Efficiency and Resilience of Resource Allocation for Next-Generation Data Centers

Ph.D. Candidate: Chao GUO

Supervisor: Moshe ZUKERMAN

Other collaborators: Gangxiang Shen (Soochow University, China), Sanjay K. Bose (Plaksha University, India), Xinyu Wang (CityU), Tianjiao Wang (CityU), Jiahe Xu (CityU)

Department of Electrical Engineering
香港城市大學
City University of Hong Kong

Outline

- Background
  - Data center (DC), data center virtualization, resource disaggregation
- Study 1: Resource allocation for *VDCs* considering *hot spots issues*
- Study 2: Reliable resource allocation for *DDCs*
- Study 3: Reliable resource allocation for *DDCs* with network effects
- Conclusion

Data Center (DC): Center of Data

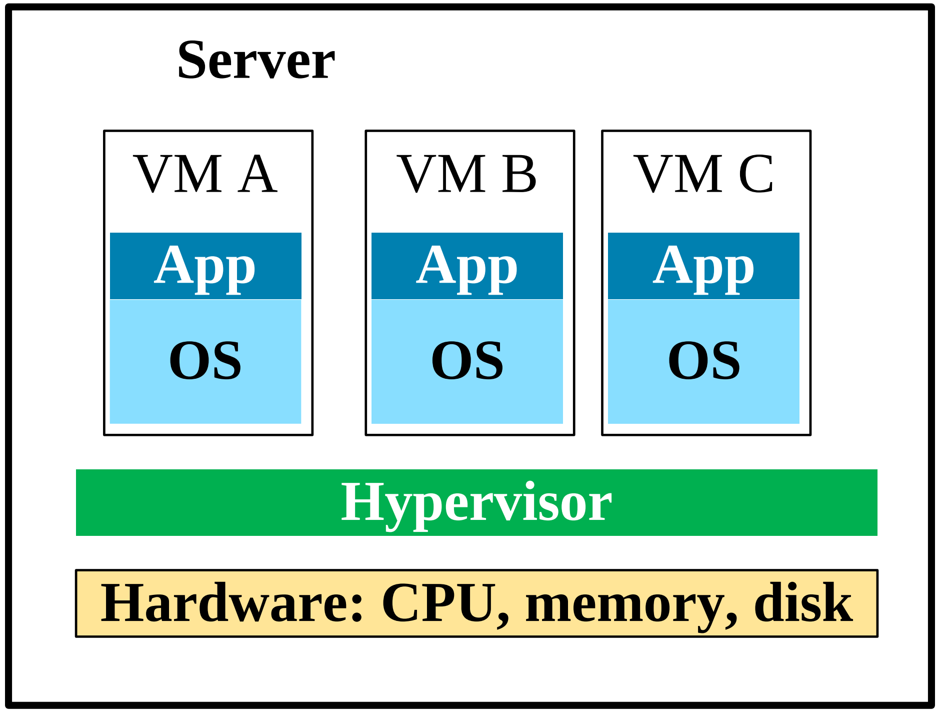
- IT infrastructure & Power & Cooling

Top of Rack (ToR) Switch

Server

