HierTopo: Towards High-Performance and Efficient Topology Optimization for Dynamic Networks

Jing Chen, Zili Meng, Yaning Guo, Mingwei Xu, Hongxin Hu

Tsinghua University

University at Buffalo
The State University of New York

Dynamic networks
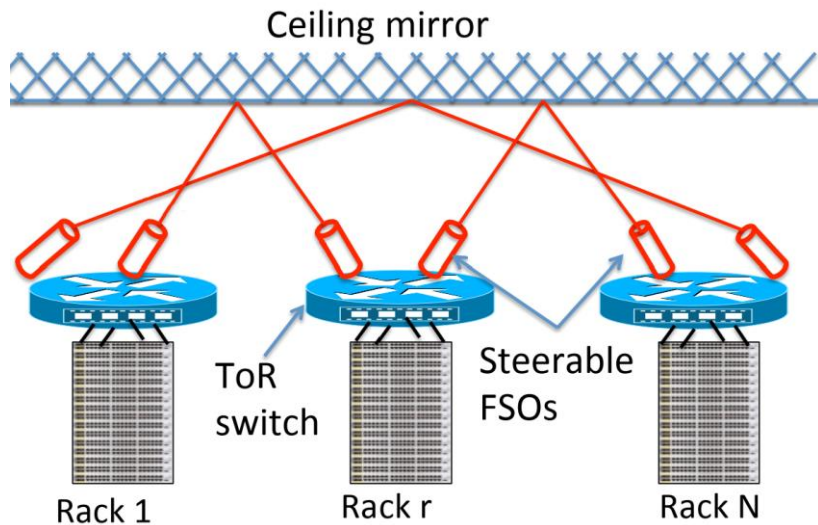
Reconfigurable Datacenter
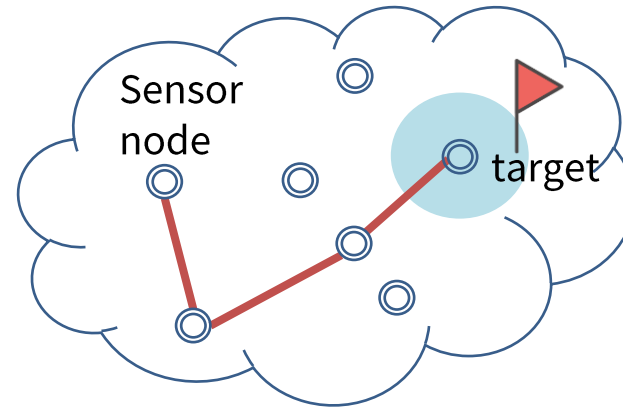
Wireless Sensor Network

Satellite Network

*Figures are partially borrowed from "FireFly: A Reconfigurable Wireless Data Center Fabric Using Free-Space Optics" in ACM SIGCOMM 2014 and "Network topology design at 27,000 km/hour" in ACM CoNext 2019.

Dynamic network topology design
adapt to real-time traffic (efficient)
achieve shorter path length (high-performance)

Network Topology Design

Sophisticated modelling

Long decision latency

- Wiro [Sigmetrics'19] takes minutes to solve ILP

Lacking generalization ability

- xWeaver [Sigmetrics'18] requires its NN model be frequently retrained
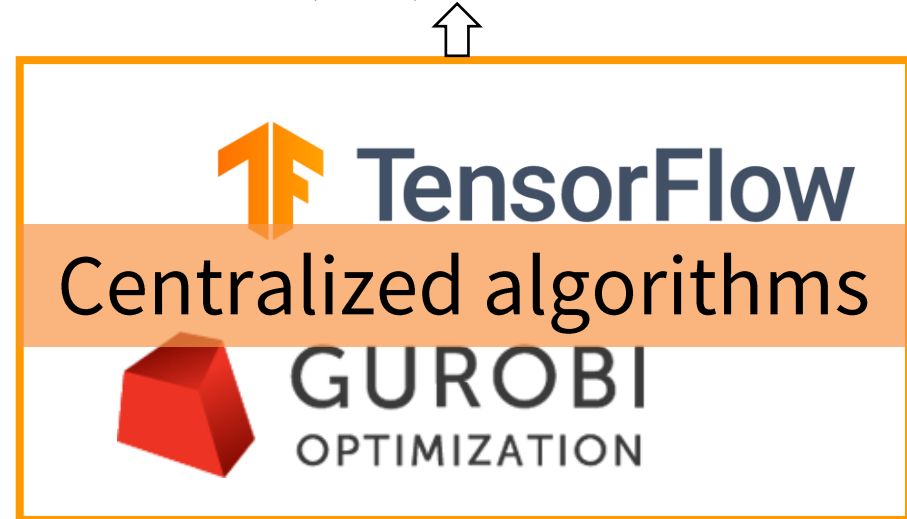
Graph theory-based heuristics

