Maximum Elastic Scheduling based on the Hose Model

基于软管模型的最大弹性调度

Jie Wu (吴杰)
Temple University (天普大学)

HPC China 2019

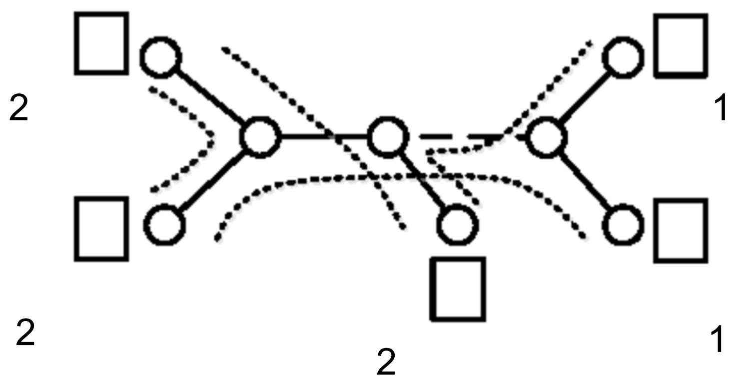
1. AI Takeoff

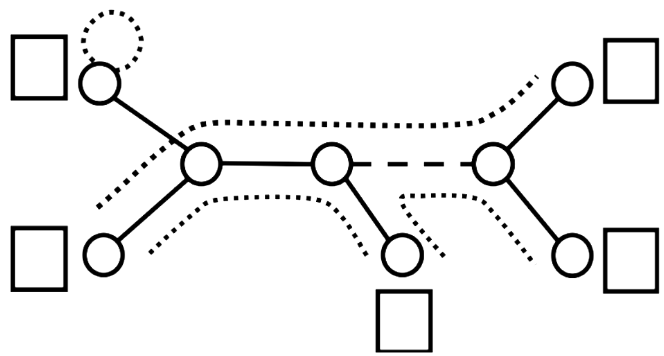
- Deep Blue
  - 1997: defeated Kasparov.
  - ICPP'96 panel: F. –H. Hsu (許峰雄) talked about LB instead.
- HPC-AI convergence
  - AI blackbox (黑箱子)
  - However, DARPA: Explainable AI (XAI)
    - Produce more explainable models
    - Enable human users to understand

- Back to fundamentals
  - Direct algorithmic/combinatoric solutions
  - A scheduling problem related to maximum elasticity

A Simple Illustration

- 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 (软管模型): statistical multiplexing

2. Model and Formulation

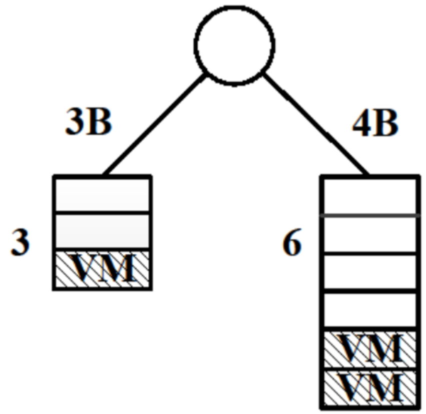
How to define elasticity?

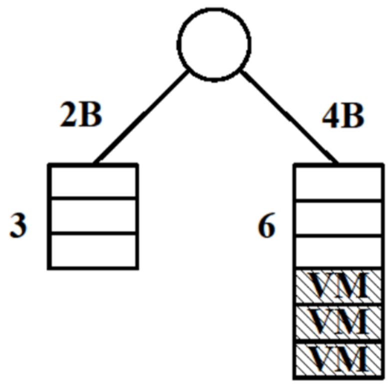
- Maximum Admissible Load (MAL) 最大容许负载
  - Provisioning MAL of VMs in PMs for hose-model-based DCNs
- Maximum Elastic Scheduling (MES) 最大弹性调度
  - A task assignment of a given load (< MAL) with potential maximum uniform growth in computation and communication

A Simple 2-Level Tree

On DCN (数据中心网络), DCN cloud, or Internet cloud

G = (V, E), V: server (服务器) or switch (交换器), E: link (链路)





MAL: 3 VM +6 VM =9 VM
MES for 3: 1+2
Max. Elasticity: 200%

MAL: 2+6 =8
MES for 3: 1+2 or 0+3
Max. Elasticity: 100%

Each VM has 1B Gbps aggregate bandwidth

How to Solve It (Polya)

If you can't solve a problem, then there is an easier problem you can solve: find it

- Tree topology (typical DCN)

Direct solutions

- Shortest path problem (最短路径)
  - LP solution
  - Greedy solution: Dijkstra algorithm
- Maximum elastic scheduling (最大弹性调度)
  - LP solution
  - Greedy solution: Two-phase sweep

LP Solution

$$maximize \qquad e \qquad (1)$$

$$\text{s.t. } e \le min_i(1 - \frac{x_i}{N_i}) \quad \text{and} \quad x_i \le N_i \quad \text{for } \forall i \qquad (2)$$

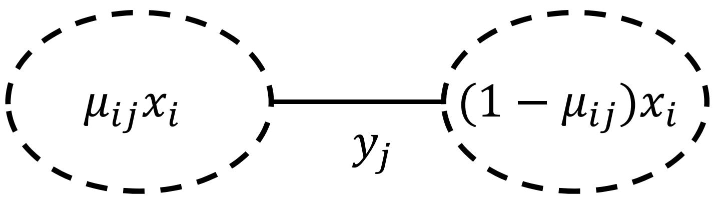
$$e \le min_j(1 - \frac{y_j}{L_j}) \quad \text{and} \quad y_j \le L_j \quad \text{for } \forall j \qquad (3)$$

$$y_j = min\left[\sum_i \mu_{ij} x_i, \sum_i (1 - \mu_{ij})\, x_i\right] \text{ for } \forall j \qquad (4)$$

