# Edge-Cloud Networks for Efficient AI/ML Implementations

Jie Wu
Dept. of Computer and Information Sciences
Temple University, USA

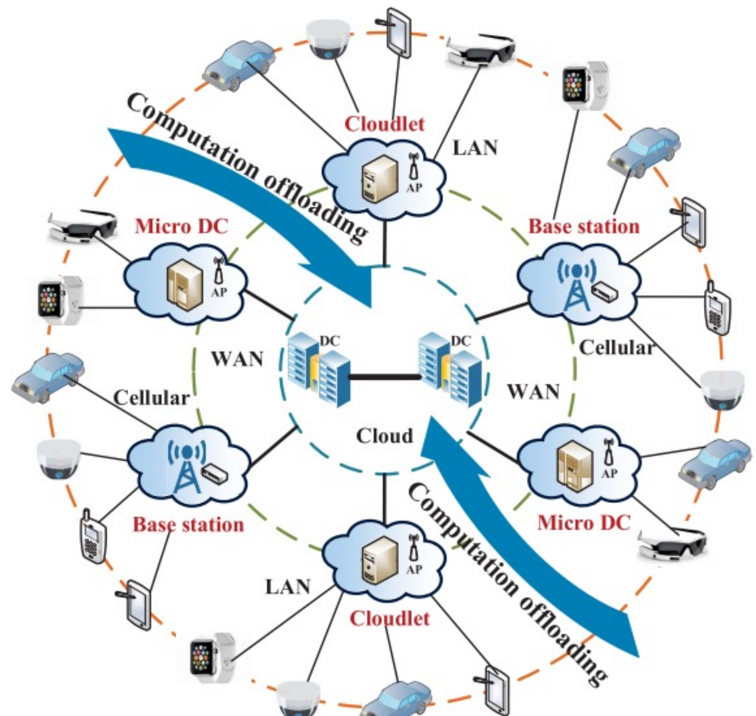
Cloudnet 2023

# Roadmap

1. Edge-Cloud Networks
2. Parallelism: model vs. data
3. Model: Collaborative Edge-Cloud
4. Data: Decentralized Federated Learning
5. Some Final Thoughts

# 1. Edge-Cloud Networks

- Application-driven: AR/VR and LLM (ChatGPT, GPT4)
- Key indicators: latency, accuracy, energy, and privacy
- Latency-sensitive
  - How edge contributes to AI/ML ?
  - How to use rich resources in cloud ?
- Collaboration
  - Edge-Cloud
  - Within Cloud




50 billion IoTs: connected intelligence

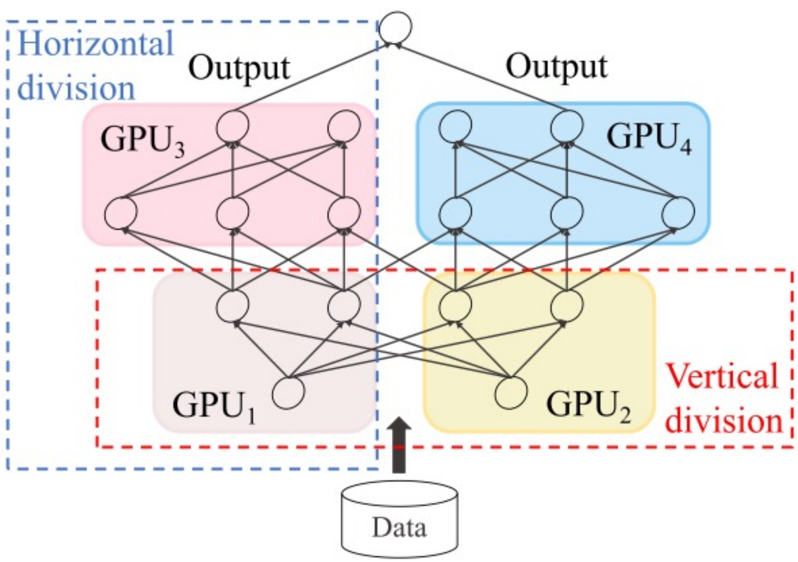
Edge: End IoTs + Edge

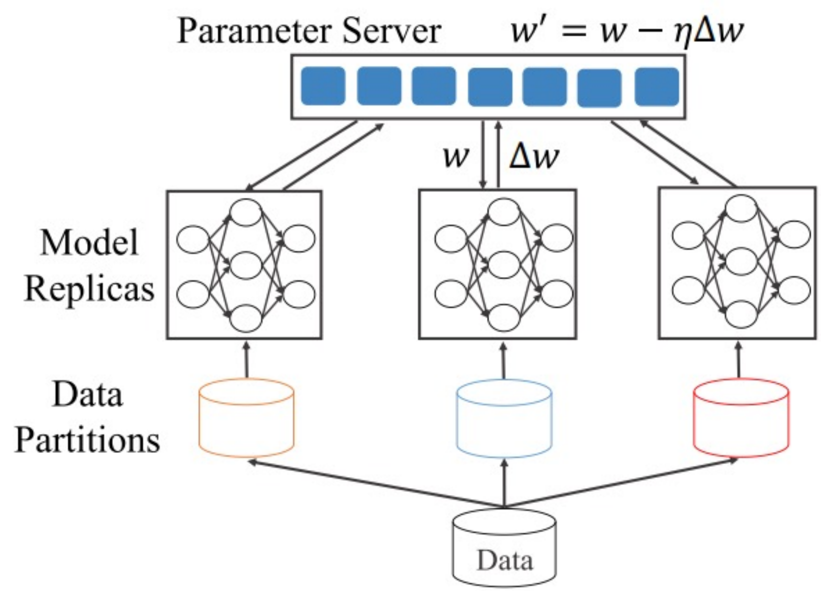
# Efficient AI/ML Implementation

- Work hard
  - Faster processing, e.g., GPU accelerator
- Work smart
  - Partition AI/ML model and map parts to edge-cloud
- AI/ML model and optimization
  - Deep neural networks (DNN)
  - Stochastic gradient descent (SGD)

# 2. Parallelism

- Model (task) parallelism: Edge-cloud collaboration
- Data parallelism: Federated Learning (FL)