Sub-optimal performance

- Greedily matching suffers >15% optimality gap within 4 steps

- Tree-based algorithm degrades the performance (50 node) by >30% against xWeaver
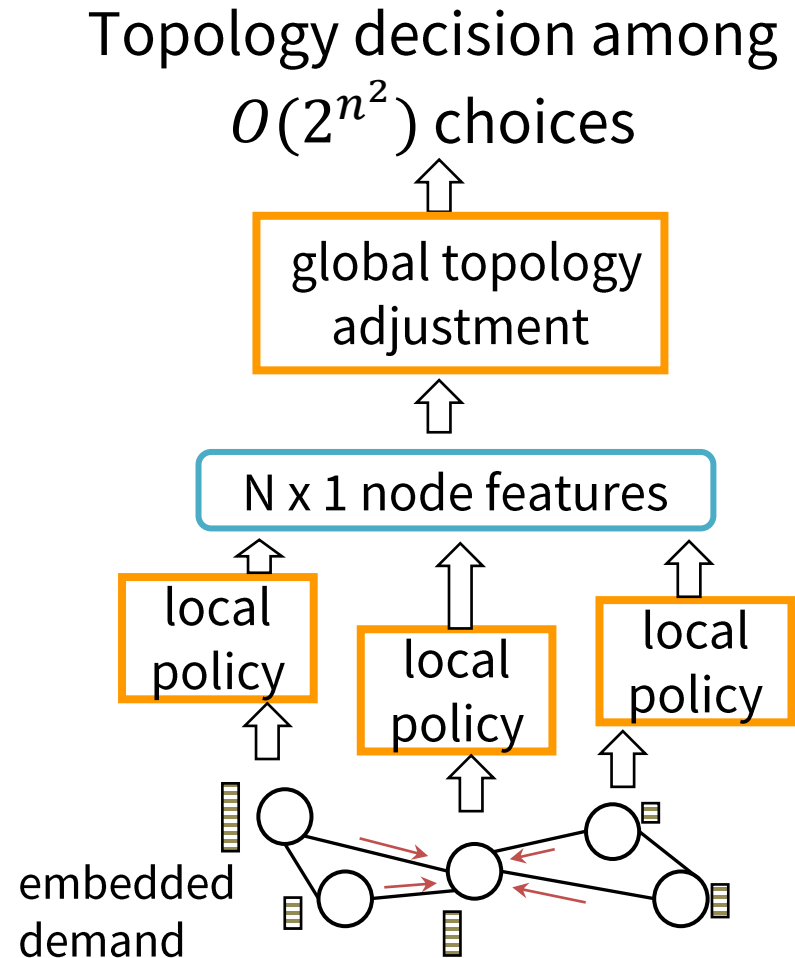
Root cause

- Topology optimization is complicated!