Eq. (1): objective function

Eq. (2) and Eq. (3): constraints on nodes (N_i) and links (L_j)

Eq. (4):



μ_{ij}: 0 or 1

i^{th} node on j^{th} link

LP Solution (cont'd)

$$\text{maximize} \quad e \qquad (5)$$

$$\text{s.t. } e \le min_i(1 - \frac{x_i}{N_i}) \quad \text{and} \quad x_i \le N_i \quad \text{for } \forall i \qquad (6)$$

$$e \le min_j(1 - \frac{y_j}{L_j}) \quad \text{and} \quad y_j \le L_j \quad \text{for } \forall j \qquad (7)$$

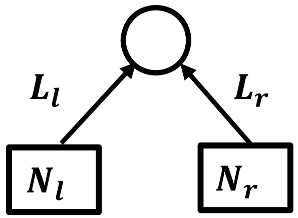
$$y_j \le \sum_i \mu_{ij} x_i \quad \text{and} \quad y_j \le \sum_i (1 - \mu_{ij}) x_i \quad \text{for } \forall j \qquad (8)$$

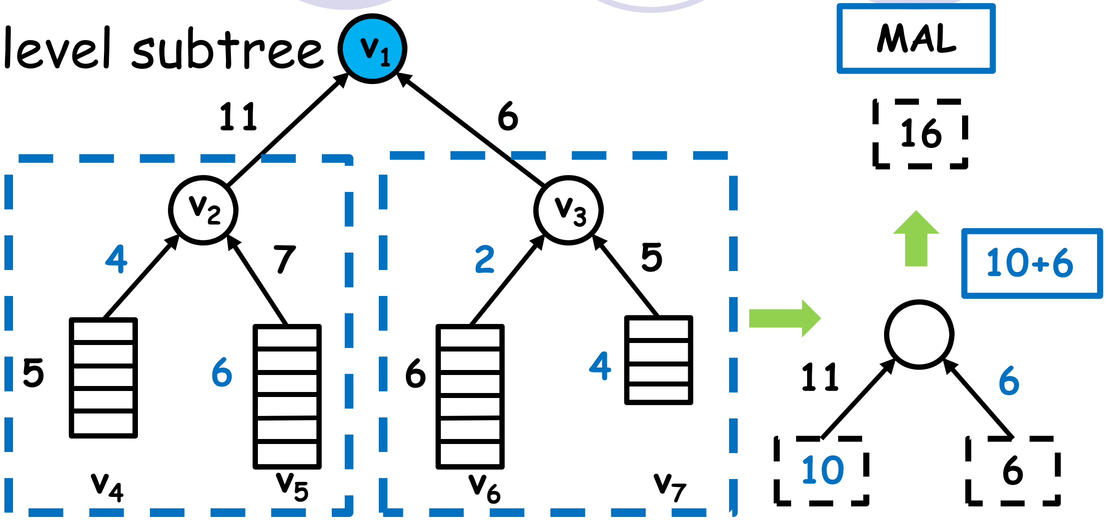
- Variables: 3n−1
  - n: # of leaf nodes
  - 2n-2: # of links
  - 1: objective function e
- Constraints: 10n−8
  - Eq. (6): 2n
  - Eq. (7): 4n – 4
  - Eq. (8): 4n – 4
- Inefficiency: Simplex or Eclipse