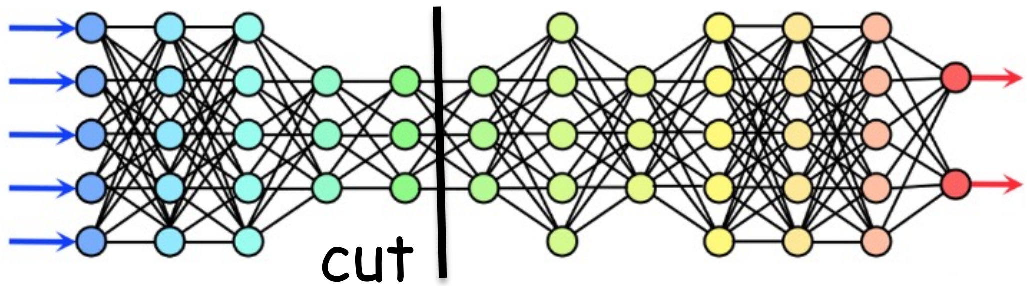

Collaborative edge-cloud

Decentralized federated learning

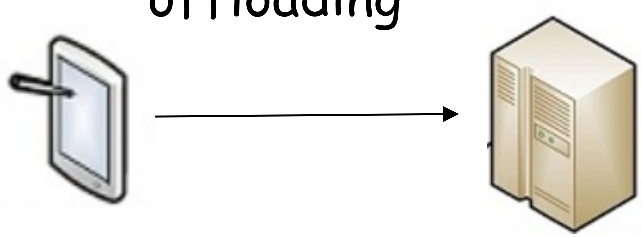
Wu, Parallel Processing: Past, Present, and Future, early 90s

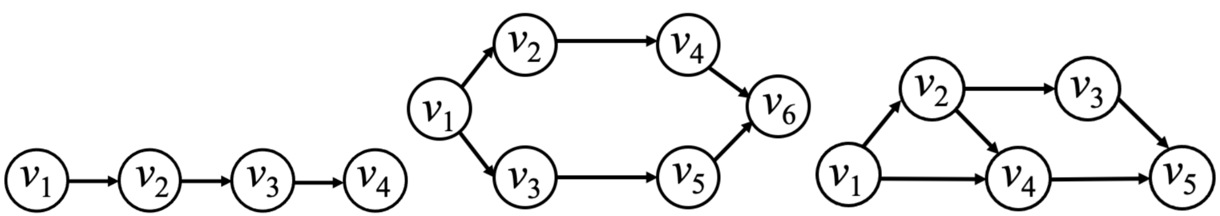
# 3. Model: Collaborative Edge-Cloud

- Three-stage collaborative pipeline and offloading
  - Local, communication, remote (Cloud)




- Three models
  - Device/edge-only
  - Cloud-only offloading
  - Mixed-mode offloading

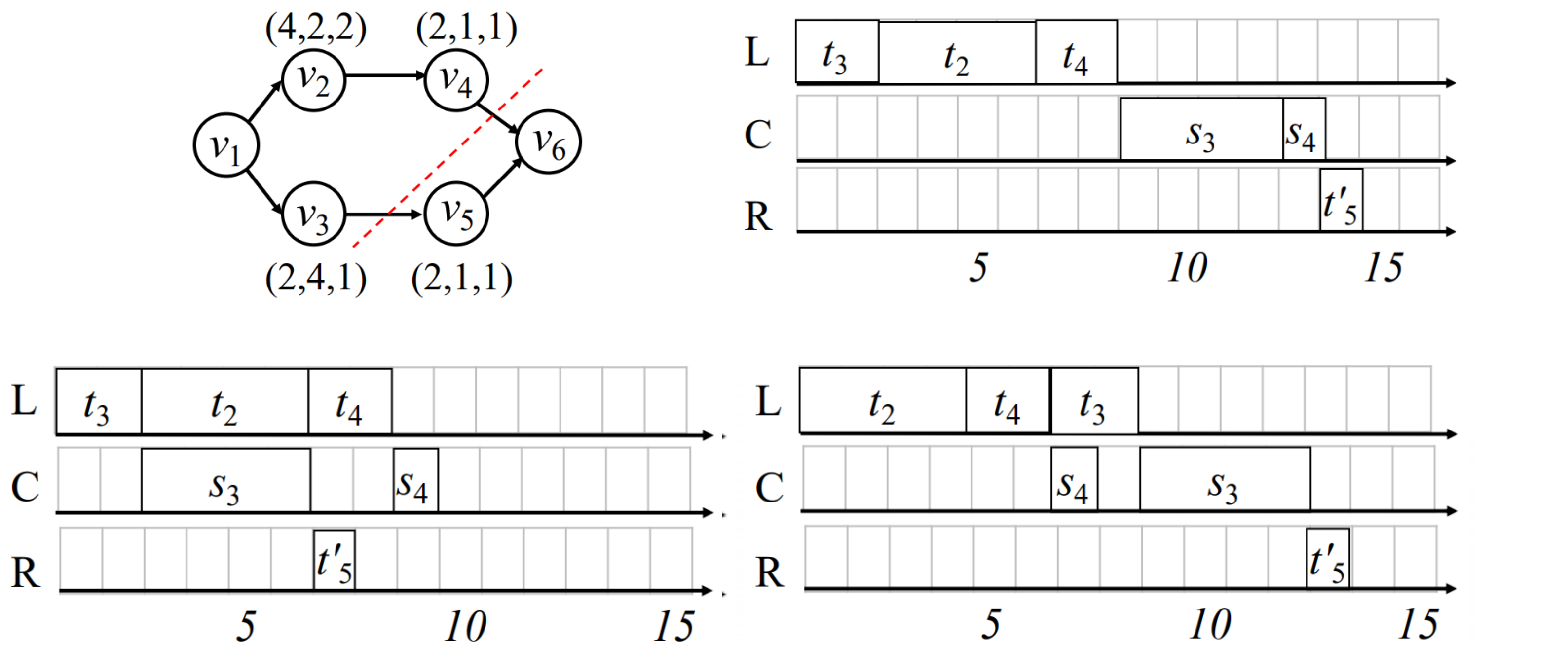



(a) line (b) multi-path (c) DAG

# Offloading Sample: Multiple Paths

- Given a partition (i.e., cut)
  - Fine-grained pipeline: path-based (rather than phase-based)
  - Extended Johnson's solution with approximation ratio



# Multiple DNNs Offloading

Internet of Vehicles: smart city

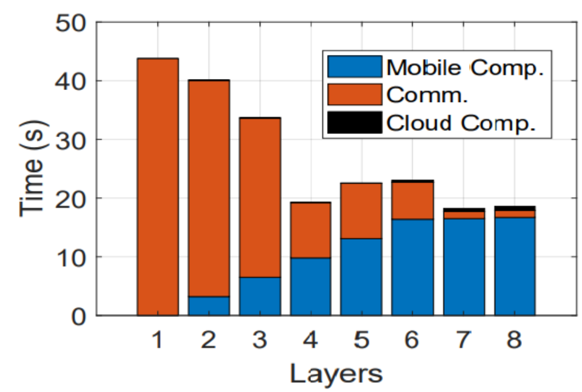
- Autonomous driving systems: perception is a key
- Multiple cameras/sensors: multiple (identical) DNNs
- V2X: V (vehicle); X for I (infrastructure), N (network), P (pedestrian)

