Joint Consensus Matrix Design and Resource Allocation for Decentralized Learning

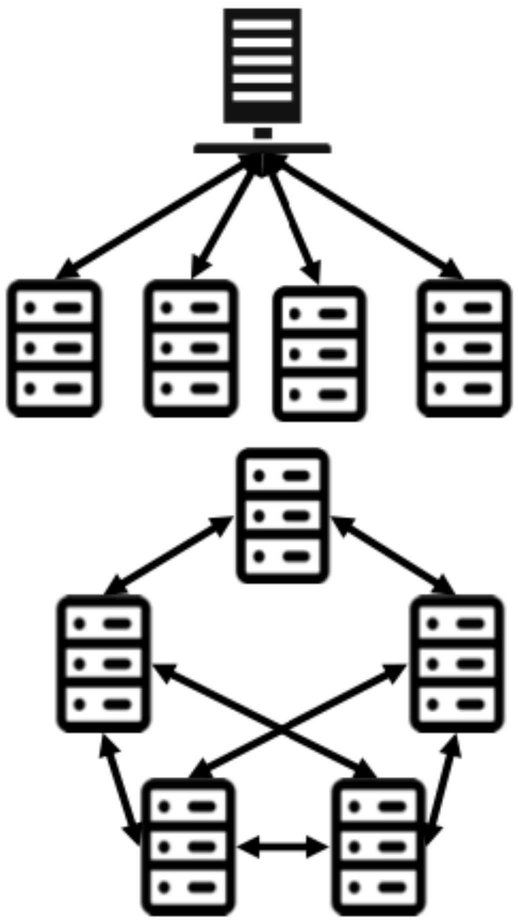
Jingrong Wang*, Ben Liang*,

Zhongwen Zhu†, Emmanuel Thepie Fapi†, Hardik Dalal†

*Department of Electrical and Computer Engineering, University of Toronto, Canada
†Ericsson Global AI Accelerator Montréal, Canada

Distributed Machine Learning (ML)

- Shift from a centralized fashion to decentralized ml
  - Alleviate the problem of computation and communication bottleneck at a central parameter server.
- In each training iteration
  - Each worker takes a weighted average of the models that are aggregated from its neighbors.
- The training performance is affected by
  - How the model information is exchanged among neighboring workers.



Related Work

- The convergence speed is governed by $\rho(W) = ||W - \frac{\mathbf{1}\mathbf{1}^\top}{N}||_2$.
  - The second-largest singular value of the consensus weight matrix.
- Optimal consensus weight matrix:
  - Fastest distributed linear averaging (FDLA) [4].
- Sparse communication graph:
  - Standard sparse network topologies, e.g., a ring [15] – [19].
  - Maximize the convergence rate s.t. some prescribed communication cost [20]-[25].
  - Minimize the communication cost s.t. a prescribed convergence rate [4], [26].
- This work:
  - Total wall-clock training time.
  - Efficient communication resource allocation.

System Model

- Latency in each training iteration

dominated by the stragglers

$$g(W,B) = \max_{i,j\in\mathcal{N}} \left\{ L_{i,j}(B_{i,j}) \mathbb{1}_{\{W_{i,j}\neq 0\}} \right\}$$

bandwidth allocation

whether there exists information exchange

latency corresponding to the link from worker i to j

- #of training iterations for a desired error ϵ is [37]