3. Two-Phase Sweep Solutions

Up phase: Cal. MAL of a 2-level subtree

$$min\{N_l, L_l\} + min\{N_r, L_r\}$$

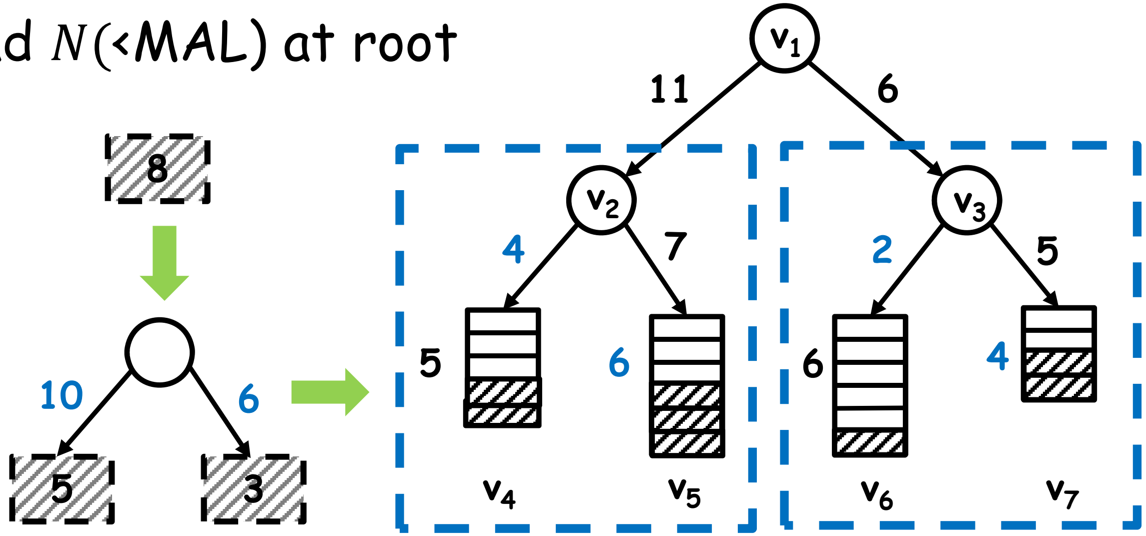




Down phase: Given a load N(<MAL) at root

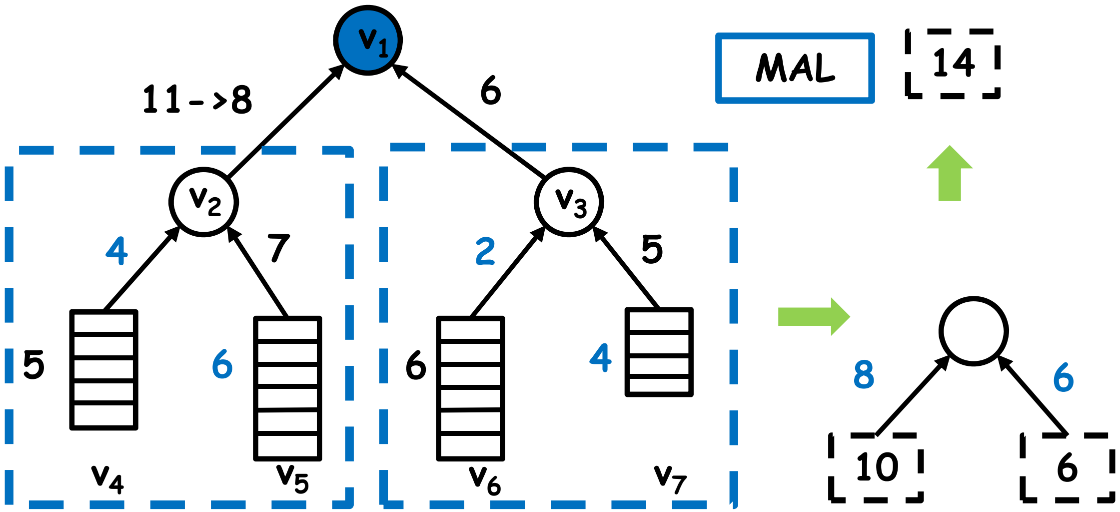
Left $min\{N_l, L_l\}/N$

Right $min\{N_r, L_r\}/N$

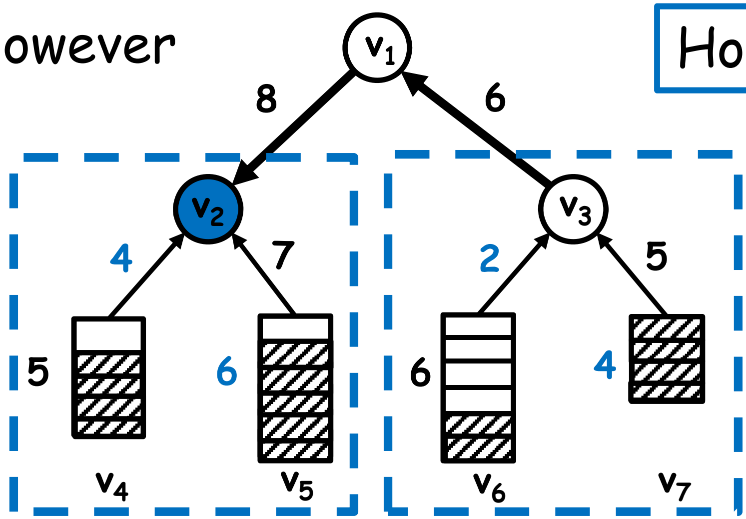


Why Simple Solution May Fail?

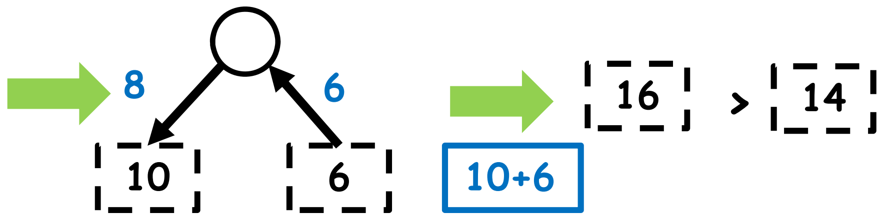
A simple solution



However



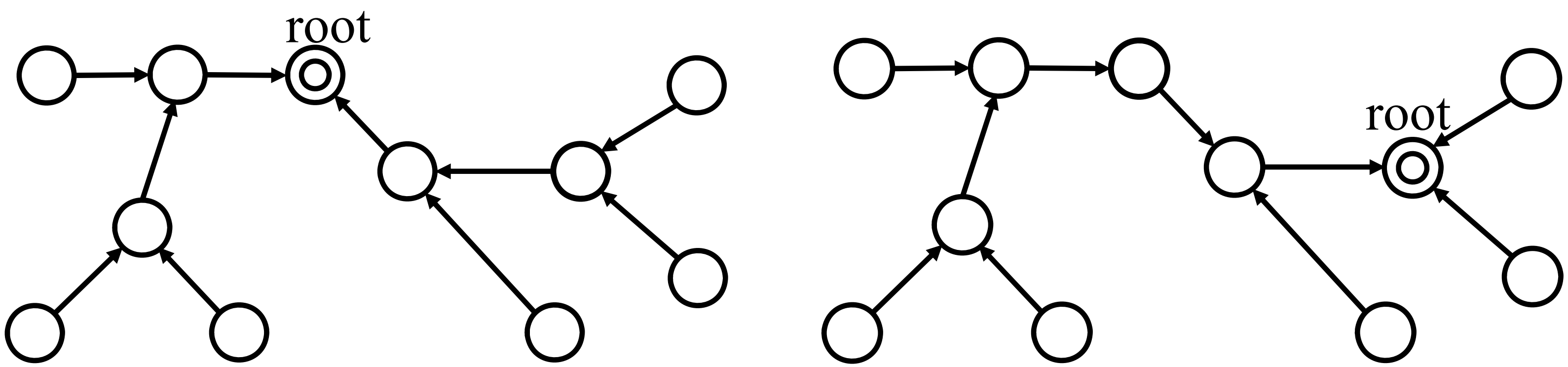
How to find the OPT Solution?



