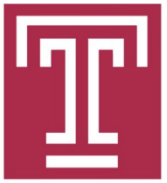
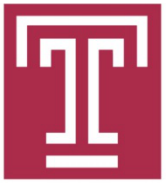


# Elasticity-aware Virtual Machine Placement in Cloud Data Centers

Kangkang Li , Jie Wu, and Adam Blaisse

Temple University, Philadelphia, PA, USA



# Outline

1. Introduction
2. Problem Formulation
3. Hierarchical VM Placement Algorithm
4. A Heterogeneous Scenario
5. Simulation
6. Conclusion

# Introduction

- The Elasicity of the VM
- Virutal Machine Placement in Data Centers
- Motivational Example

# The Elasticity of the VM

In this paper, we assume that VMs have identical machine resource demands (i.e. CPU) of $R$ and bandwidth demands of $B$. Due to various reasons (e.g. incremental tasks from users), the resource demands may fluctuate. If $R$ and $B$ increase to $R'$ and $B'$, then the growth ratios of $\frac{R'-R}{R}$ and $\frac{B'-B}{B}$ describe, respectively, to what extent the growth of machine and bandwidth demands could be satisfied. So we define the *machine / bandwidth elasticity* as the largest ratio that the machine / bandwidth demand of each VM could increase. Due

- For one VM, the elasticity of machine resources and bandwidth resources are not the same
- We choose the smaller one as the ***combinational*** elasticity of the VM

# Virtual Machine Placement in Cloud Data Centers

- Data Centers:

  ----PMs with idential VM slots

  ----links connected by a tree-topology structure

- VMs：

  ----machine resource demand for the PMs

  ---- bandwidth resource demand for ***communication*** on the links

- The VM placement in a Data Center should consider both dimensions resources in machine slots and bandwidth

# Virtual Machine Placement in Cloud Data Centers

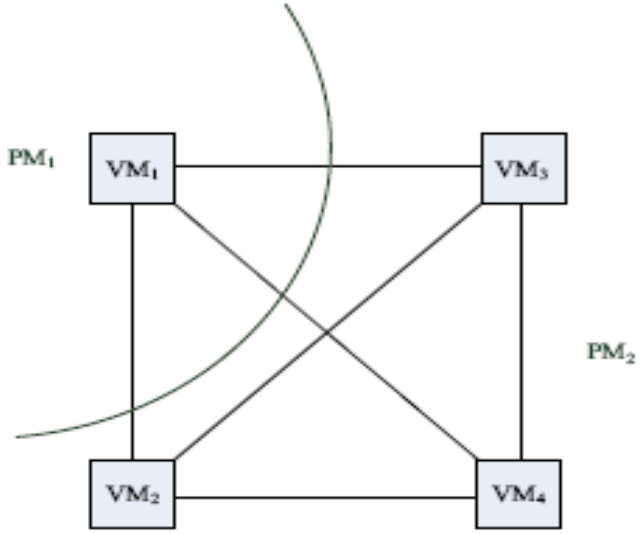
- Communication Model: hose model

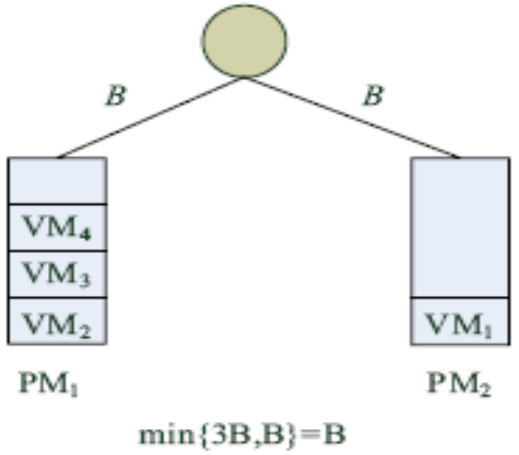
  --- PM1 can use at most B bandwidth, since only VM1 is located inside it

  ---VMs placed outside PM1 (three VMs) can use at most 3B bandwidth

  ---The bandwidth desired by PM1 is limited to both the VMs located inside and outside PM1, i.e., min{B, 3B} = B.