$$T_\epsilon \in \mathcal{O}\left(\frac{1}{\epsilon^2(1-\rho(W))}\right)$$

- Joint consensus matrix design and communication resource allocation

$$\min_{W,B} \quad \frac{1}{1-\rho(W)} g(W,B),$$

$$\text{s.t.} \quad \sum_{i,j\in\mathcal{N}} B_{i,j} \leq \bar{B},$$

Resource constraints

$$B_{i,j} \geq 0, \forall i,j \in \mathcal{N},$$

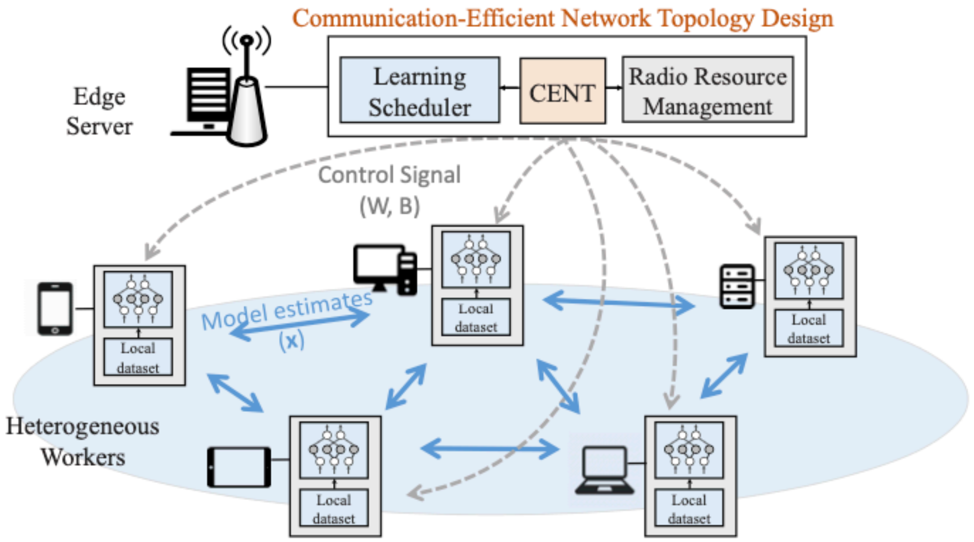
$$\rho(W) < 1,$$

ML convergence requirement

$$W\mathbf{1} = \mathbf{1},$$

$$W = W^\top,$$

$$W \in S_A,$$



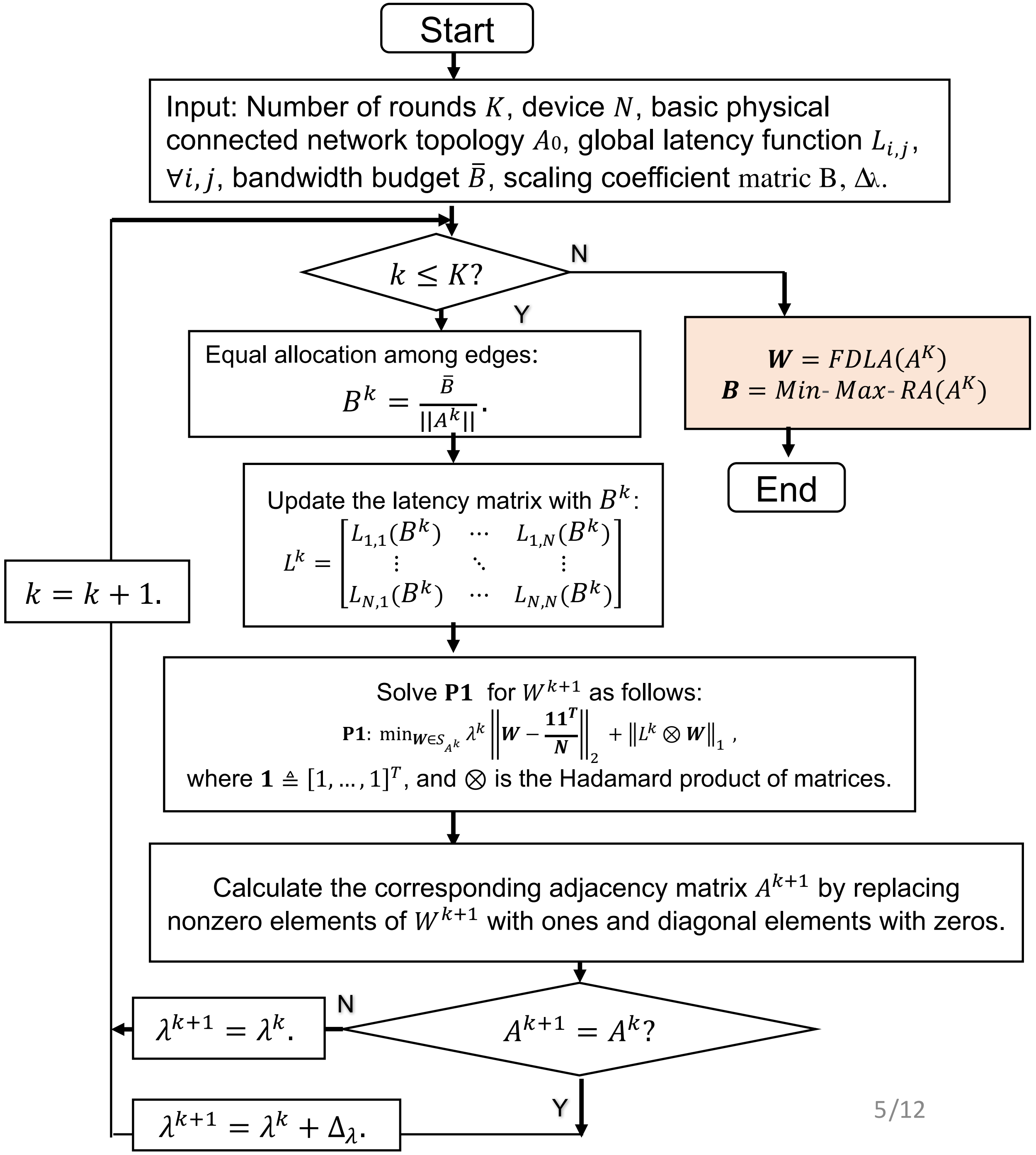
Challenges and Design Highlights

- Challenges
  - W and B are coupled and restricted by the physical network topology.
  - Non-convex and non-smooth due to the existence of the indicator function.
  - Exhaustive search is computationally expensive due to vast search space, 2^{N^2}.
  - Existing solutions for multivariable non-convex function are not applicable.
    - **Lemma**: Coordinate descent method becomes stuck after two iterations.
- Motivation:
  - Preserves the training convergence rate.
  - Reduces latency by enforcing communication graph sparsity and avoiding selecting poor communication links.

Communication-Efficient Network Topology

Design highlight 1:

- Given the optimal sparse topology, W and B can be optimized independently.