How to Calculate?

Hose-model tree orientation

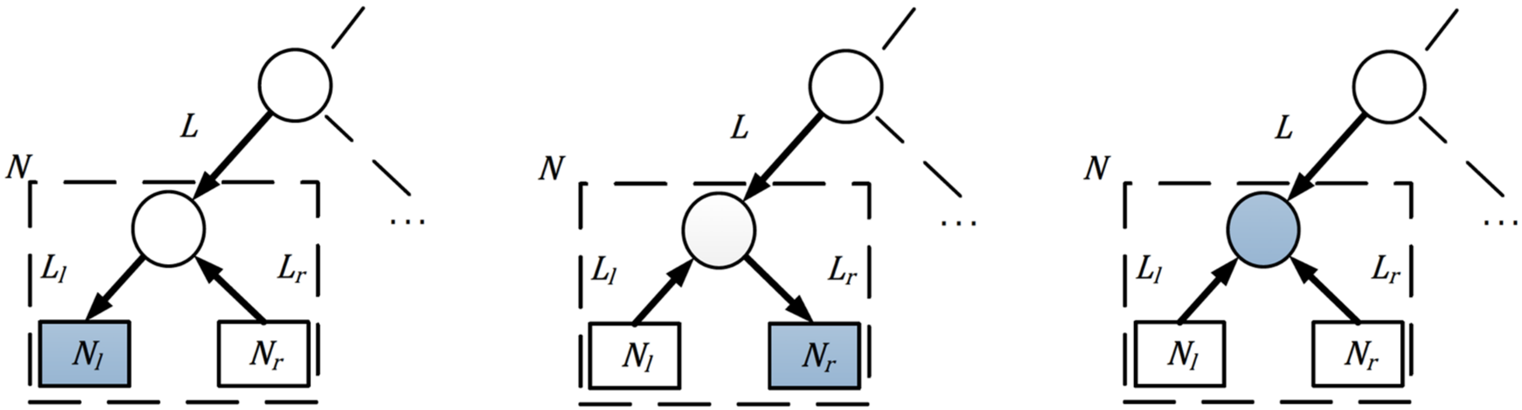
- Directed tree: Link orientation is based on the selected root.
- Find a root with the maximum summation of branch values.



Optimal Solution

Insights

- Apply the simple solution to different orientations.
- Select the best orientation.

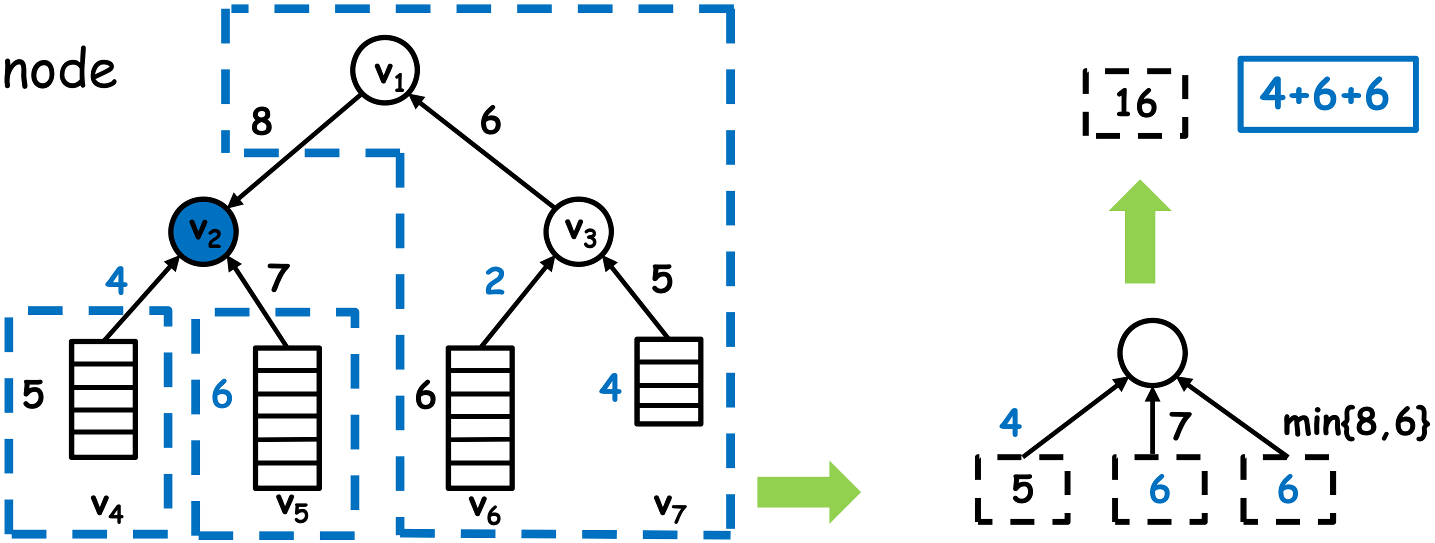


(a) (b) (c)