# Johnson Algorithm: Multiple Lines

- Computation at cloud can be neglected
- JA for optimal schedule
  - 2-stage pipeline with given partitions

**Algorithm 2** Johnson Algorithm (JA)

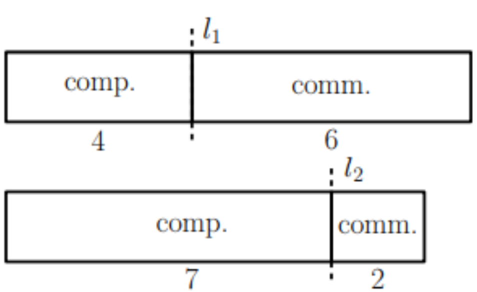
$H \leftarrow L \leftarrow \phi$
**for** $i = 1$ to $m$ **do**
  **if** $p_1(i) \leq p_2(i)$ **then**
    $H = H \cup p(i)$
  **else**
    $L = L \cup p(i)$
Sort $H$ increasingly based on $p_1(i)$
Sort $L$ decreasingly based on $p_2(i)$
Concatenate $H$ and $L$ to obtain $\sigma$

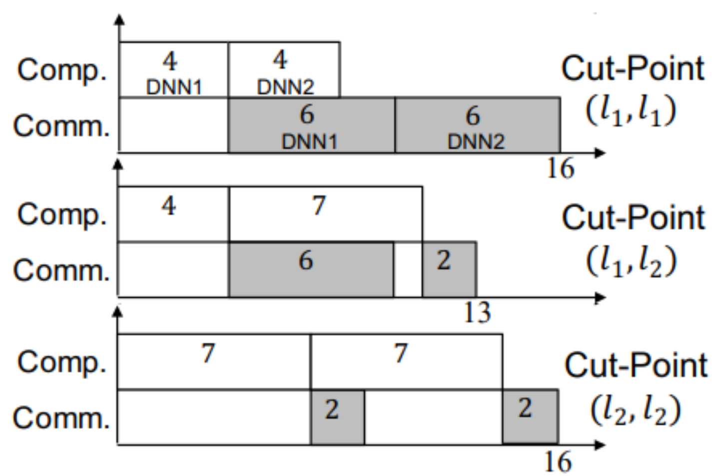


# Partition and Scheduling

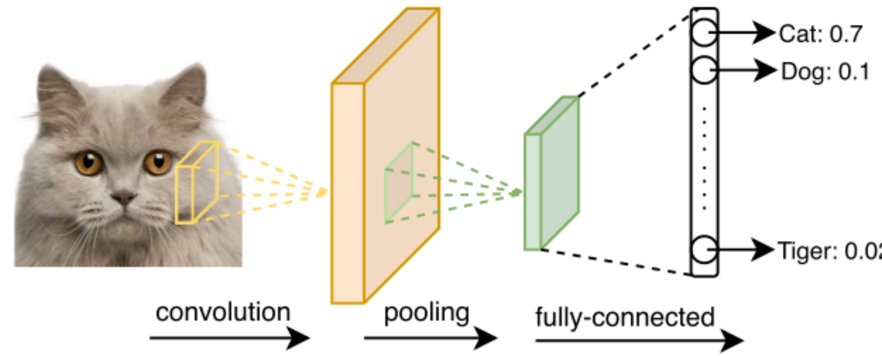
- Partition and scheduling of 2-stage pipeline
  - Brute force: $O(k^n)$

    n: # of copies, k: # of layers
- Existence of a better solution?
  - Exploring special properties

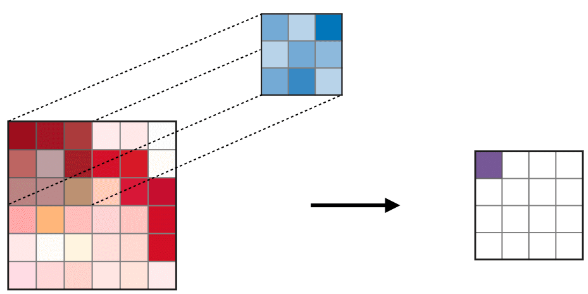


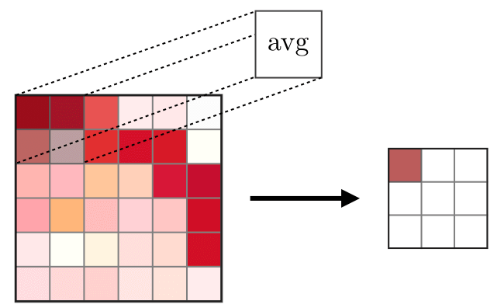



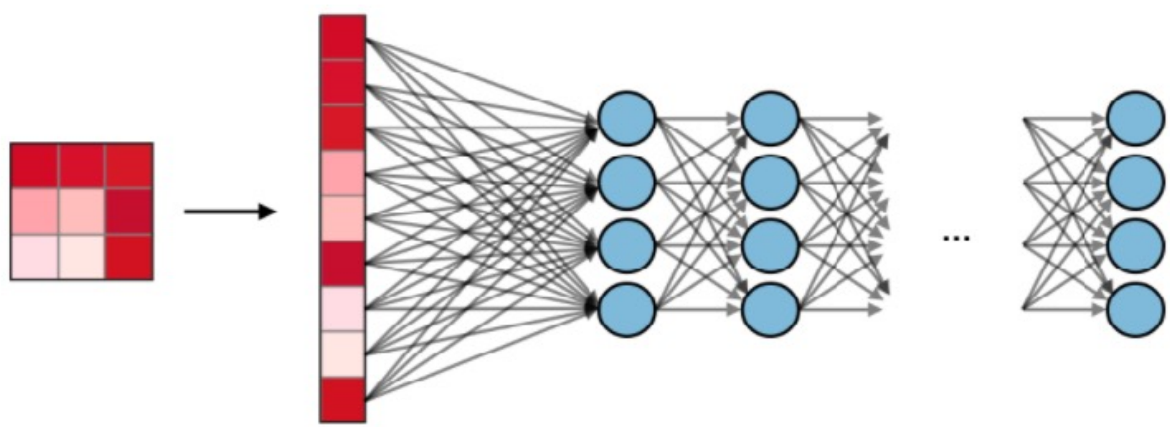
# Convolution NNs




- CNNs (image classification)
- convolution (filtering), pooling (max/avg), fully-connected (neurons)

