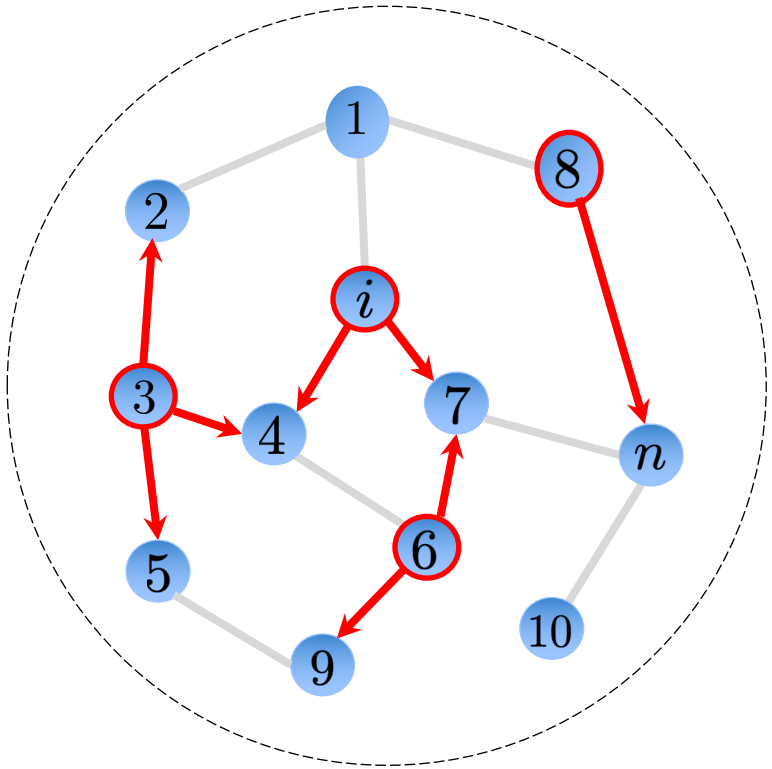
- Clarify the influence of row-stochastic digraphs on decentralized algorithms
- Clarify the influence of digraphs over push-pull algorithms
- Lower bound and optimal algorithms for pull-sum and push-sum family

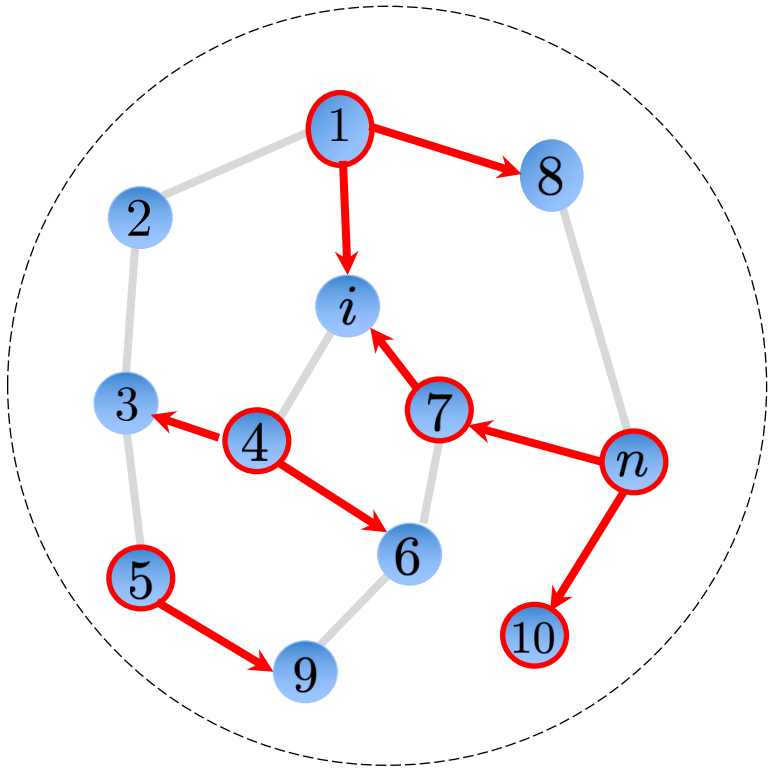
北京大学
PEKING UNIVERSITY

Thank you!

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

BlueFog homepage: https://github.com/Bluefog-Lib/bluefog


1
2
8
i
3
4
7
n
6
5
10
9


1
2
8
i
3
4
7
n
6
5
10
9