MAL at the left leaf | MAL at the right leaf | MAL at the center

Distributed Implementation

At each node



	v_1	v_2	v_3	v_4	v_5	v_6	v_7
Step 1	-	-	-	**send** 5 **to** v_2	**send** 6 **to** v_2	**send** 6 **to** v_3	**send** 4 **to** v_3
Step 2	-	**send** $\min\{5,4\}+\min\{6,7\}=10$ **to** v_1	**send** $\min\{6,2\}+\min\{4,5\}=6$ **to** v_1	-	-	-	-
Step 3	**send** $\min\{6,6\}=6$ **to** v_2 **send** $\min\{10,8\}=8$ **to** v_3	-	-	-	-	-	-
Step 4		**send** $\min\{6,8\}+\min\{6,7\}=12$ **to** v_4 **send** $\min\{6,8\}+\min\{5,4\}=10$ **to** v_5	**send** $\min\{8,6\}+\min\{4,5\}=10$ **to** v_6 **send** $\min\{8,6\}+\min\{6,2\}=8$ **to** v_7	-	-	-	-
MAL	$\min\{10,8\}+\min\{6,6\}=14$	$\min\{5,4\}+\min\{6,7\}+\min\{8,6\}=16$	$\min\{6,2\}+\min\{4,5\}+\min\{8,6\}=12$	$\min\{12,4\}+\min\{5,\infty\}=9$	$\min\{10,7\}+\min\{6,\infty\}=13$	$\min\{10,2\}+\min\{6,\infty\}=8$	$\min\{8,5\}+\min\{4,\infty\}=9$

4. Properties and Extensions

Theorem 1: The up-phase determines the MAL.

Theorem 2: The two-phase solution generates a schedule with maximum elasticity.

Theorem 3: The two-phase solution uses $2\log n+1$ parallel steps. The computation complexity is $5(n-1)$, and the communication complexity is $4(n-1)$.

Extensions

- General trees
  - Any k-nary trees
- Optimal simple solution
  - Trees with computational-bottleneck
- Fat trees (used in DCN)
  - Still work !

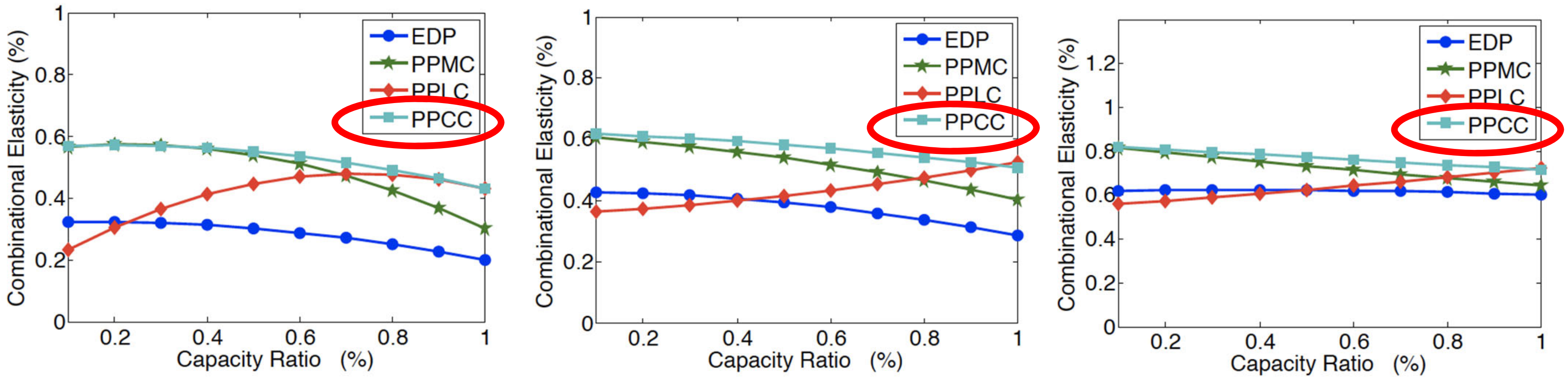
5. Performance Comparisons

- Basic setting
  - Binary trees with levels: k = 4, 5, and 6
  - Node capacity: 0 to 100 units
  - Link bandwidth: 0 to 100 GB
  - Bandwidth demand: 1 Gbps
- Comparison algorithms
  - Equally Distributed Placement (EDP)
  - Proportion to PM Capacities (PPMC)
  - Proportion to Physical Link Capacities (PPLC)
  - Proportion to PM and Channel Capacities (PPCC)