Credit: https://www.istockphoto.com/hk/%E5%9C%96%E7%89%87/datacenter

Data Center Virtualization

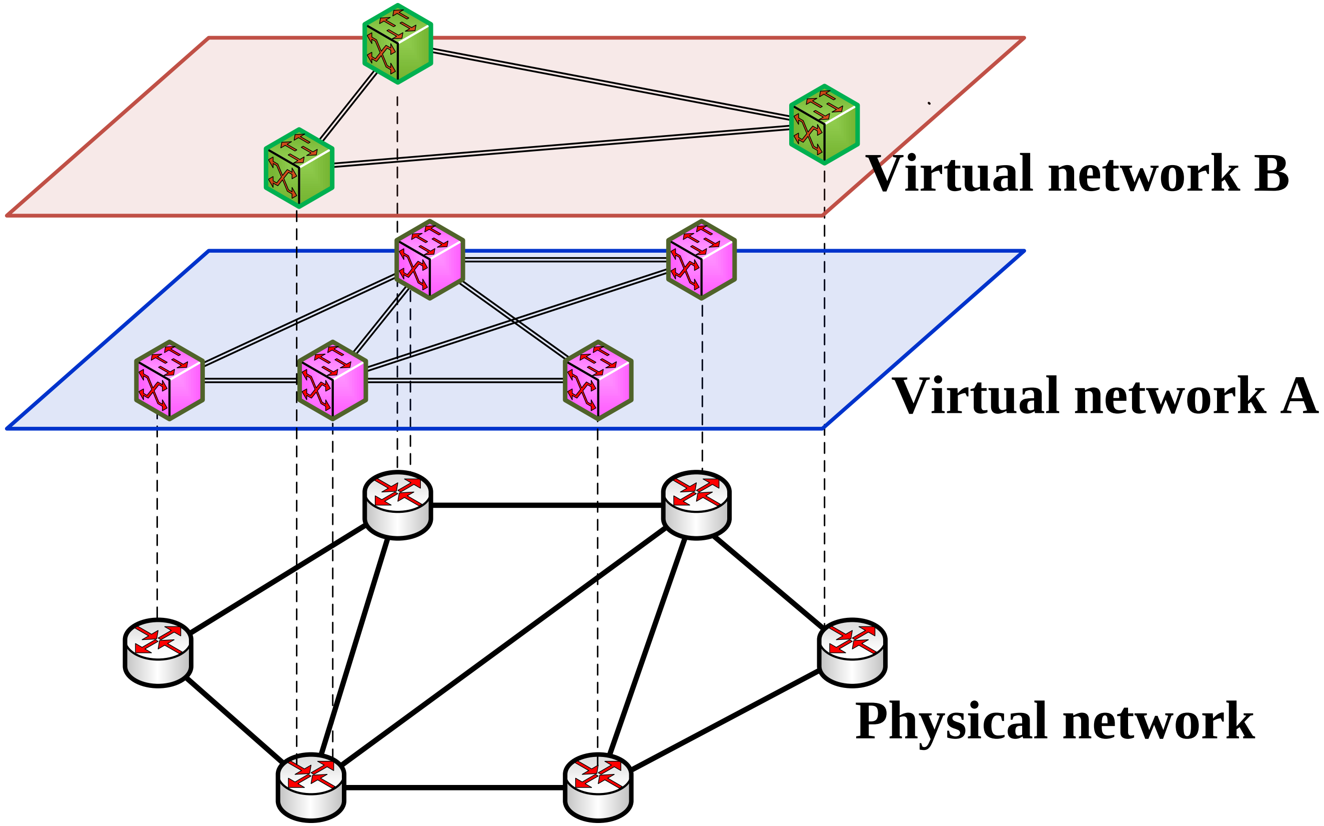


Server

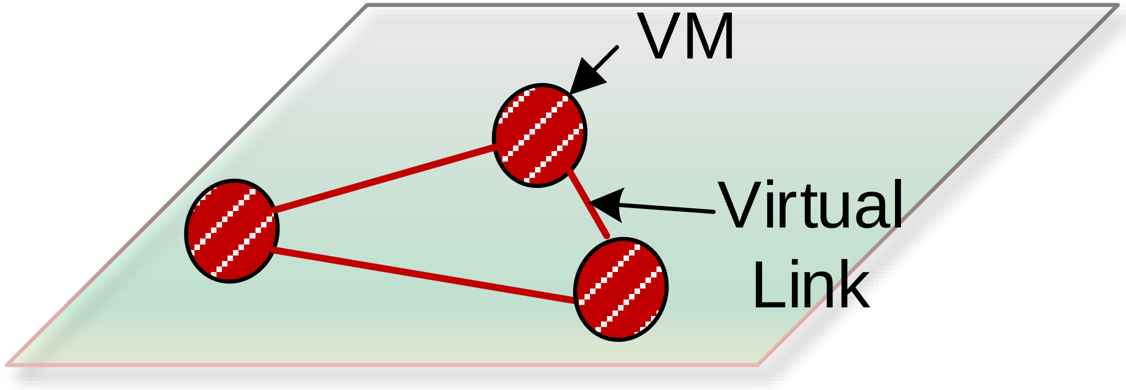
VM A	VM B	VM C
App	App	App
OS	OS	OS

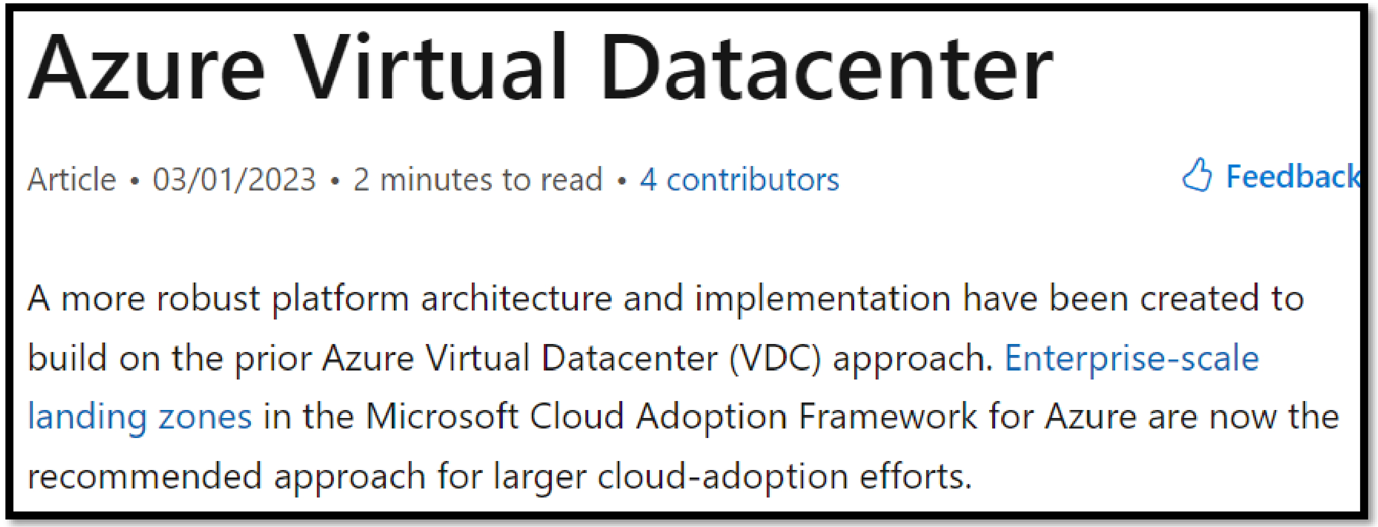
Hypervisor

Hardware: CPU, memory, disk



Virtual Data Center (VDC)





Azure Virtual Datacenter

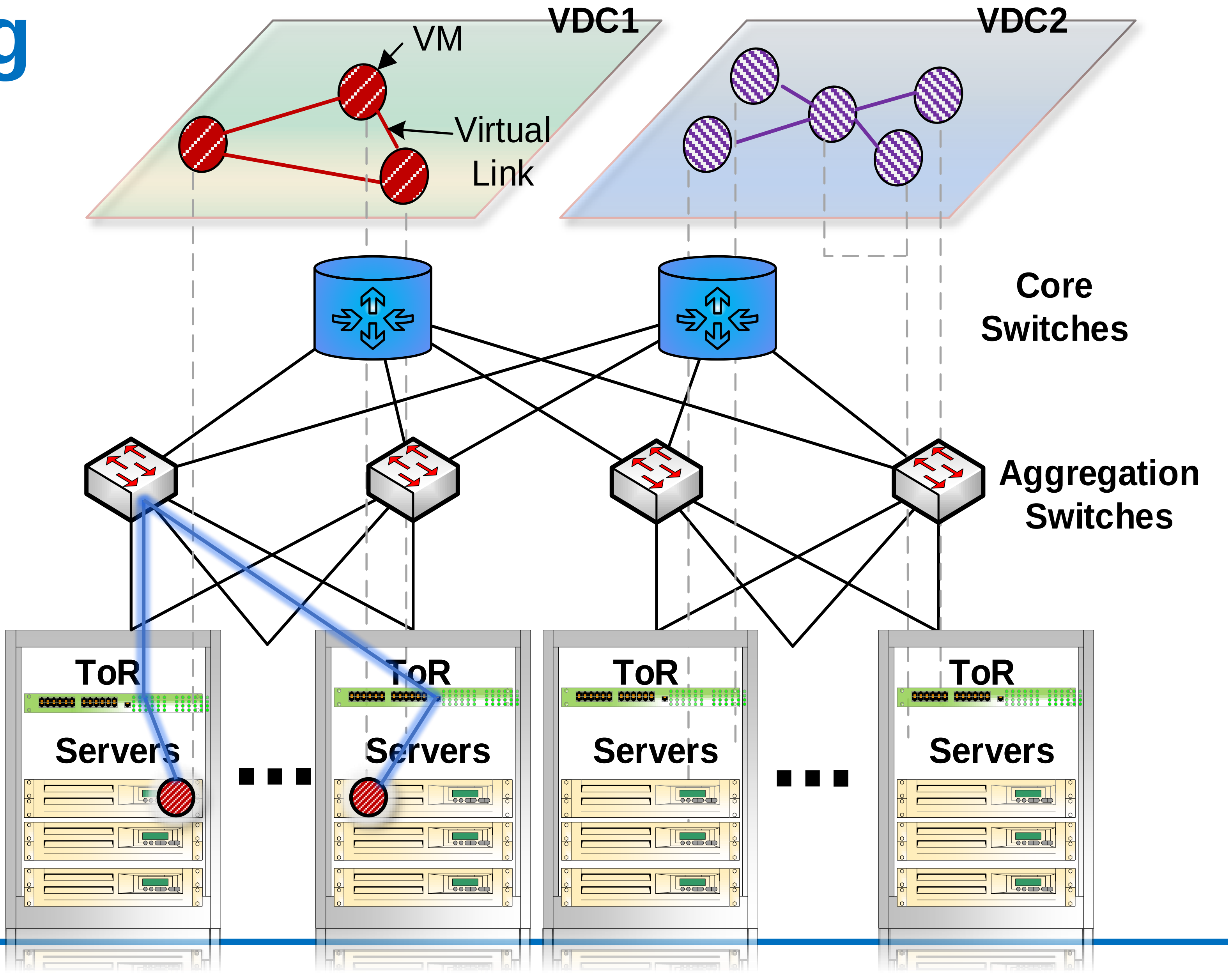
Article • 03/01/2023 • 2 minutes to read • 4 contributors