- complex inputs

- huge action space

- NP-hard [Sigmetrics'18]

- Centralized algorithms are inevitably complicated

Topology decision among $O(2^{n^2})$ choices

⇧

Centralized algorithms

⇧

current topology   N x N demand

$$\begin{bmatrix} 0 & \cdots & 4.2 \\ \vdots & \ddots & \vdots \\ 1.1 & \cdots & 0 \end{bmatrix}$$

Our Contribution: HierTopo

- A *hierarchical* way

- Offload the aggregation of network information to each node

- The global topology adjustment algorithm could be kept simple
  - global algorithm details please refer to our paper

Topology decision among $O(2^{n^2})$ choices

⇧

global topology adjustment

⇧

N x 1 node features

⇧   ⇧   ⇧

local policy   local policy   local policy

⇧   ⇧   ⇧

embedded demand

Design Challenge: constraints

- limits on the number of neighbors due to physical constraints
- Manually enumerating and ruling out all the invalid topologies is time-consuming
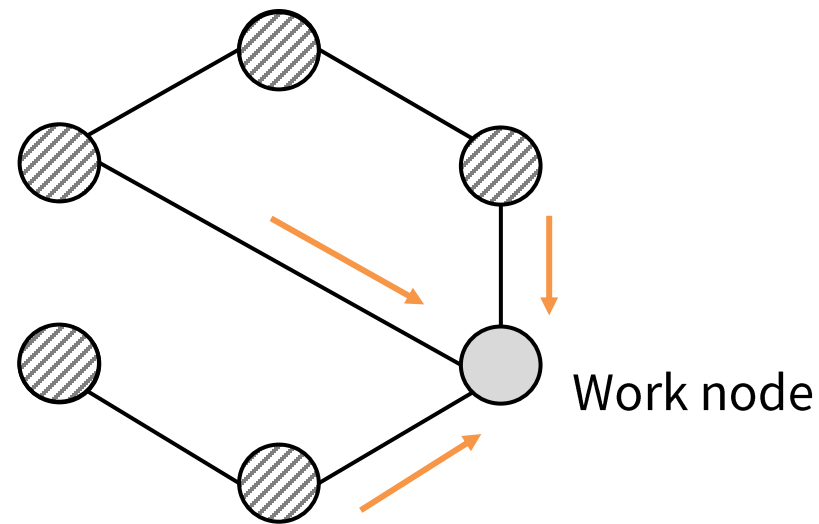- Incremental adjustment:
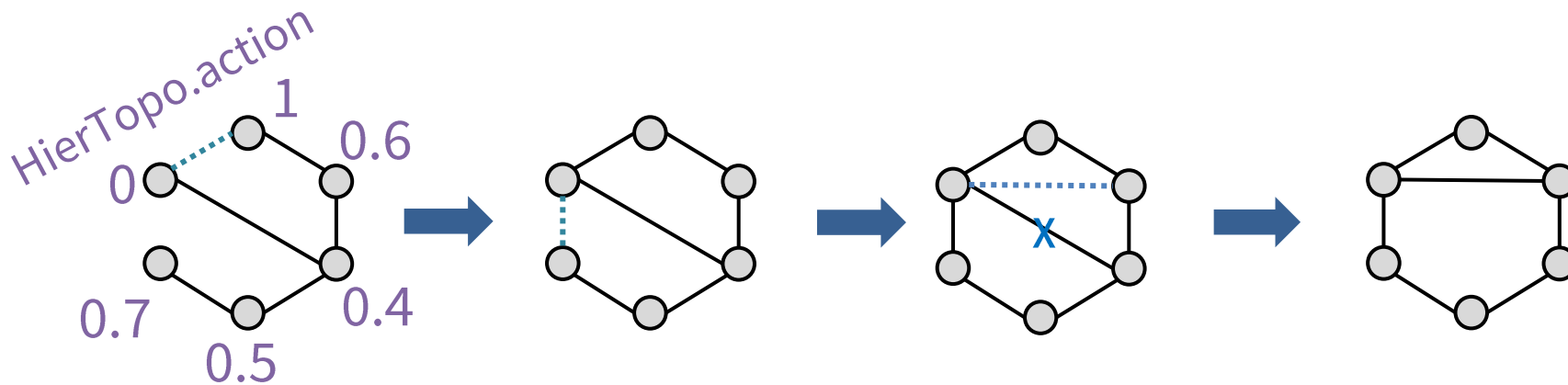
Design Challenge: feeding local policy