Binary Tree Simulation

Comparison of the elasticities

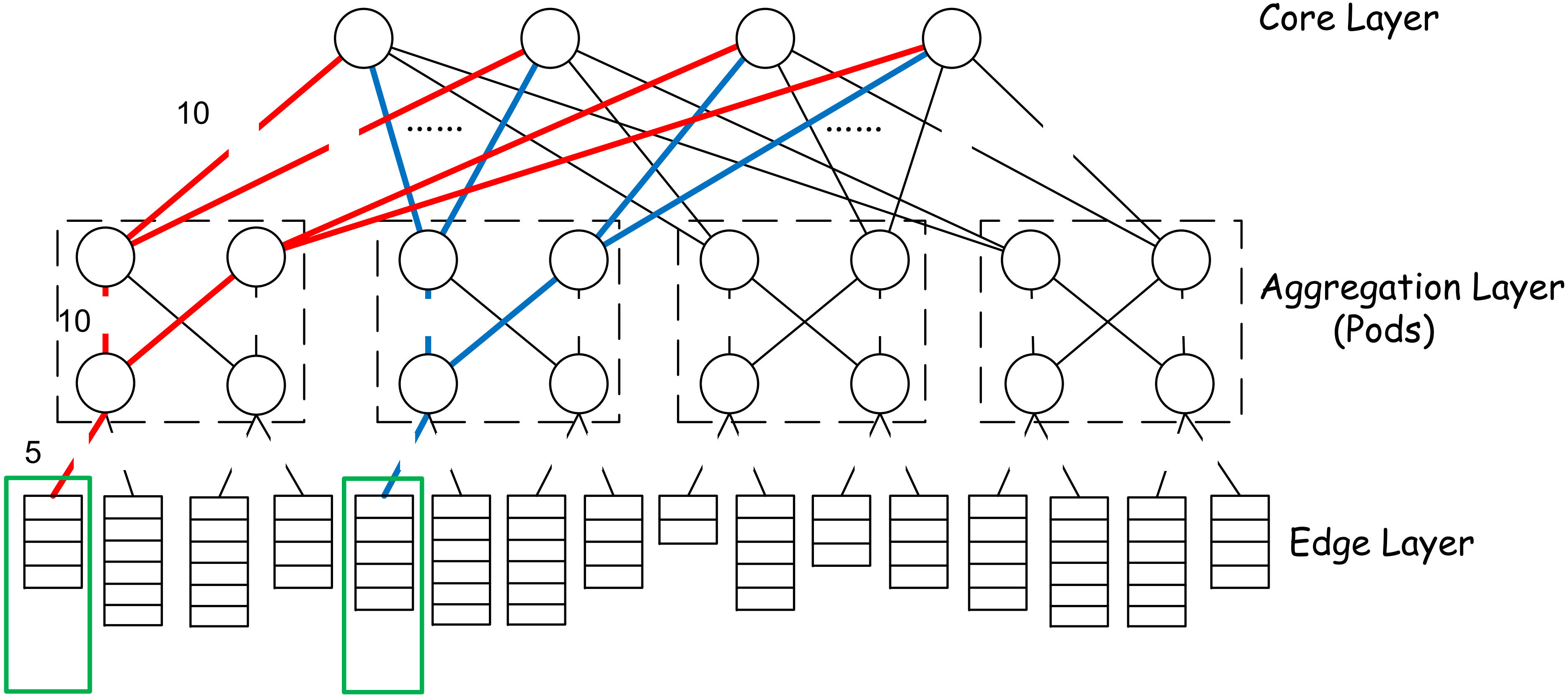
- Three comparison algorithms and PPCC
- Capacity ratio: average link capacity / node capacity