- Therefore, the bandwidth desired by a PM is equivalent to the minimum bandwidth demands of the VMs located inside and outside it.



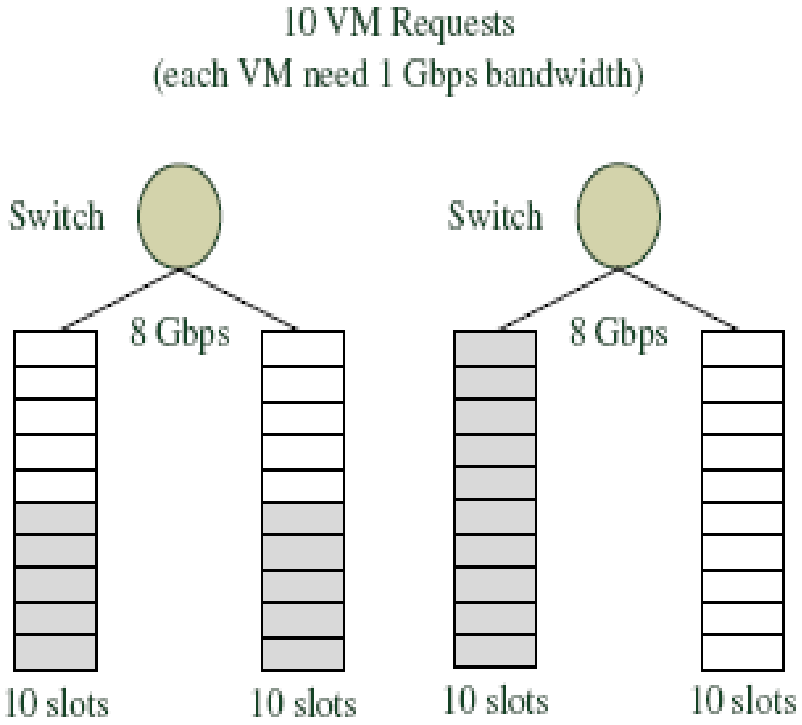
# Motivational Example

- Objective: maximize the combinational elasticity of input VMs

- Left subfigure:
  machine elasticity: 100%
  bandwidth elasticity： 60%

- Right subfigure:
  machine elasticity: 0%
  bandwidth elasticity: infinite

***conflict*** on the optimization of both machine and bandwidth elasticity



# Outline

1. Introduction
2. Problem Formulation
3. Hierarchical VM Placement Algorithm
4. A Heterogeneous Scenario
5. Simulation
6. Conclusion

# Problem Formulation

Data center：
Semi-homogeneous configuration
Each PM has an indentical capacity： C
Each link of the same layer has the same bandwidth capacity
The upper layer link capacity is larger than the lower layer in reducing upper layer congestion.