Communication-Efficient Network Topology

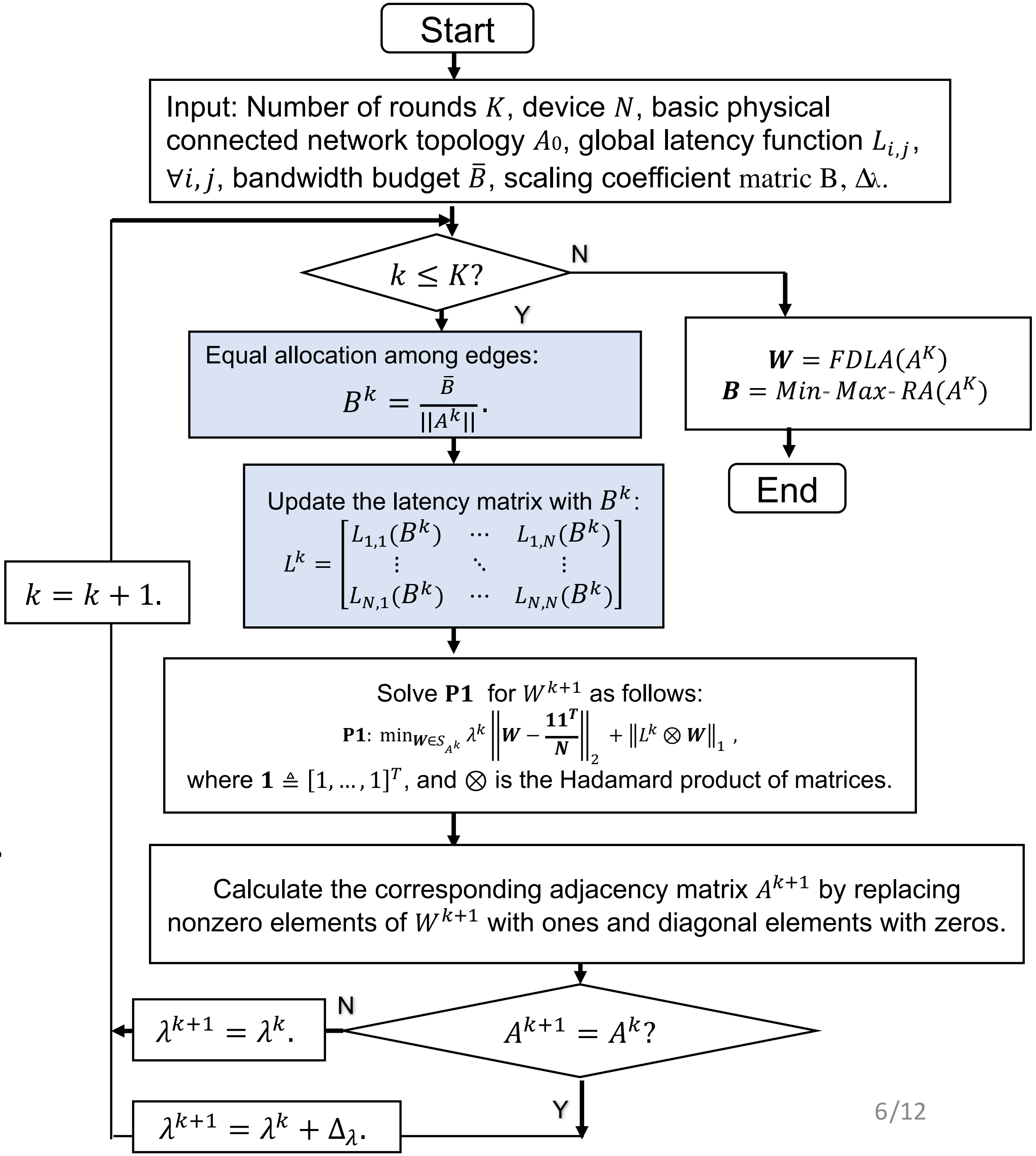
Design highlight 2:

- We use equal bandwidth allocation in the intermediate step to capture the inherent goodness of the links.

Rationale: If optimal resource allocation

-> results in equal latency.