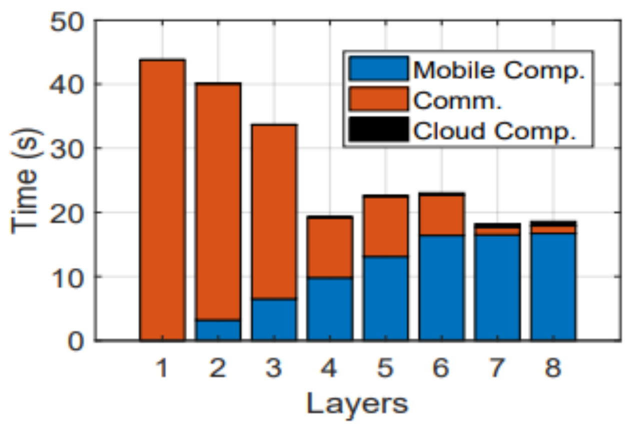


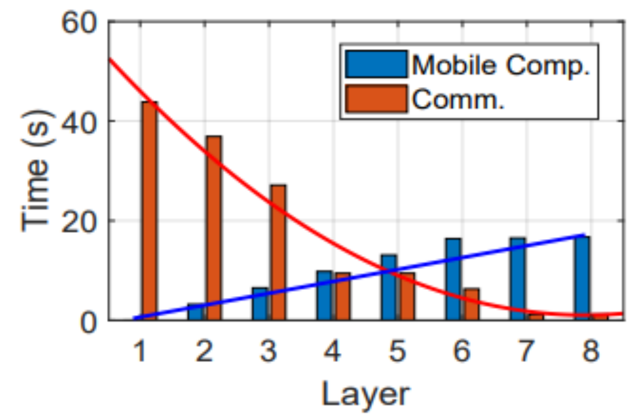


# Special Application Property

- As the number of layers increase
  - Computation time: linear increasing (convex) function
  - Communication time: monotonic decreasing convex function

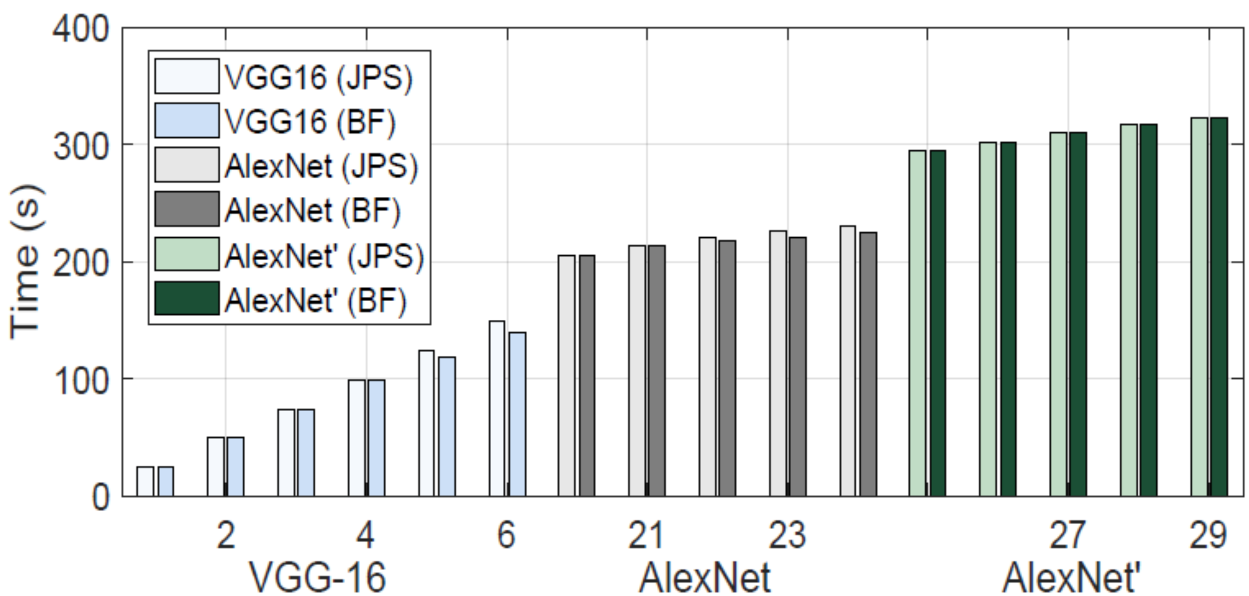



Theorem : A uniform partition of n line DNNs at the intersection will guarantee an approximation of $1 + \frac{1}{n}$.

# Simulation

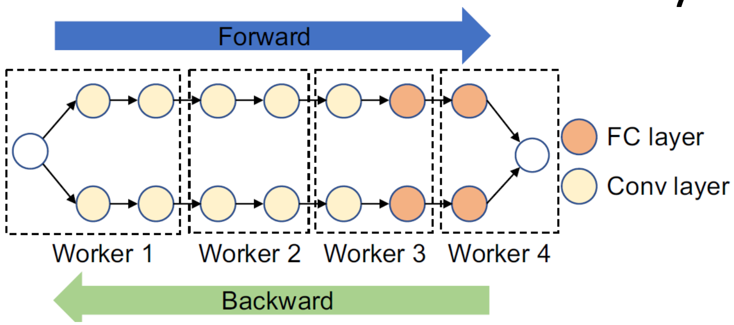
- Partition methods
  - Joint Partition and Scheduling: JPS, Brute Force: BF
- Applications
  - VGG-16, AlexNet, and AlexNet' (curve fitting) with n = 1, …, 29

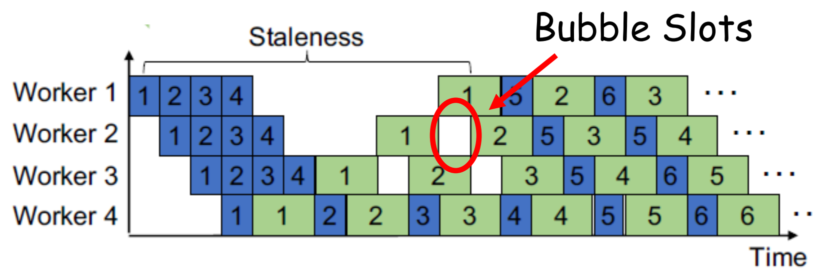


# Extension: Training

## Inference forward pass/training backward pass

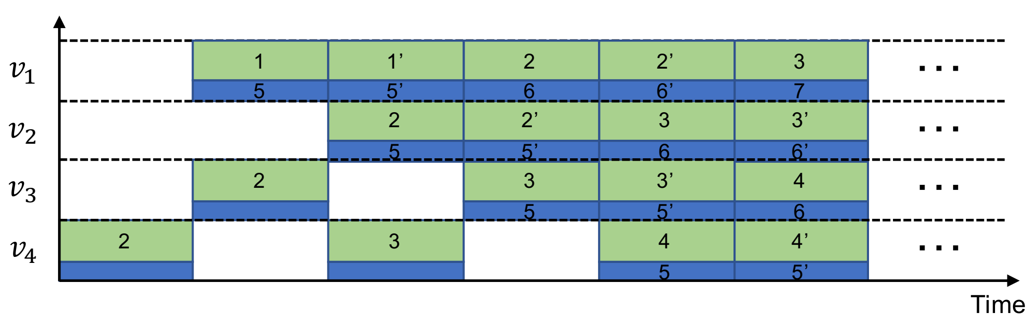
- Reduce resource idle time by adjusting the ratio of resources