Feedback

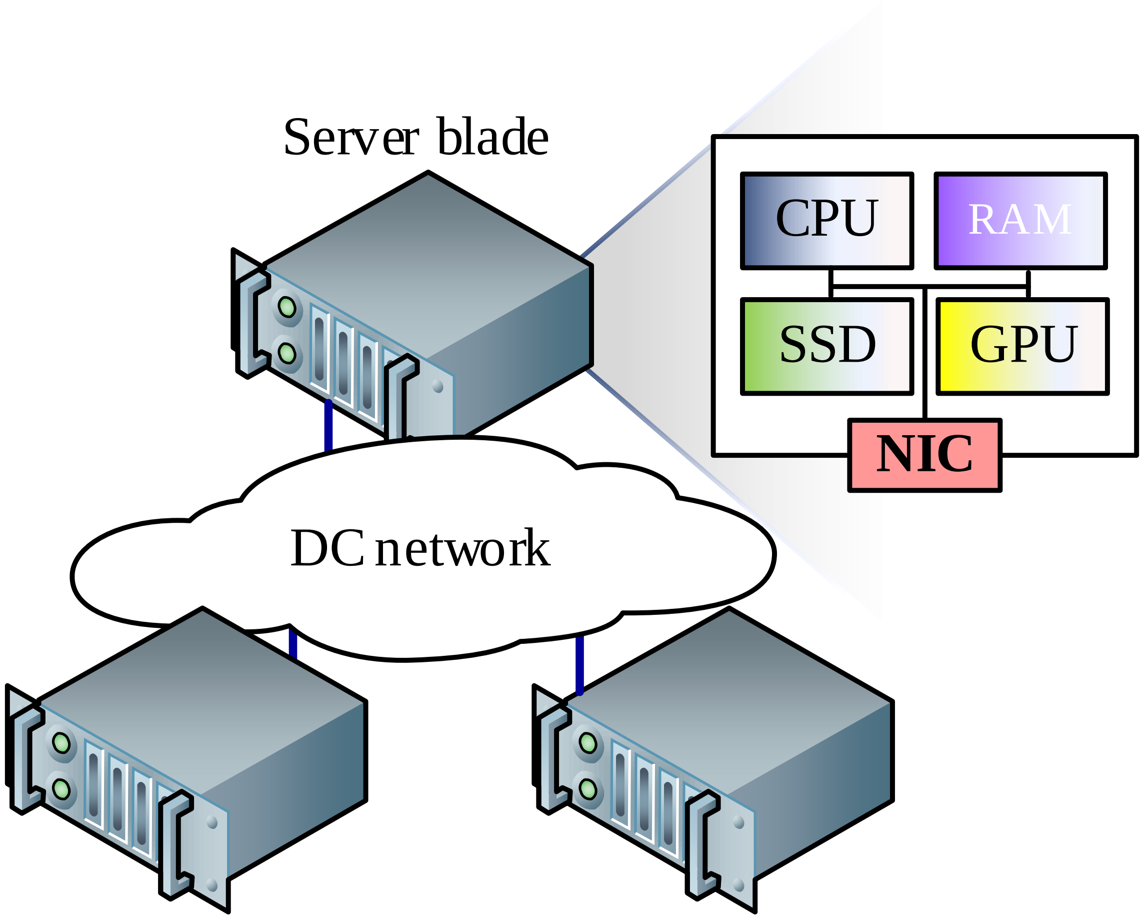
A more robust platform architecture and implementation have been created to build on the prior Azure Virtual Datacenter (VDC) approach. Enterprise-scale landing zones in the Microsoft Cloud Adoption Framework for Azure are now the recommended approach for larger cloud-adoption efforts.