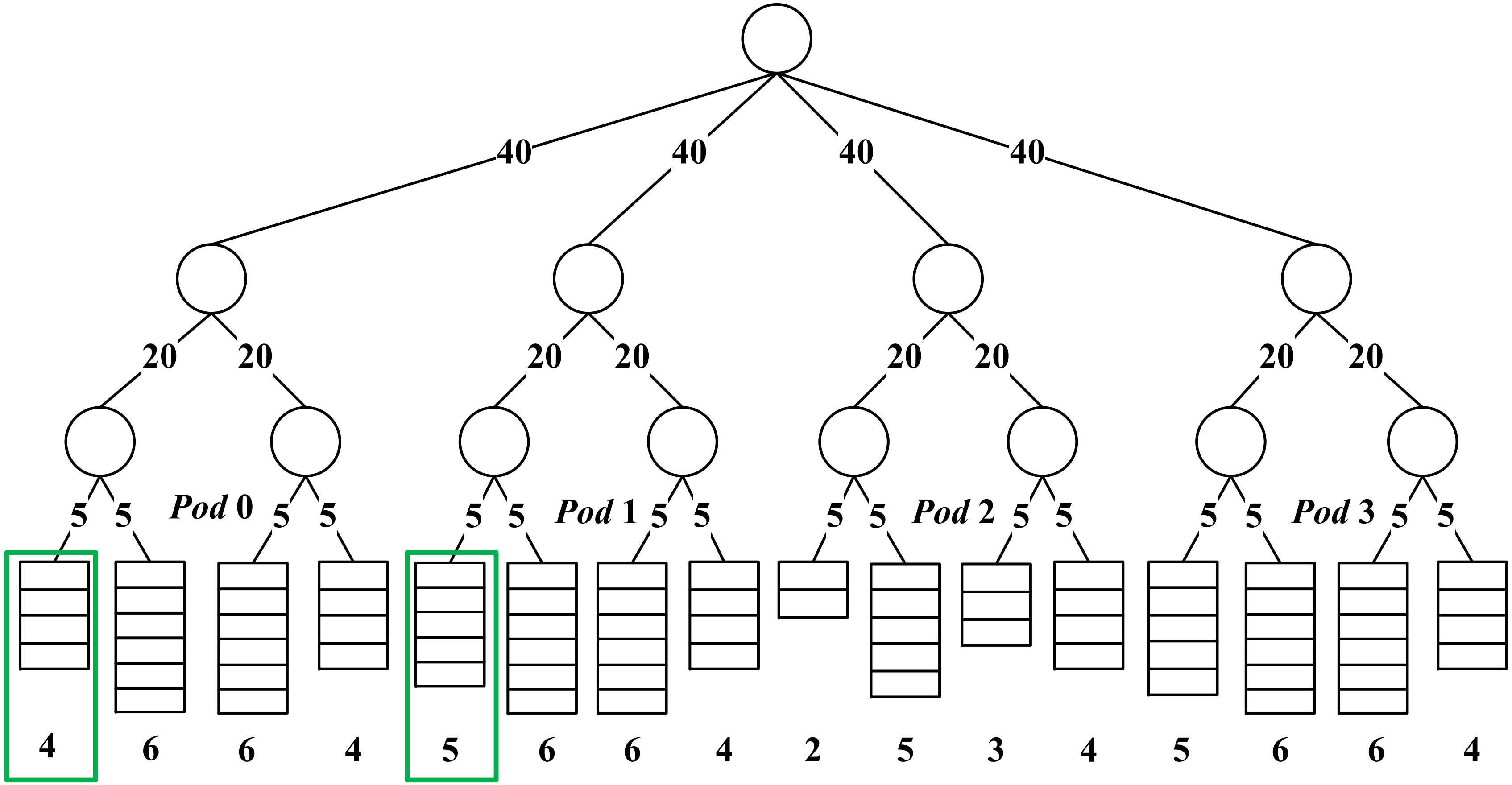
(a) k = 4 (b) k = 5 (c) k = 6

Fat Tree

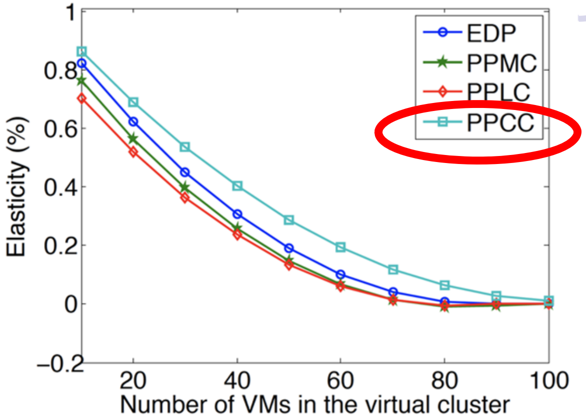
Equal-cost multi-path routing (ECMP) with m=4 (ports)