## Aligning Pipeline with Resource Allocation

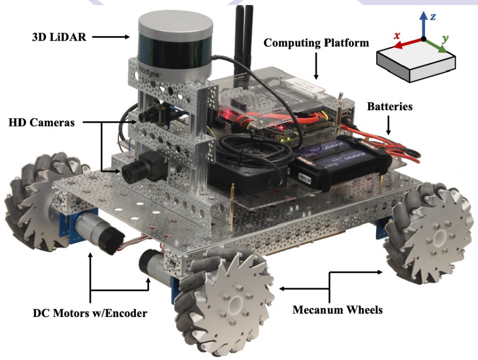
- Combine forward/backward passes (insert 1′ after 1 to fill up space)

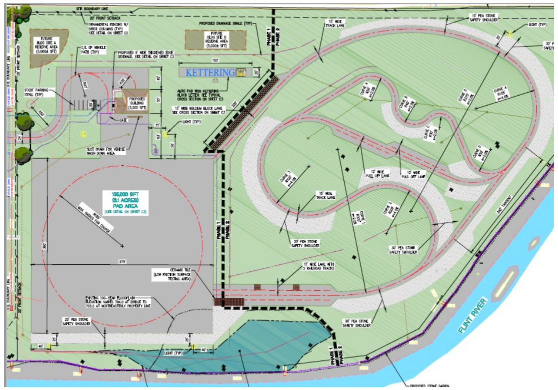



Duan and Wu, Optimizing Job Offloading Schedule for Collaborative DNN Inference, IEEE TMC, 2023.

# An Ongoing Project

- Extension to DNN training
  - Data compression
- Testbed implementation
  - Visual detection & tracking
- Field test
  - KUSARA at Kettering University



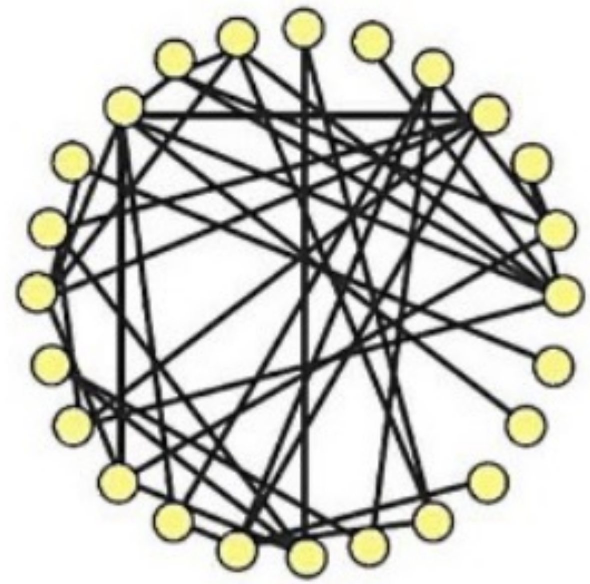



NSF CNS Medium: Cooperative AI Inference in Vehicular Edge Networks for Advanced Driver-Assistance Systems (PI, 2021-2024)

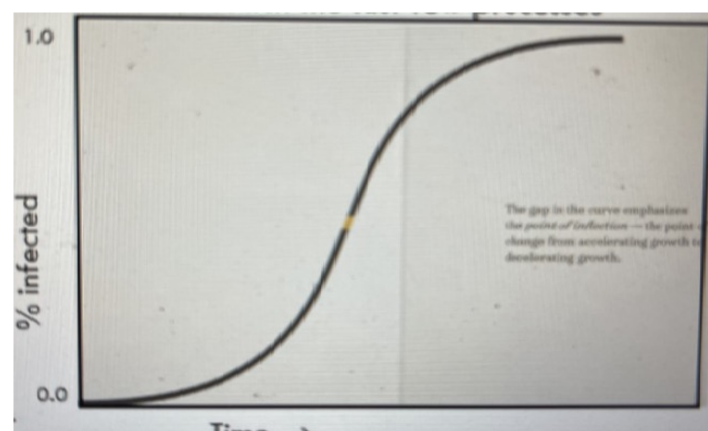
(Temple, Stony Brook, Rowan, and Kettering)

# 4. Data: Decentralized Federated Learning

- DFL
  - CFL shortcoming: central failure
  - Nodes coordinate themselves to obtain the global model

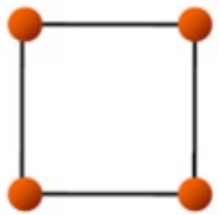


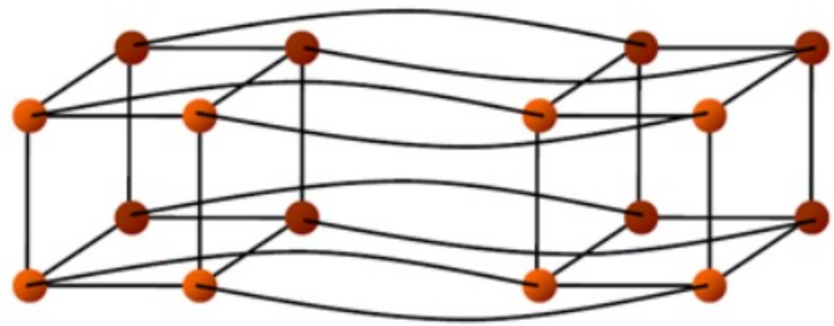
- Gossip learning
  - Exchange/aggregate models
  - Random perfect pairing
  - Comparable performance to CFL
- Merits and drawbacks
  - Easy to use, robust, and robust
  - Drawbacks: long-tail



# Structured Peer-to-Peer (P2P)

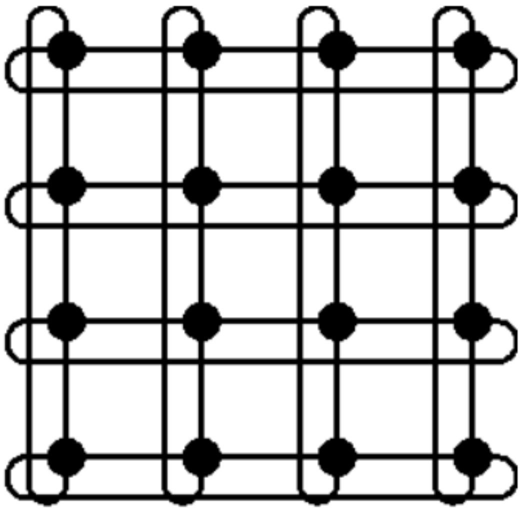
- Spectral gap δ (ML community)
  - The difference between the moduli of the two largest eigenvalues of adjacency matrix W
  - The larger δ is, the faster the convergence
- Sample regular topologies with n nodes (HPC community)
  - Ring (# neighbors d: 2; diameter D: n/2)
  - 2-D torus (4; $\sqrt{n}$ ), and hypercube (log n; log n)