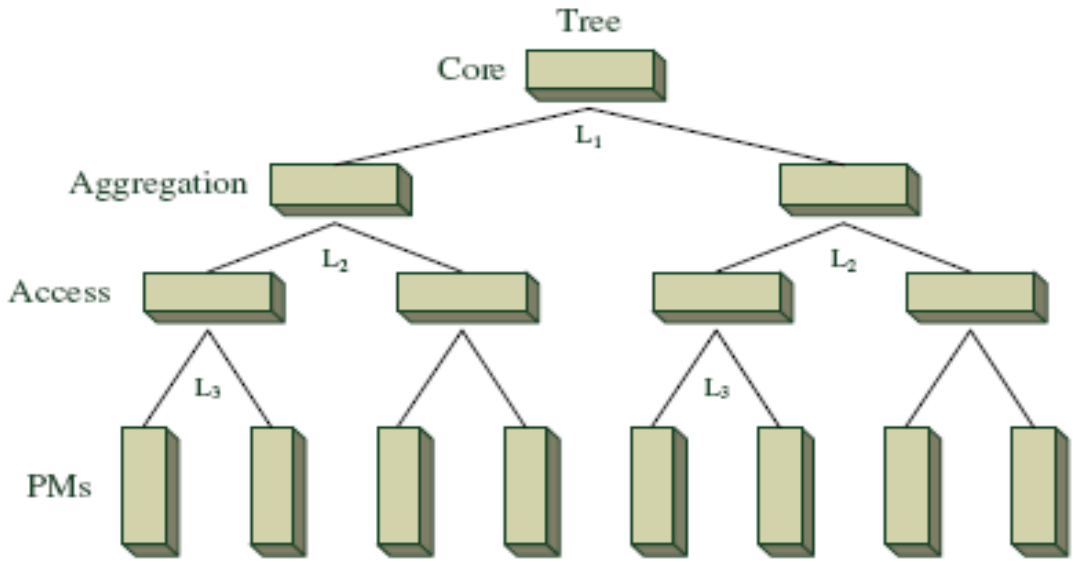



Fig. 3. Tree-based network topology

# Problem Formulation

Objective:

maximize the achievable combinational elasticity among all input VMs

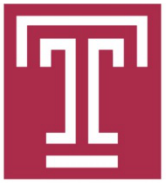
$$Maximize \ \min_i\{E_i\}$$

where E is

$$E = \min\{\min_r\{\frac{C}{R * m_r}\}, \min_l\{\frac{L}{B * \min\{m_l, N - m_l\}}\}\}$$

This problem is equivalent to minimizing the combinational utilization of the data centers.

$$U = \max\{\max_r\{m_r \frac{R}{C}\}, \max_l\{\min\{m_l, N - m_l\}\frac{B}{L}\}\}$$



# Outline

1. Introduction
2. Problem Formulation
3. Hierarchical VM Placement Algorithm
4. A Heterogeneous Scenario
5. Simulation
6. Conclusion

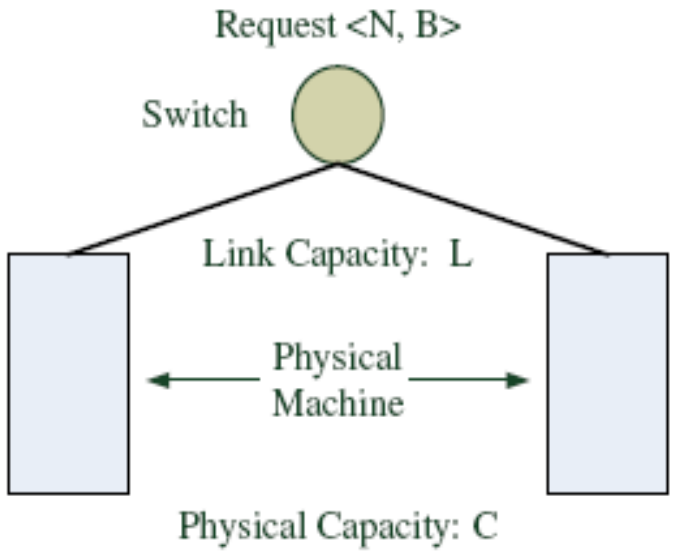
# Hierarchical VM Placement Algorithm

One-layer schedulability:

$$N^* = min\{N, 2C, C + \frac{L}{B}\}$$

One-layer optimality:

$$U(x) = \max \{x\frac{B}{L}, \frac{N-x}{C}\}$$

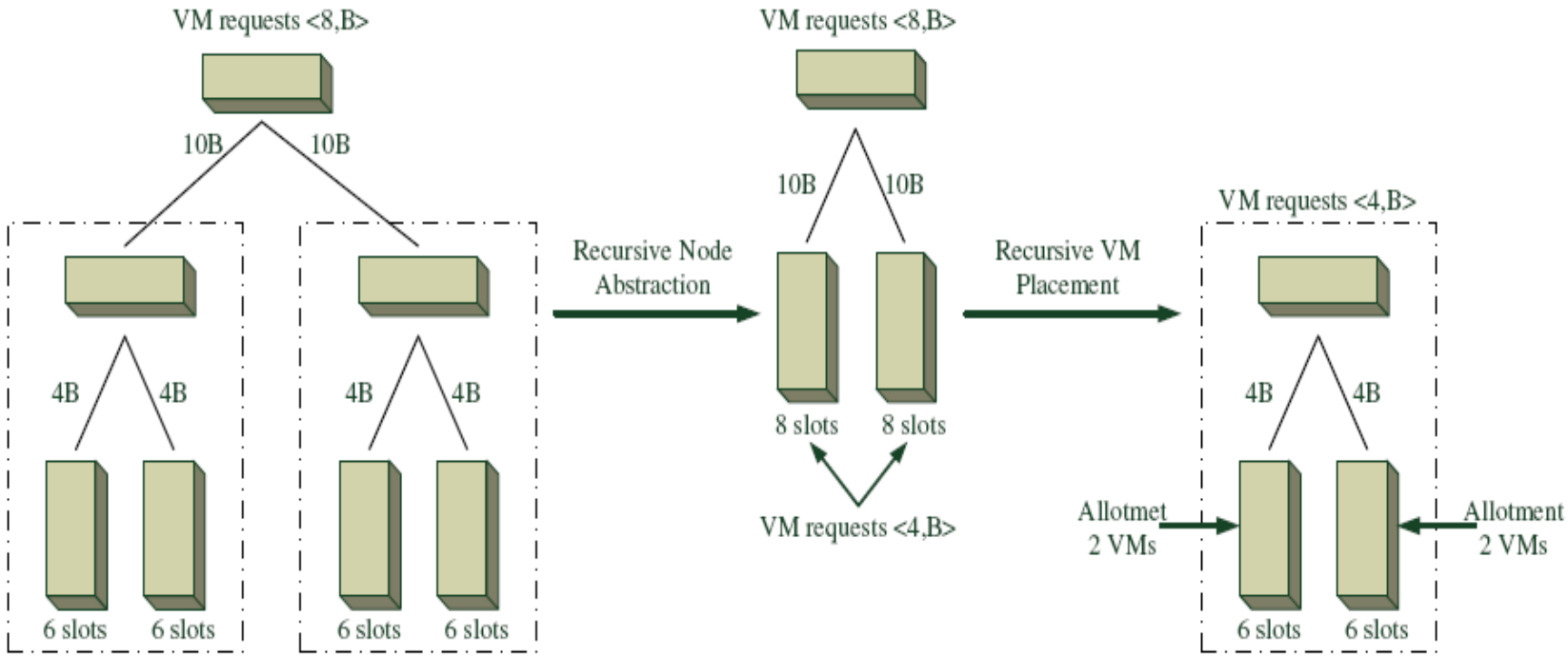


# Hierarchical VM Placement Algorithm

Multi-layer Cluster

----Binary abstraction (bottom to top)

----Recursive VM placement (top to bottom)