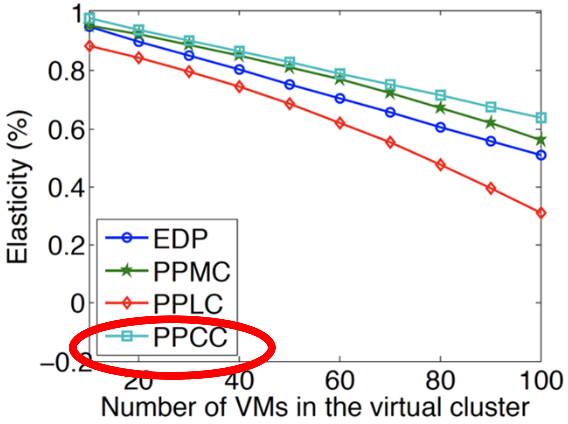
Fat Tree Equivalence



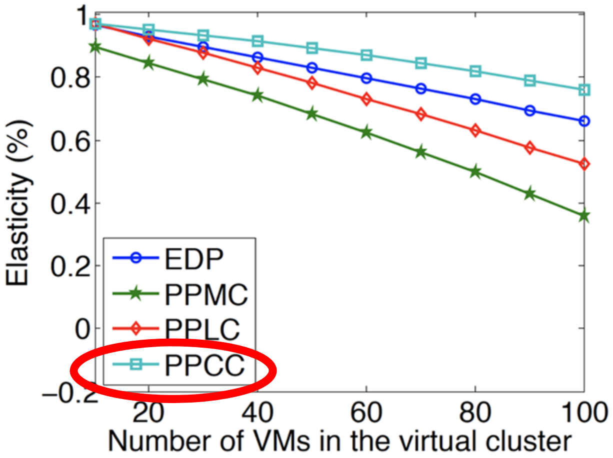
Fat Tree Simulation



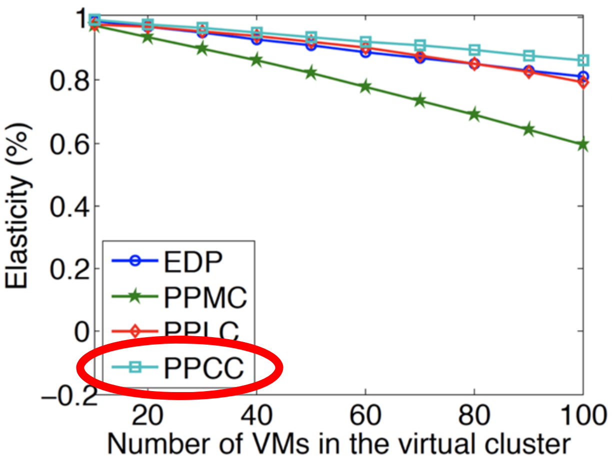
m=4



m=6



m=8

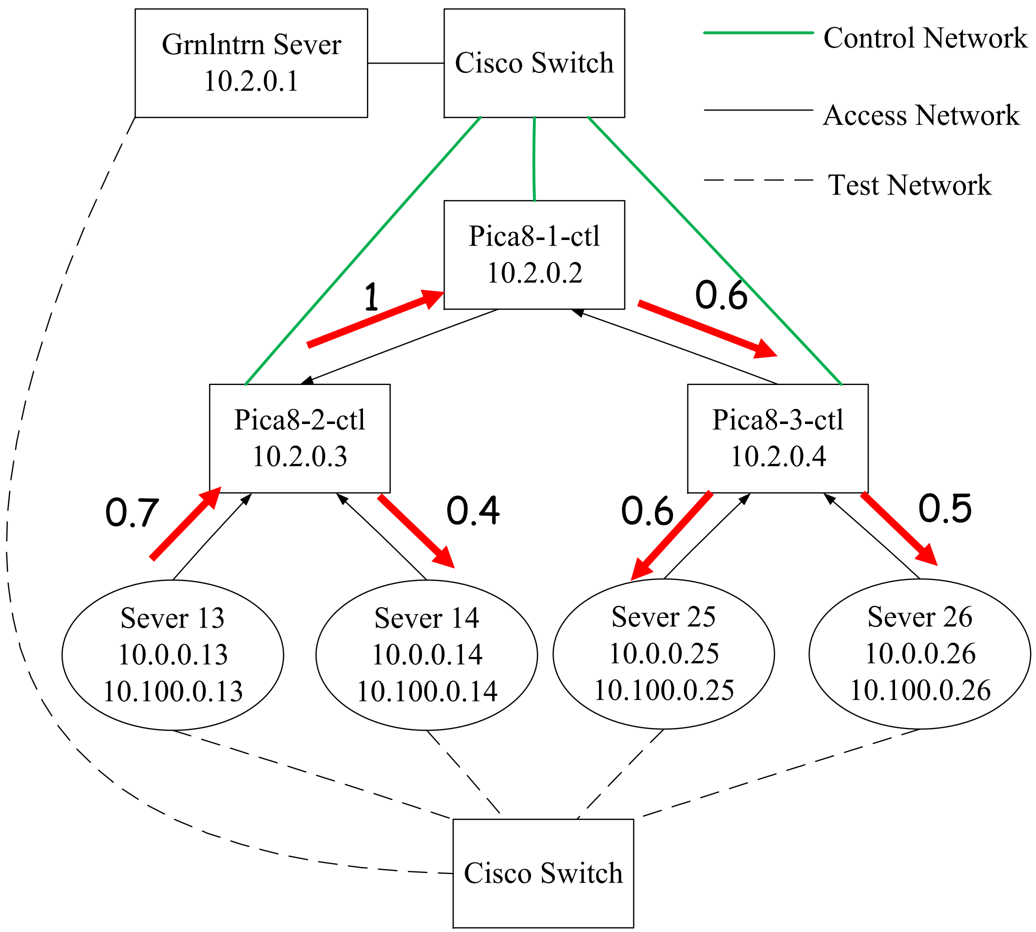


m=10

- Settings
  - m = 4, 6, 8, and 10
- Node capacity
  - PM: 0 to 100 slots
  - VM comm. bandwidth: 1 Gpbs

Link bandwidth
  - edge layer: [0, 10] Gbps
  - aggregation layer: [0, 15] Gbps
  - core layer: [0, 30] Gbps

Tree Testbed

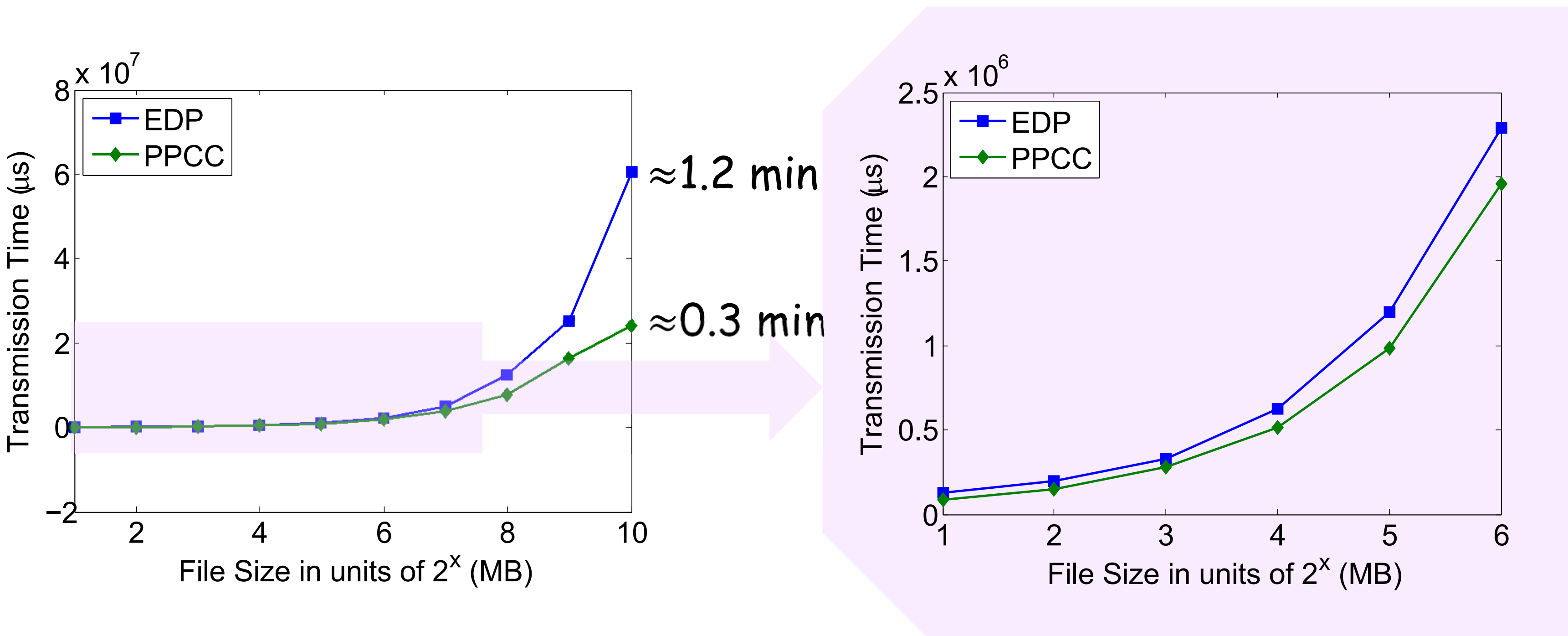


EDP	25%	25%	25%	25%
PPCC	41%	24%	19%	16%

- Central server: Grnlntrn
- Cisco switch: 8-port connector
- Pica8 switch: 48 ports
- Sever: Dell Power Edge R210 (2.4 GHz CPU, 4 GB memory)
- Maximum link capacity: 1 Gbps

Testbed Results

- One-to-all comm.
- Stress-test on a hose:
  Map (comp.), shuffle (scatter/gather comm.), and reduce (comp.)



6. Conclusions

- Models
  - Hose model on trees
- Elastic scheduling
  - Maximum admissible load (MAL)
  - Maximum elastic scheduling (MES)
- Future work
  - Other topologies
  - Applications: Hadoop and Spark

J. Wu, S. Lu, and H. Zheng, " On maximum elastic scheduling of virtual machines for cloud-based data center networks. " IEEE ICC, 2018.