# Relationship between δ and D

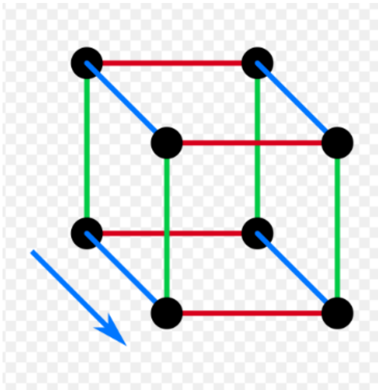
- Known results of δ
  - Rings: $O(1/n^2)$
  - 2-D torus: $O(1/n)$
- Hypercube δ: $O(1/\log n)$
- In general, $\delta = 1/\sqrt{D}$
- To maximize spectral gap is to minimize diameter!



Duan, Li, and Wu, Topology Design and Graph Embedding for Decentralized Federated Learning, accepted to appear in ICN.

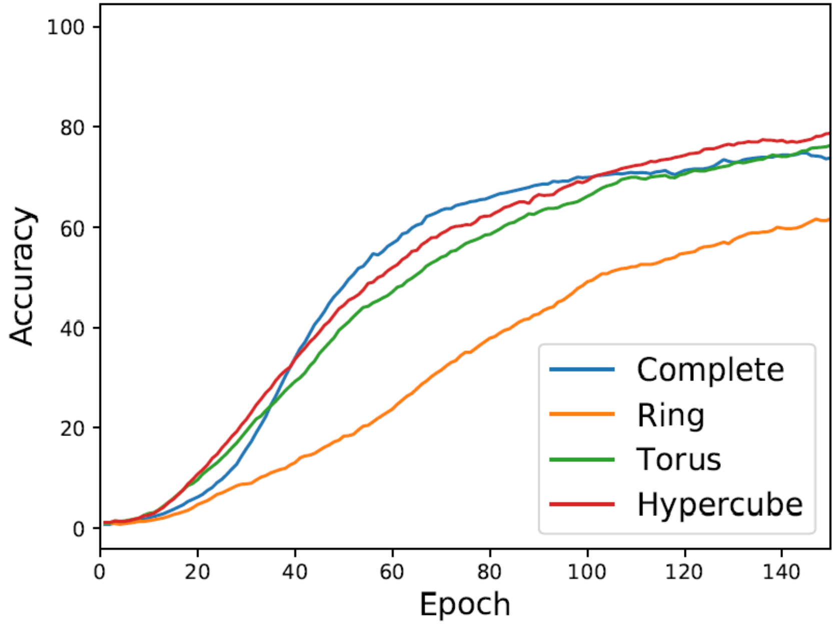
# Graph Embedding based on Similarity

- Select neighbors with max-similarity
  - Maximize total neighbor similarity (product of feature vectors)
  - Similar to a graph embedding in a complete graph
- NP-hard problem: max-similarity for a ring
- Heuristic polynomial algorithms
  - 2-D torus and hypercube
  - 1/log n approximation ratio for hypercube
- Reducing communication frequency
  - Scan dimensions in sequence



# Simulation Results

- Roles of topology on convergence and accuracy




Cluster: 8 NVIDIA Tesla V100 GPUs (448 GB RAM, 5.9 TB)