- How to collect critical information for the local policy

- Iterating the local policy

Iteration 1

Work node

Iteration 2

Work node
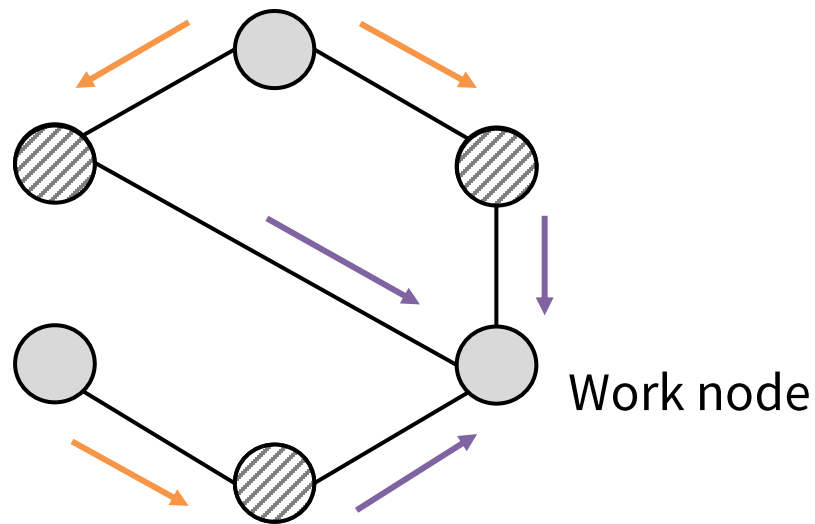
Design Challenge: optimization goal

- optimization goals are hard to be explicitly expressed

Design choices:

- Local policy: polynomial function
  - Expressive enough

- Optimization method: Genetic Algorithm
  - GA does not require a smooth objective function

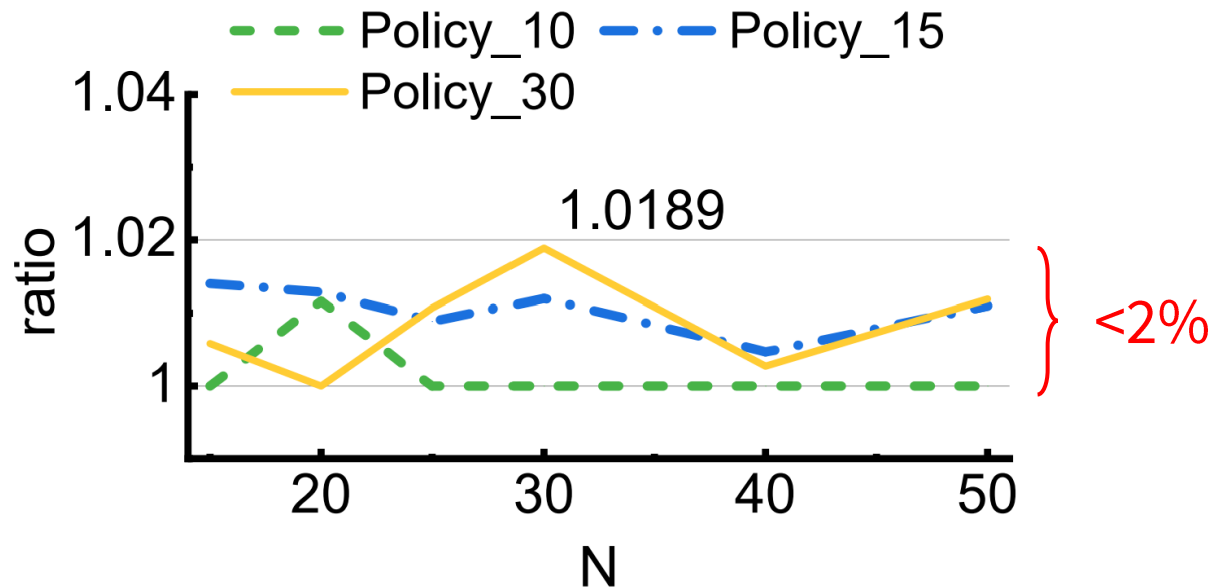# Please refer to our paper for design details and theoretical analysis