VDC Embedding

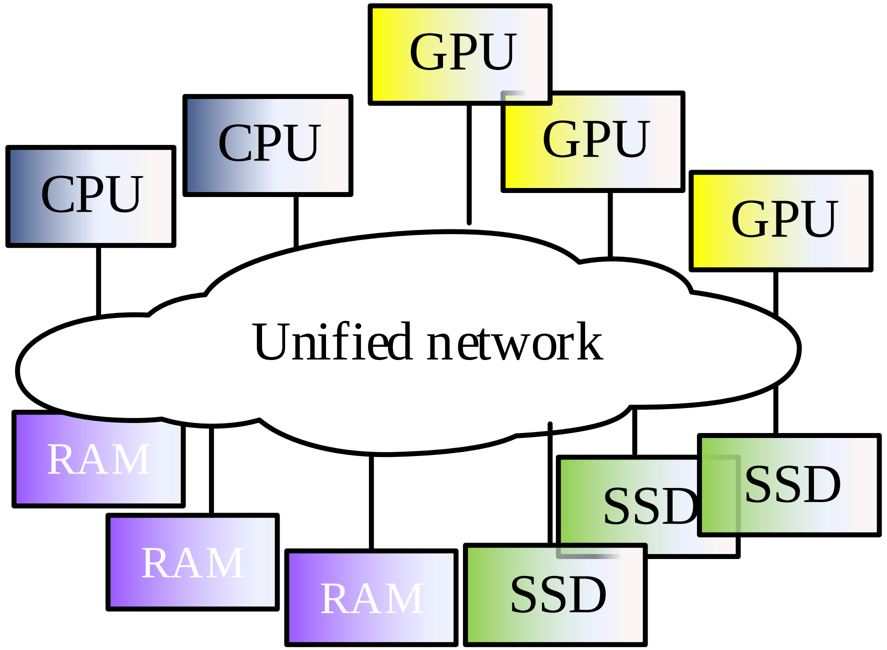
- VM mapping
- Virtual link mapping



Resource Disaggregation: For Resource Pooling and Composability



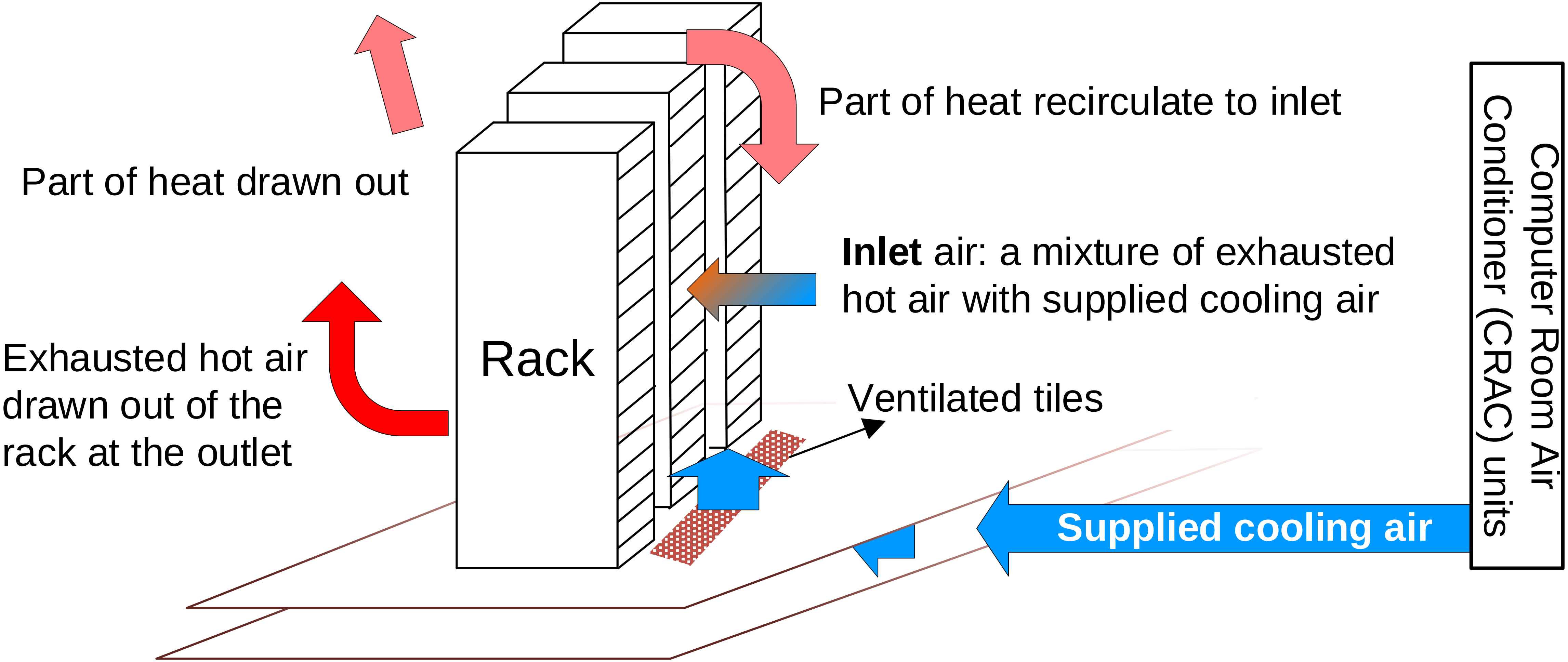
(a) Server-based data center (SDC)



(b) Disaggregated data center (DDC)