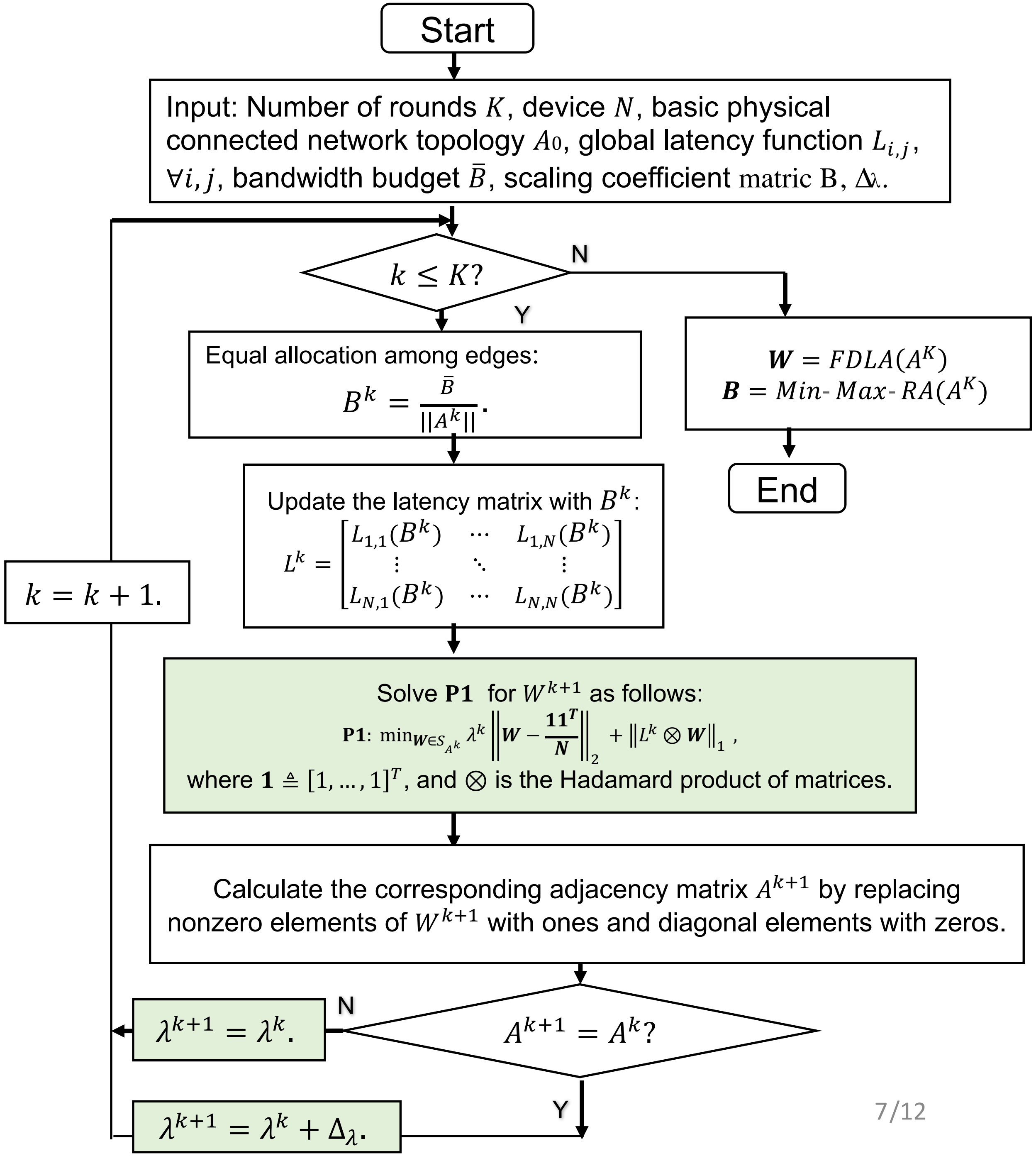
-> Defeats the purpose of differentiating links.



Communication-Efficient Network Topology

Design highlight 3:

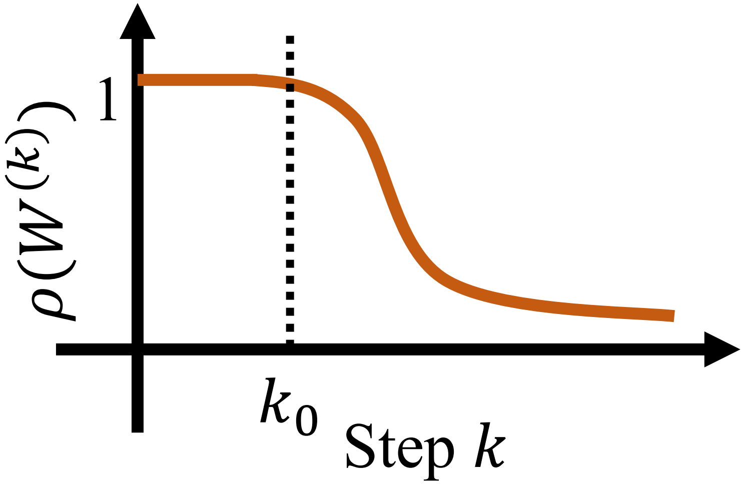
- We iteratively design a trade-off factor to efficiently to balance the convergence rate and the sparsity of the consensus weight matrix.
- We solve a convex problem in the intermediate step.