Evaluation: Performance

- Traffic distribution

- dataset of cache workload in Facebook

- Baselines

- 1 graph theory baseline

- 2 optimization-based baselines
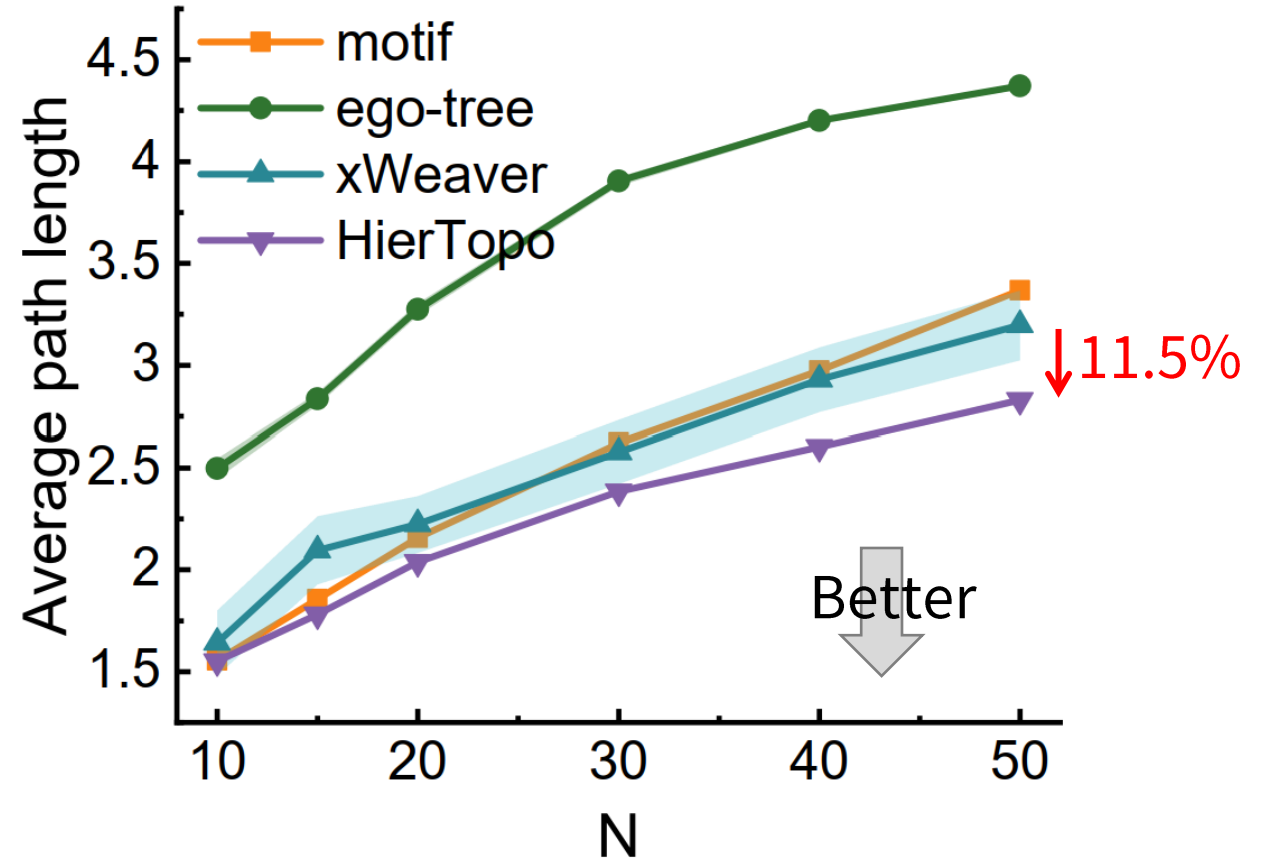