Study 1: Temperature-Aware VDC Embedding

Thermal Fluid Cycle



Part of heat drawn out

Part of heat recirculate to inlet

Inlet air: a mixture of exhausted hot air with supplied cooling air

Exhausted hot air drawn out of the rack at the outlet

Rack

Ventilated tiles

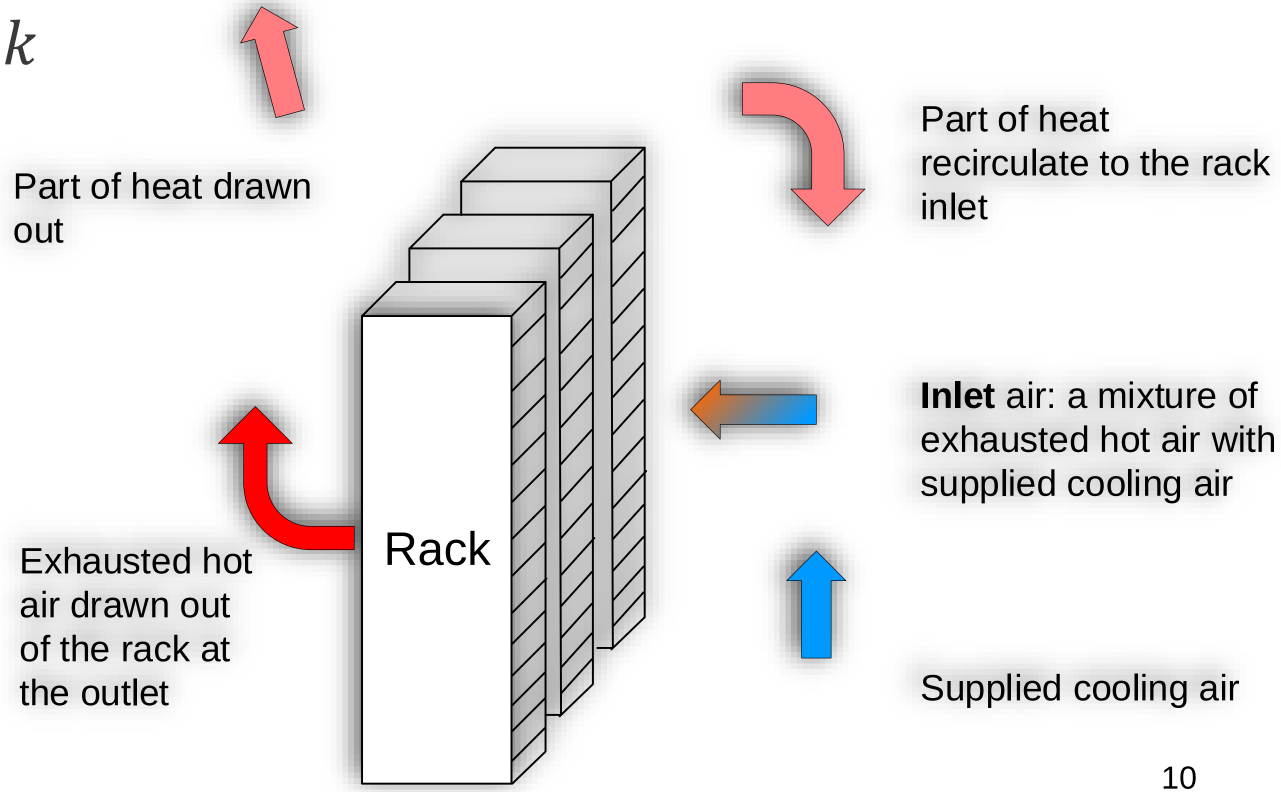
Supplied cooling air

Computer Room Air Conditioner (CRAC) units

Rack-Level Inlet Temperature Model

$$T_k^{in} = T_k^{sup} + \sum_{l \in \Phi} d_{kl} \cdot P_l^{rack}$$

- T_k^{in} - inlet temperature of rack k
- T_k^{sup} - cooling temperature supplied to k
- P_l^{rack} - total power of rack l
- d_{kl} - *heat transfer matrix*:
  increase rate of rack k's inlet temperature caused by P_l^{rack}



Temperature-Aware VDC Embedding Problem

- Given: Physical DCN, T_k^{sup} , d_{kl} ; VDC requests
- Objective

$$\text{Minimize: } T_{max}^{in} + \alpha \cdot \sum_{n \in \boldsymbol{N} \cup \boldsymbol{S}} P_n$$

Maximum rack inlet
temperature of all racks

total power of all IT equipment

- Solution 1: Mixed integer linear programming (Chapter 3.4)