The time complexity is O(M), given M machines at the bottom

# Outline

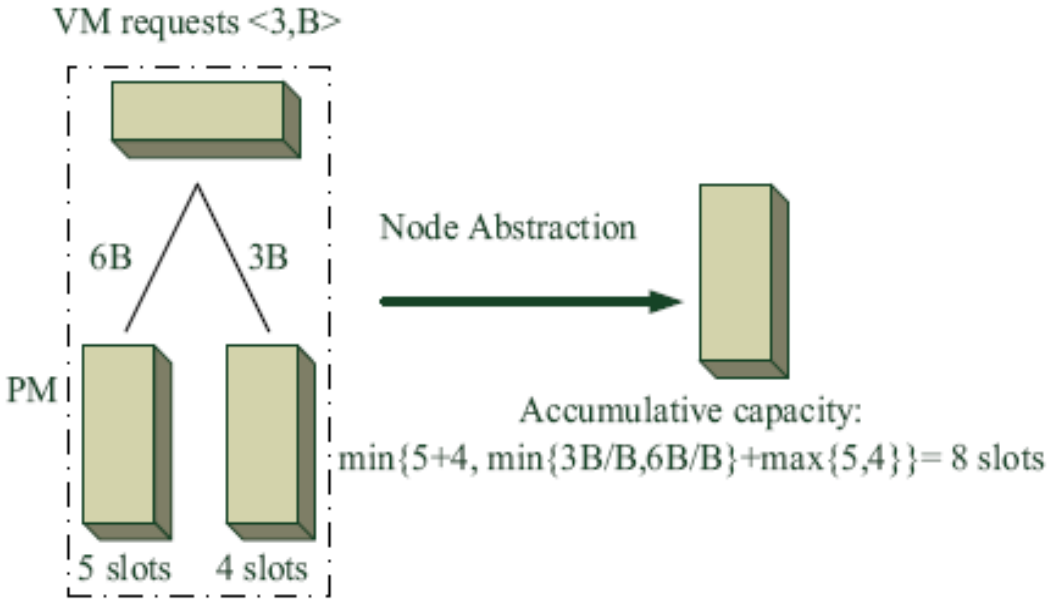
1. Introduction
2. Problem Formulation
3. Hierarchical VM Placement Algorithm
4. A Heterogeneous Scenario
5. Simulation
6. Conclusion

# A Heterogeneous Scenario

Motivation:

Today's datacenters can support multiple tenants' requests. The VM requests of different tenants may come at different times. After one tenant's VMs are placed into the datacenter, all the links and PMs capacities will change, making the datacenter a heterogeneous configuration.

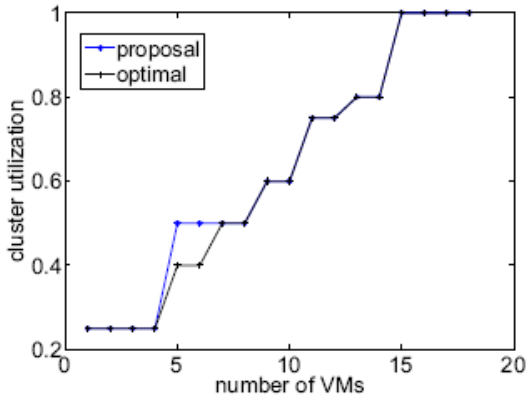
Two steps:
----One-layer optimality
----Mutli-layer binary abstraction and resursive VM placement



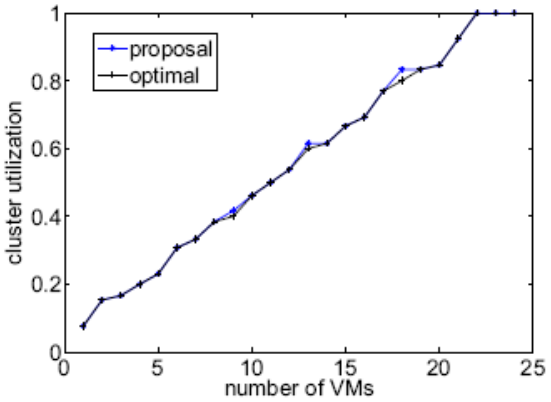
# Outline

1. Introduction
2. Problem Formulation
3. Hierarchical VM Placement Algorithm
4. A Heterogeneous Scenario
5. Simulation
6. Conclusion