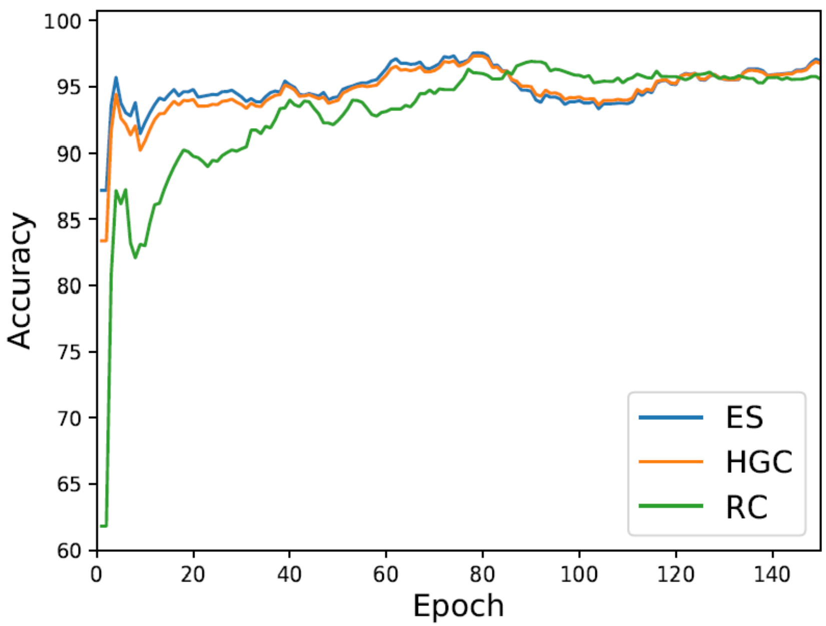
PyTorch and OpenMPI

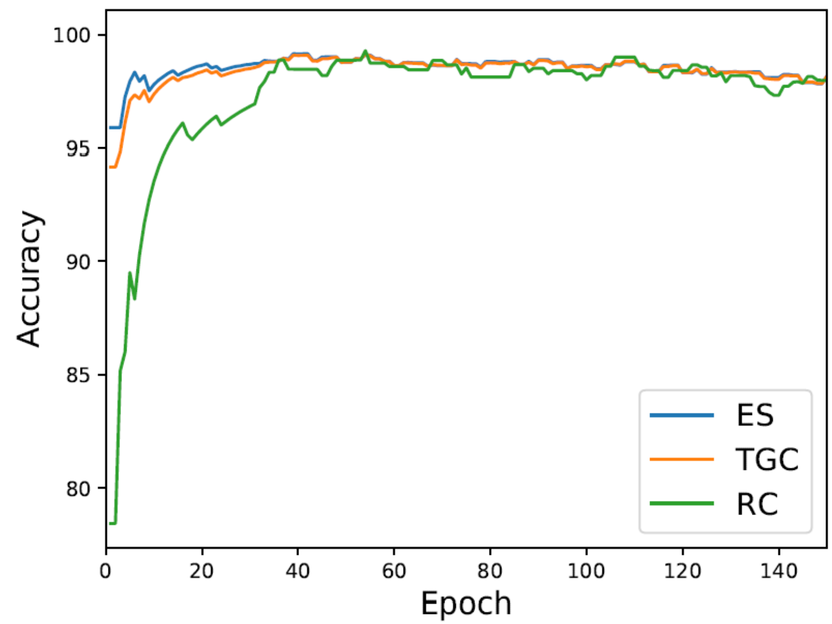
Data: CIFAR-10 and CIFAR-100

ResNet-50 on CIFAR-100 with 64 workers

# Simulation Results (Cont'd)

- Roles of graph embedding based on data similarity

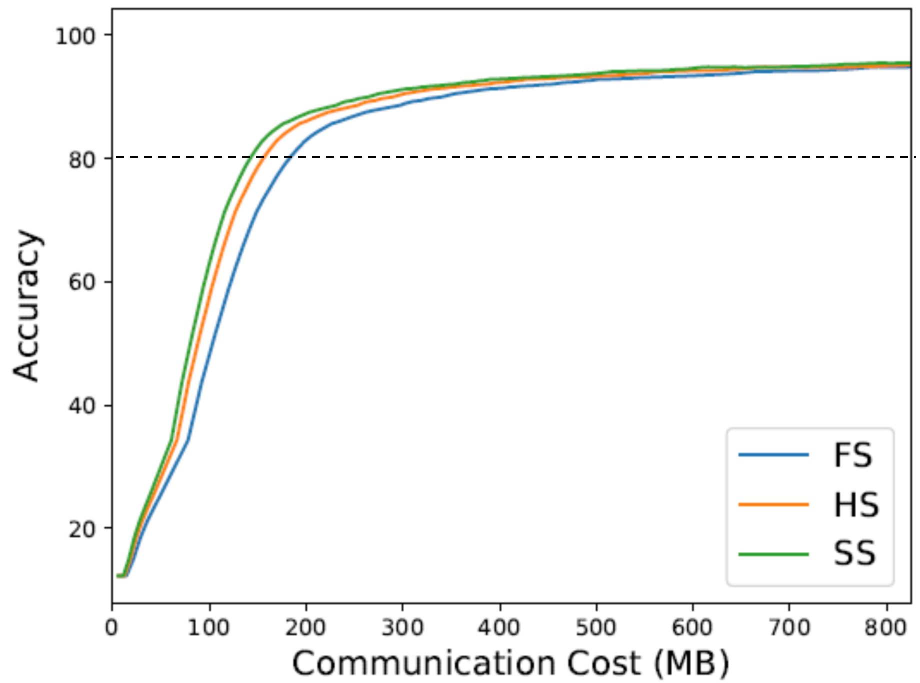



ResNet-50, HGC/TGC: hypercube/torus, ES: optimal, and RC: random

# Simulation Results (Cont'd)

- Roles of communication orchestration on hypercubes

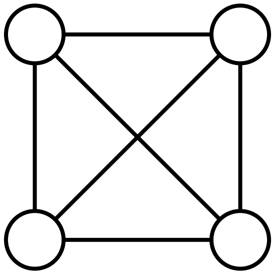


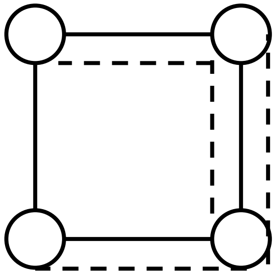

Reaching 80% accuracy for CIFAR-10, SS has 19% lower cost compared to FS (FS: parallel, HS, half-parallel, and SS: sequential)

# Overlay Networks

- Tunneling
  - For fast convergence
  - Emulation of any topology (e.g., all-to-all comm.), but network congestion


- Measurement: load, dilation, and congestion





# 5. Some Final Thoughts

- Offloading: dynamic comm. channel conditions
  - Dynamic cut, compression, link pruning, and phase freezing
  - AUTO-SPLIT for offloading in Huawei Cloud
- DFL: random vs. symmetric graphs
  - Flexibility (on the number of nodes)
  - Congestion (at the network level)
- Resource allocation and elastic computation
  - Resource allocation based on data distribution
  - Data distribution under constraints

Niknami, Sawwan, and Wu, SmartPipe: Intelligently Freezing Layers in Pipeline Parallelism for Distributed DNN Training,"ICPADS, 2023

# Random Graphs

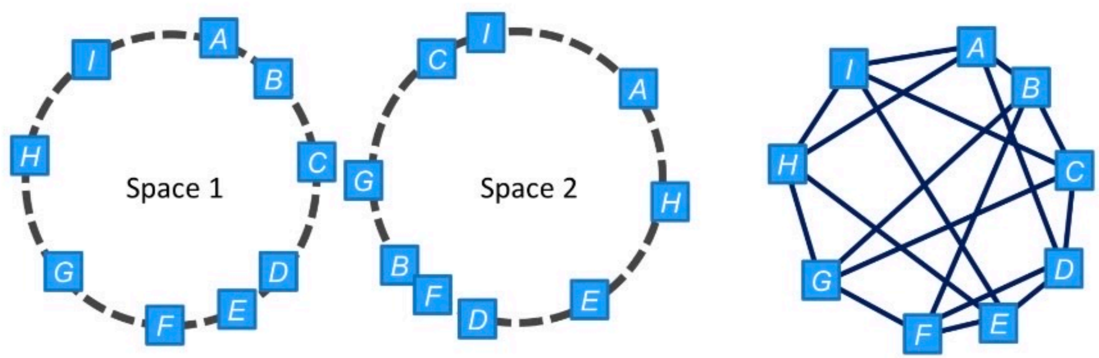
- Works for any number of nodes
- Controlled random graph

e.g., d-regular graph https://arxiv.org/abs/2112.15486

l (=d/2) virtual random space/rings, approaching toward a Ramanujan graph (with a large spectral gap)

| Node ID | Coor. 1 | Coor. 2 |
|---|---|---|
| A | 0.05 | 0.17 |
| B | 0.13 | 0.62 |
| C | 0.23 | 0.91 |
| D | 0.36 | 0.53 |
| E | 0.42 | 0.42 |
| F | 0.51 | 0.58 |
| G | 0.63 | 0.73 |
| H | 0.78 | 0.26 |
| I | 0.91 | 0.97 |