Theoretical Analysis

Theorem 6. *CENT converges as k approaches infinity. Furthermore, the objective $\frac{1}{1-\rho(W^{(k)})} g(W^{(k)}, B^{(k)})$ is non-increasing in k for $k > k_0$.*

- $\{\rho(W^{(k)})\}_{k>0}$ is non-increasing and bounded below.



Theorem 7. *If $K > k_0$, decentralized ML converges.*

- $\rho(\widehat{W}) \leq \rho(W^{(K)}) < 1$.
- Communication latency in each training iteration is finite.

Evaluation

- MNIST + LeNet.
- 50 workers uniformly randomly distributed in a 100 m X 100 m area.
- Each realization has 200 edges [4].
- Benchmarks:
  - FDLA[4]: fastest convergence rate in terms of the number of training iterations.
  - Max-degree [5]: maximum degree of the graph.
  - Metropolis [6]: maximum degree of its two adjacent workers.
  - Best-constant [7]: the eigenvalues of the Laplacian matrix of the graph.

Convergence Factor $\rho(W)$

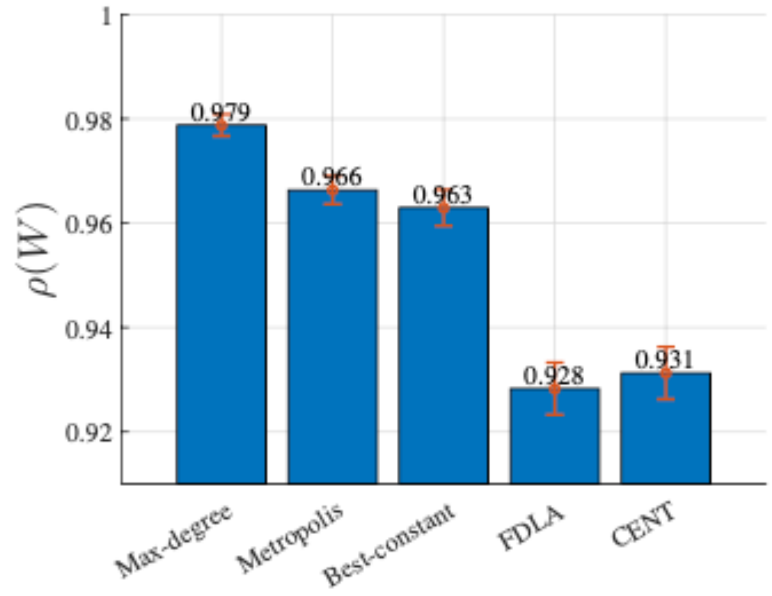


Fig. 4. Convergence factor $\rho(W)$.

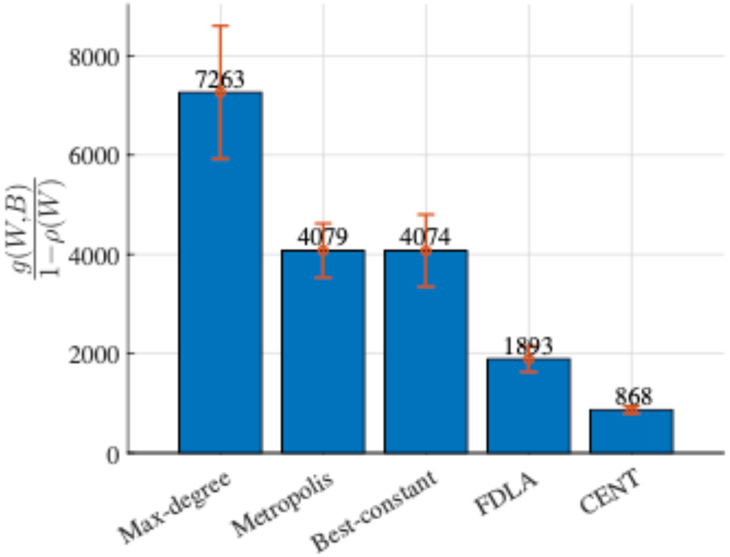


Fig. 5. Training time objective $\frac{g(W,B)}{1-\rho(W)}$.