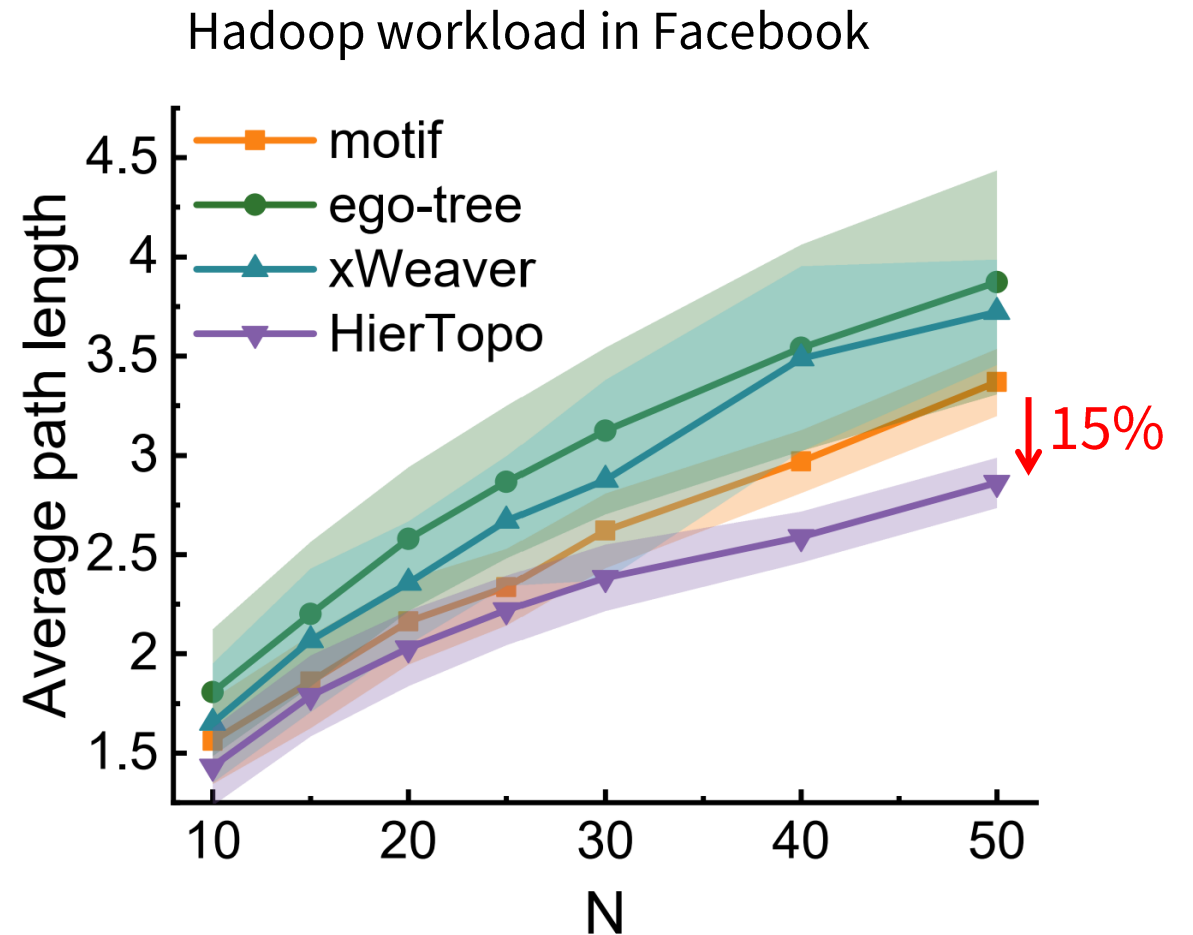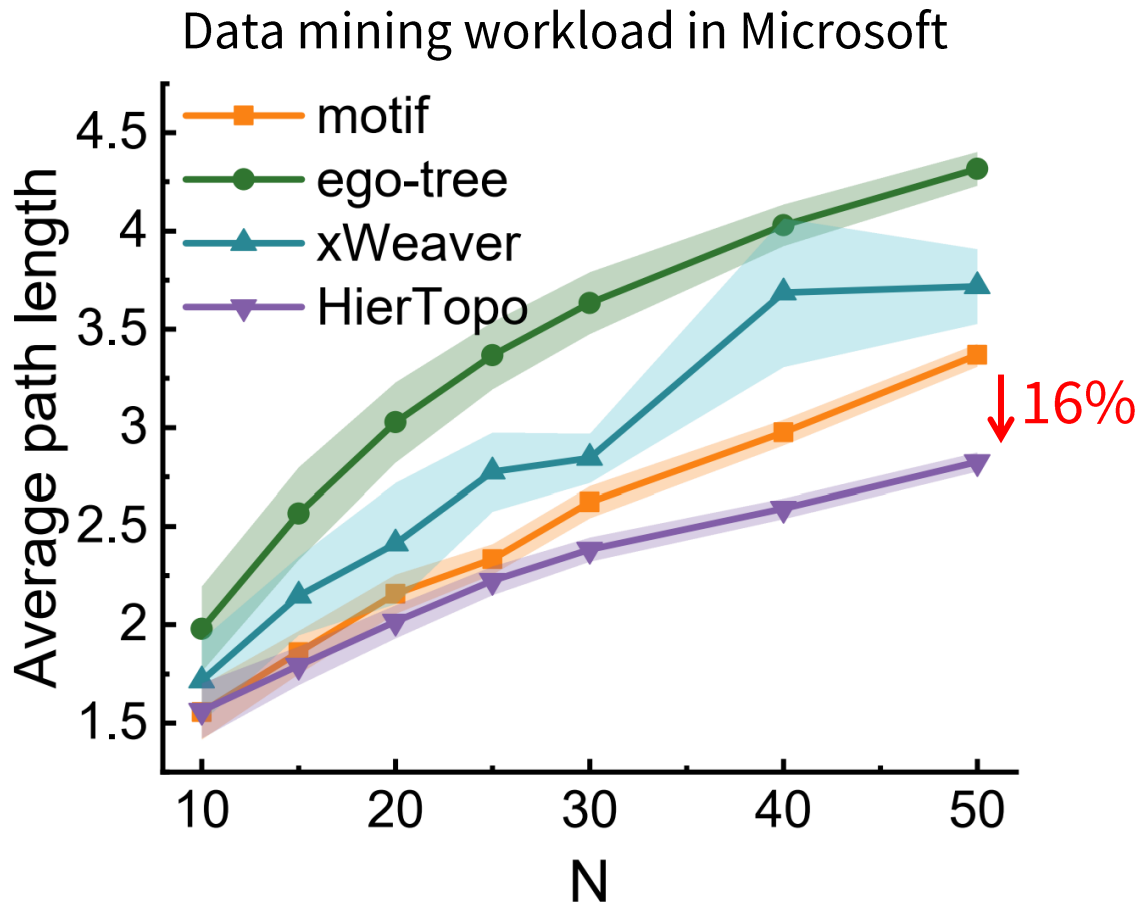# Evaluation: Generalization ability