(a) Coordinates

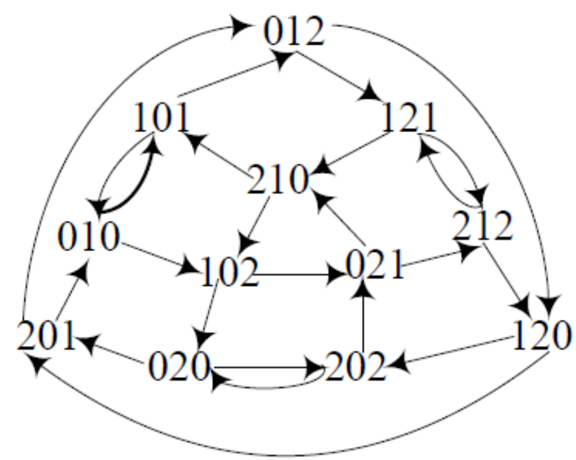



(b) Virtual spaces (c) Actual topology

# Symmetric Graphs

- Moore bound
  - max n, given diameter D and node degree d

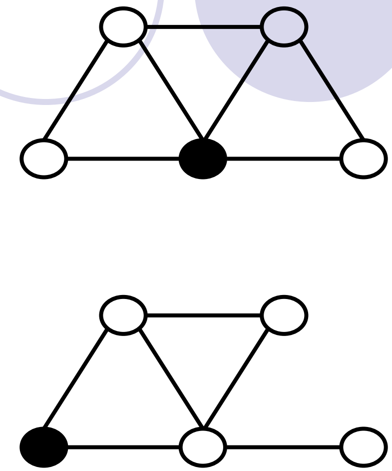
    $n = d^D + d^{D-1} + \ldots + d^1$




- Symmetric graphs
  - Moore bound has not been reached
- Kautz digraph
  - $n = d^D + d^{D-1}$ i.e., $D = O(\log n)$ for const. d, symmetric, and c-congestion-free
  - Simulate all-to-all comm. with c congestion

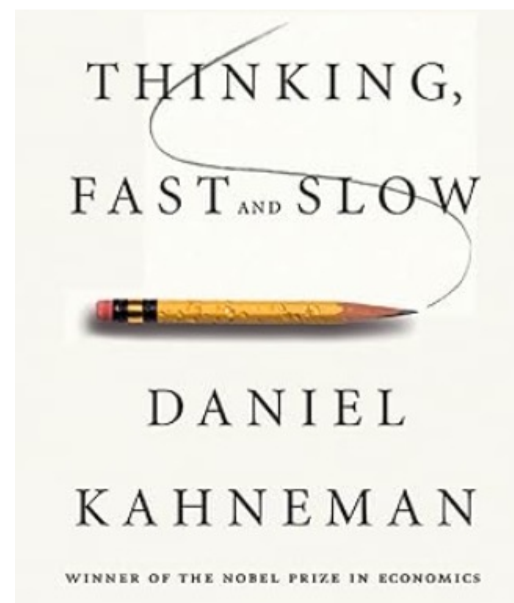
Li, Lu, and Wu, FISSIONE: A Scalable Constant Degree and Low Congestion DHT Scheme Based on Kautz Graphs, INFOCOM 2005

# Other learning models



2-hop local views in graph learning

- Other FL models
  - CFL, DFL, and HFL (multi-tier)
  - Federated reinforcement/graph learning
- Time-varying graphs
  - Learning rates and topology selections
- Beyond P2P: multi-models
  - Local (fast) and global (slow) models
- Information sharing
  - Push, Pull, and hybrid

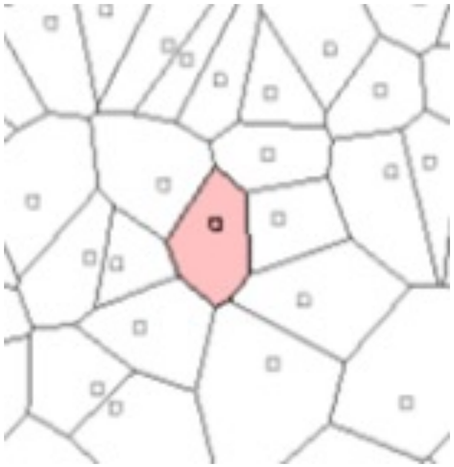


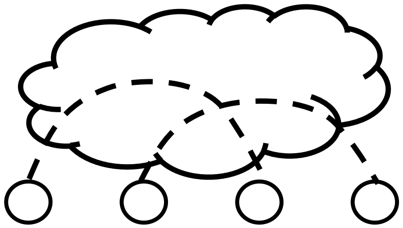

D. Kahneman: Thinking, Fast and Slow, 2011

# Resource Allocation

Site and data locations are fixed

- Resource assignment based on data
  - Voronoi diagram to min. data-movement distance
- Data assignment based on resource
  - based on site and network capacity
  - Elasticity: offering maximum future growth under the gossip model

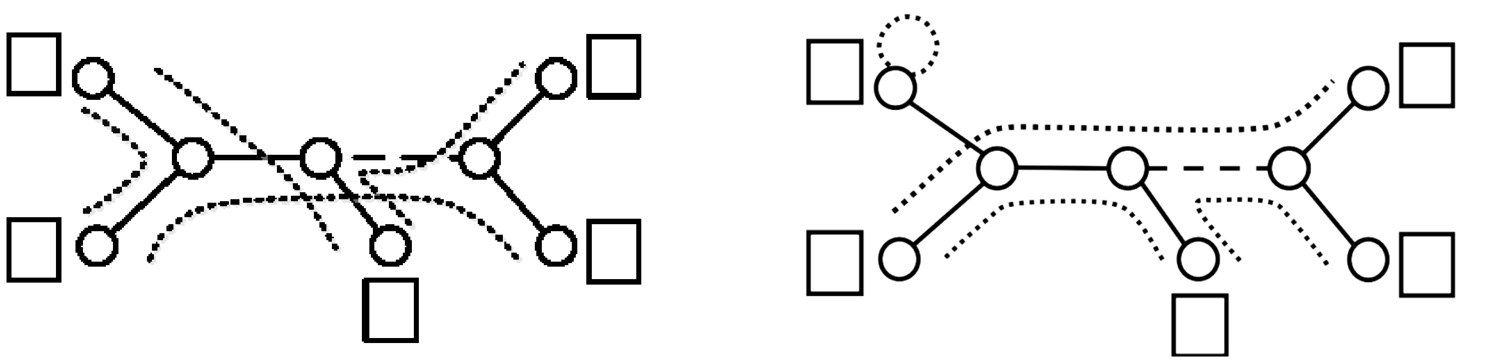




Wu, Lu, and Zheng, On Maximum Elastic Scheduling of Virtual Machines for Cloud-based Data Center Networks, ICC, 2018.

# Maximum Elastic Scheduling

- Given a cable connection in a graph, each household has an *occupancy limit* and each cable section has *bandwidth limit*.
- What is the maximum total occupancy that can support all possible simultaneous pairwise telephone conversations (hose model)?
- What is the schedule with the maximum elasticity (i.e., maximum uniform growth in occupancy)?



hose model: statistical multiplexing. topology: tree and general graph

# Questions