- CENT requires significantly shorter wall-clock training time than the other methods, while retaining $\rho(W)$ as FDLA.

Training/Test Accuracy and Network Scale

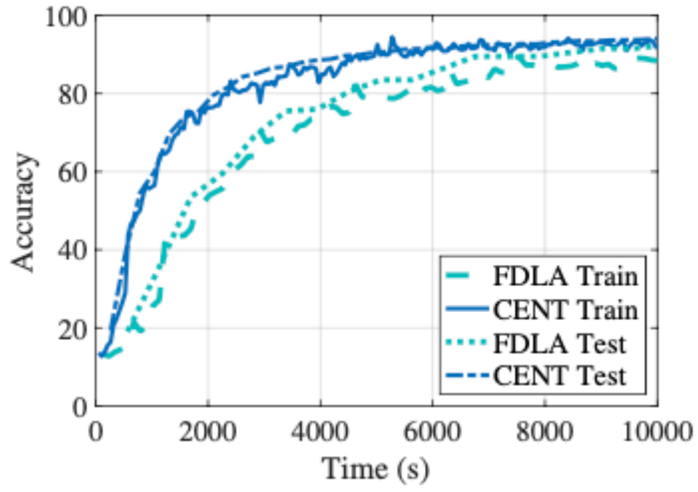


Fig. 6. Accuracy vs. wall-clock time.

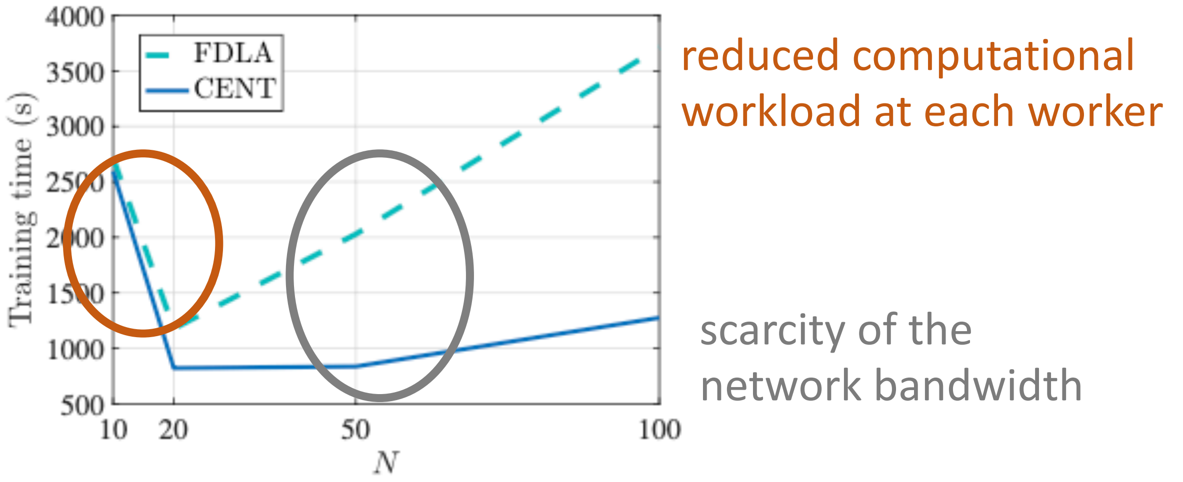


Fig. 7. Wall-clock training time over N.

- CENT requires less time achieving the same level of training accuracy.
- CENT excels in robustness
  - With efficient sparse graph design and bandwidth allocation.

Takeaways

- Formulated the problem of joint consensus weight matrix design and communication resource allocation in decentralized ML
  - The wall-clock training time:
    - Latency in each training iteration + number of iterations needed to reach convergence.
- Proposed CENT:
  - Iteratively enforces graph sparsity while retaining the convergence rate.
- Analyzed
  - The convergence of CENT.
  - The convergence of decentralized ML while applying the output of cent.
- Experiments: significantly faster wall-clock training time.