- Generalize over network scale

Policy_10, Policy_15, Policy_30: the local policy optimized under 10-, 15-, and 30-node networks
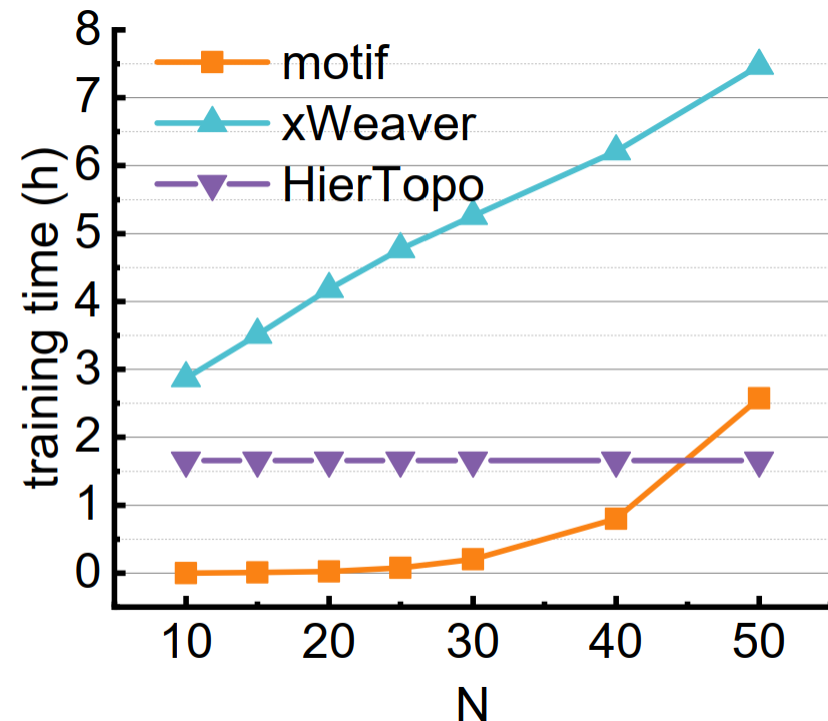
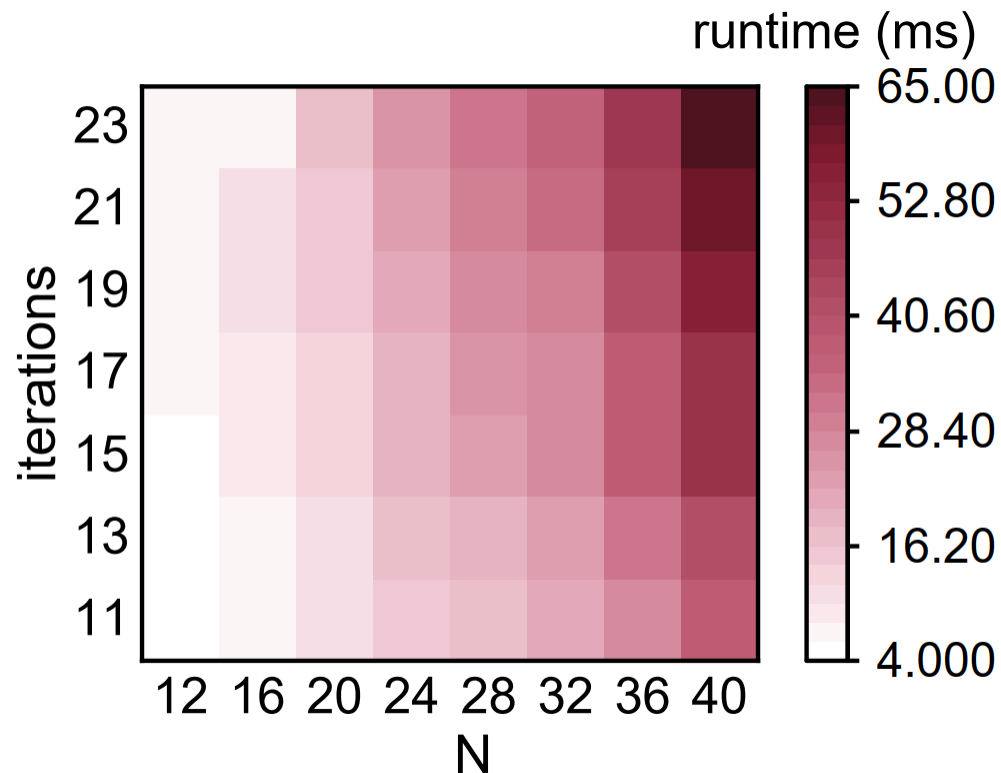# Evaluation: Generalization ability

- Generalize over traffic demand distribution

Data mining workload in Microsoft

Hadoop workload in Facebook

Evaluation: time overhead

- Millisecond-level decision latency
- Training / optimization time

Takeaways

- Current topology design methods suffer a tradeoff between performance and efficiency.

- HierTopo designs network topology in a hierarchical way.

    - Decouple the problem into (local) network information aggregation and (global) link decision.

- HierTopo constructs high-performance topologies efficiently and has advantageous generalization ability

Thank you!
Questions & comments?

j-chen16@tsinghua.org.cn