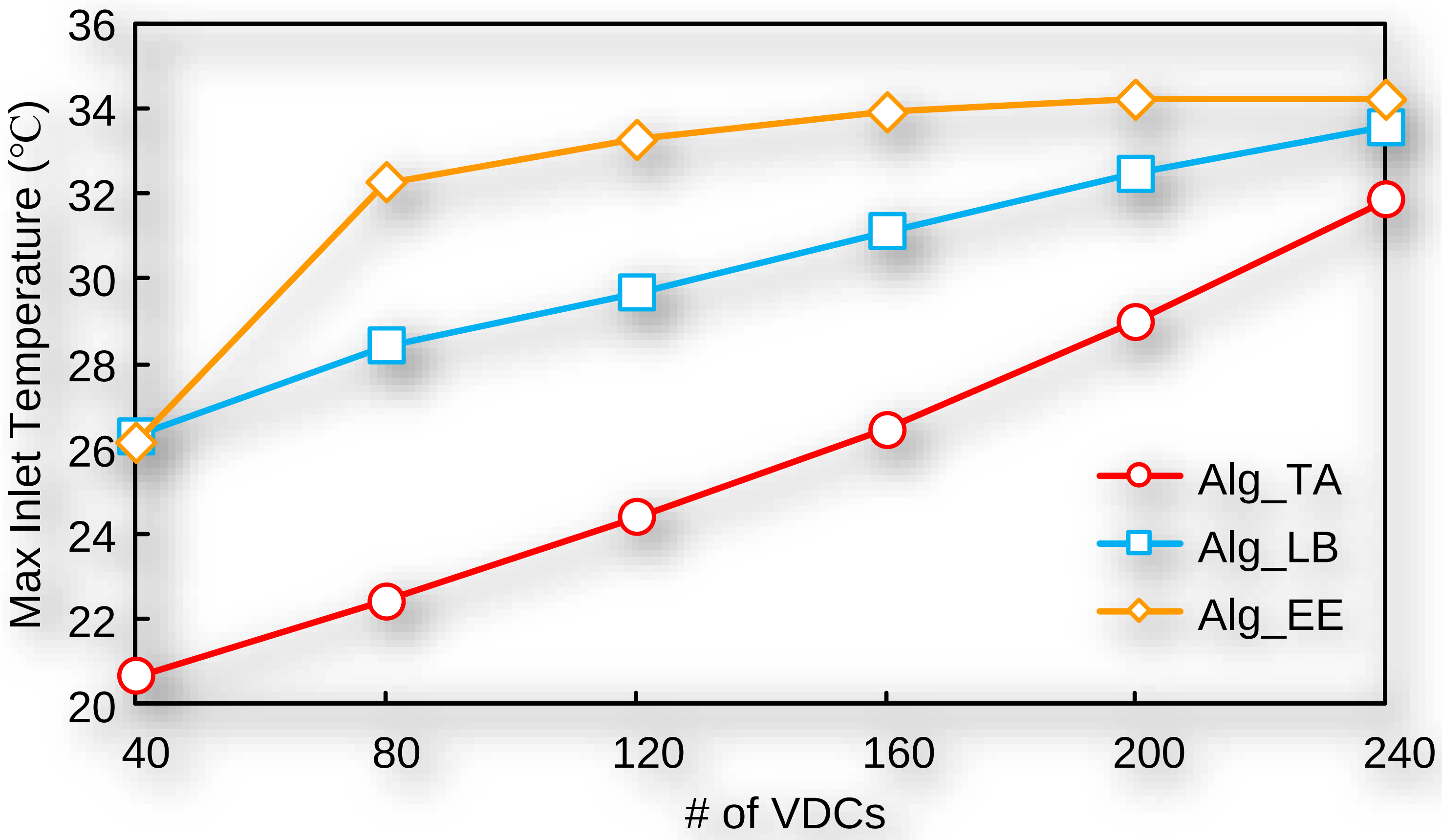
Heuristic Method (Chapter 3.5)

1. Place more workloads to colder racks while less to hotter racks
    - Inlet temperature can be well balanced
    - Failure risk can be mitigated, and cooling energy can be well saved

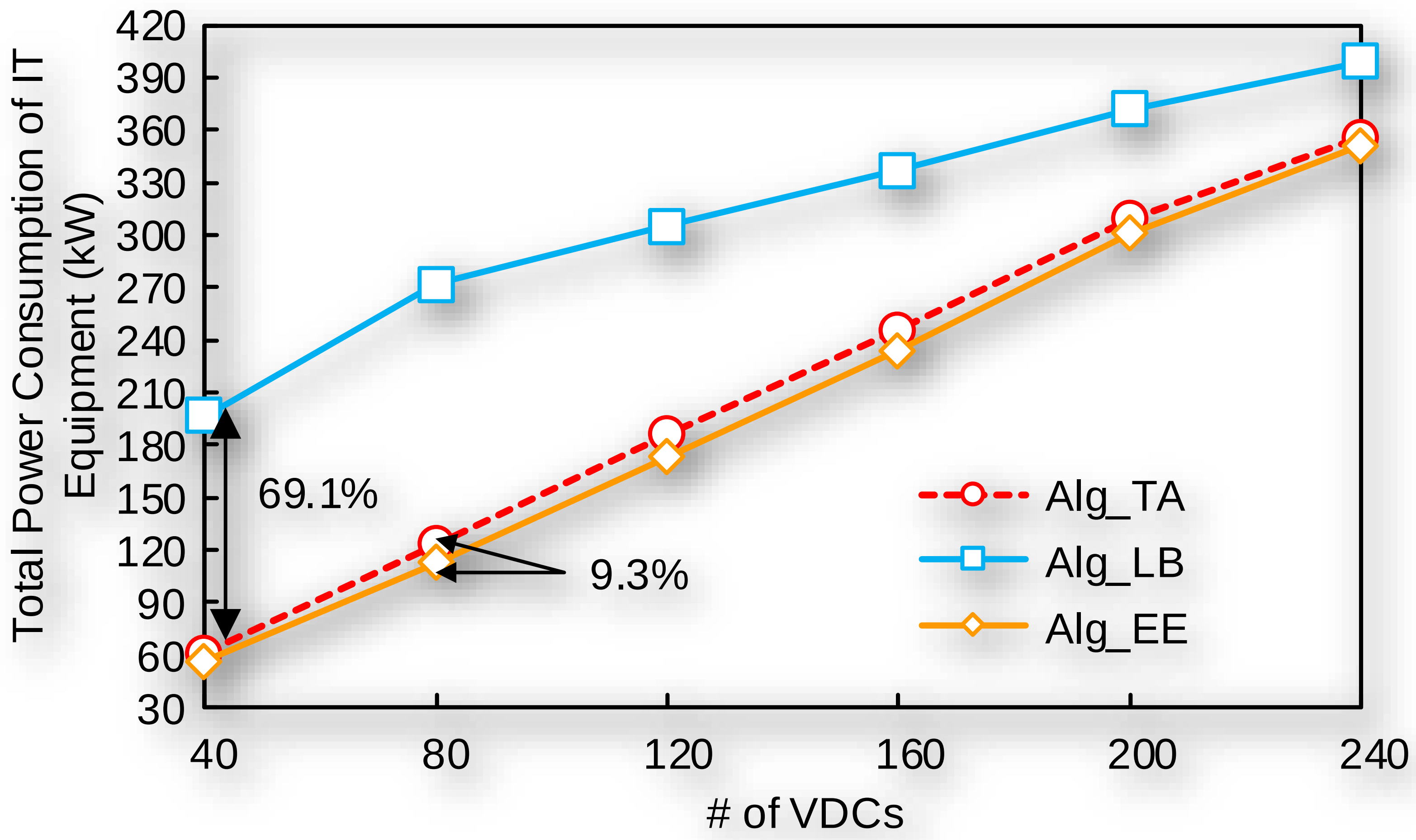
2. Consolidate workloads in each rack to fewest devices
    - Energy consumption of IT equipment keeps in low level