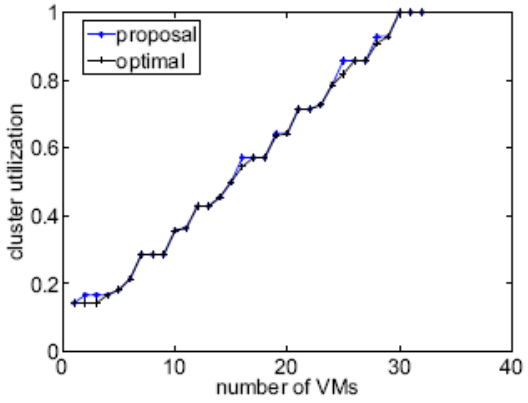
# Simulations



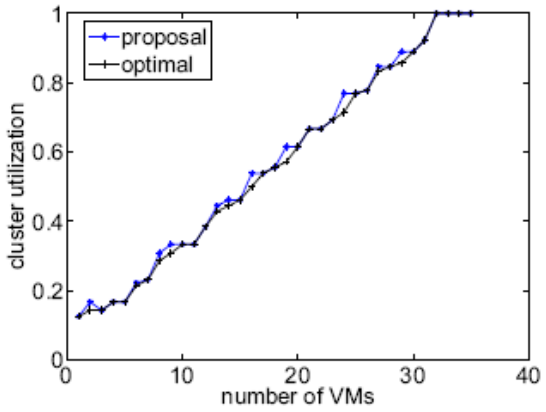

(a) bottom-layer links capacities: 2 Gbps



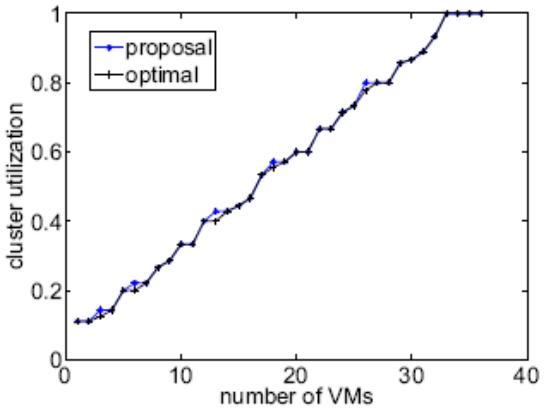

(b) bottom-layer links capacities: 4 Gbps



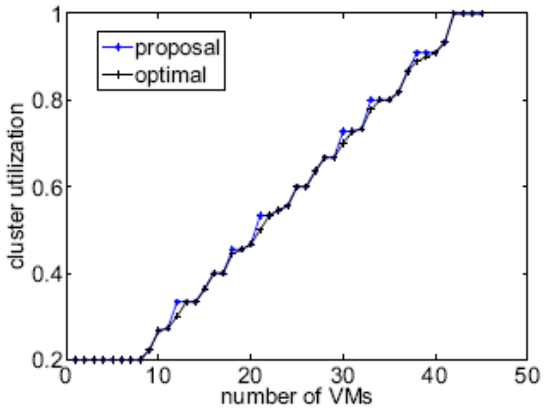

(c) bottom-layer links capacities: 6 Gbps




(a) link capacity range: [5,10] Gbps




(b) link capacity range: [5,15] Gbps




(c) link capacity range: [5,20] Gbps

# Conclusions

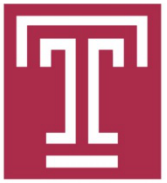
We propose the concept of VM's elasticity to meet the on-demand scaling requirement in cloud computing.

We propose the hierarchical VM placement algorithm to maximize the elasticity of the input VMs.

We study the scheduability of the input VMs and also prove the optimality of our algorithm under a frequently used datacenter configuration.

We also conduct a study on the heterogeneous scenario to meet the requirements of a multi-tenant datacenter.

The evaluation results show the high efficiency of our algorithm



# Thank you!

## Questions?

kang.kang.li@temple.edu