Maximum Rack Inlet Temperature

- ◆Alg_TA: temperature-aware
- ◆Alg_LB: load-balanced
- ◆Alg_EE: IT-only energy-efficient



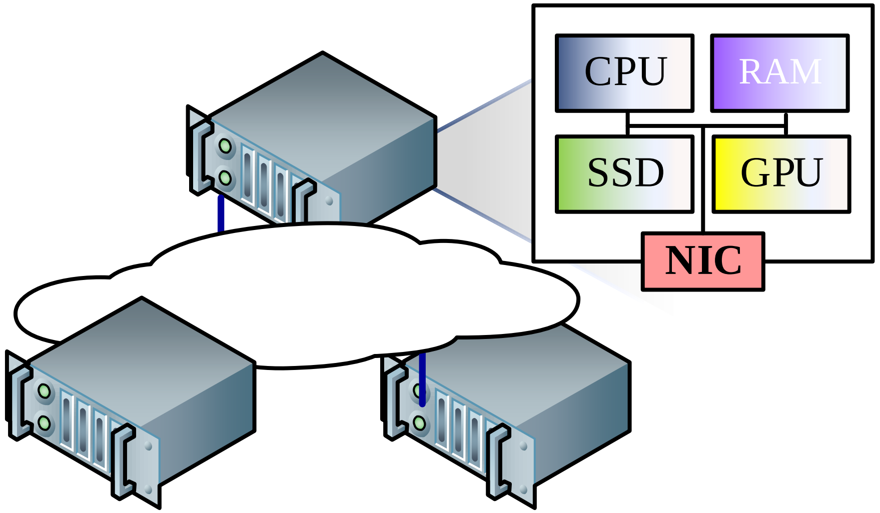
Total Power Consumption of IT Equipment



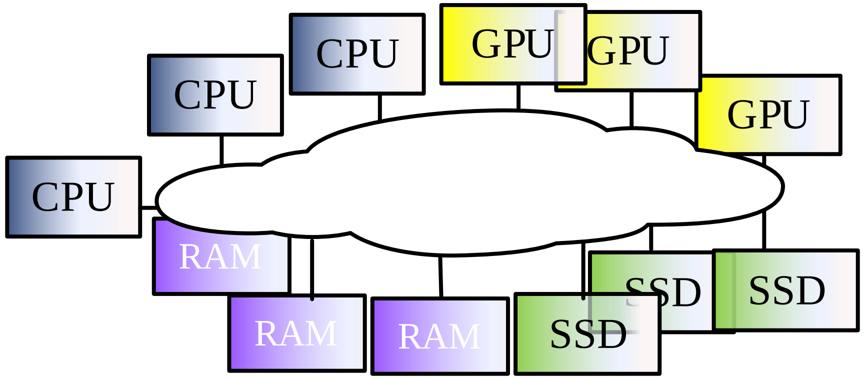
Study 2: Exploring the Benefits of Resource Disaggregation in Service Reliability

Reliability Benefits of Disaggregation

- High flexibility
  - Expand optimization regions
- New failure pattern
  - Different modules fail more independently



(a) Server-based architecture (SDC)



(b) Disaggregated architecture (DDC)

Reliability-Aware Resource Allocation for DDCs

- Input: Hardware (Capacity and reliability) and requests (resource demand, reliability requirement)
- Objective:

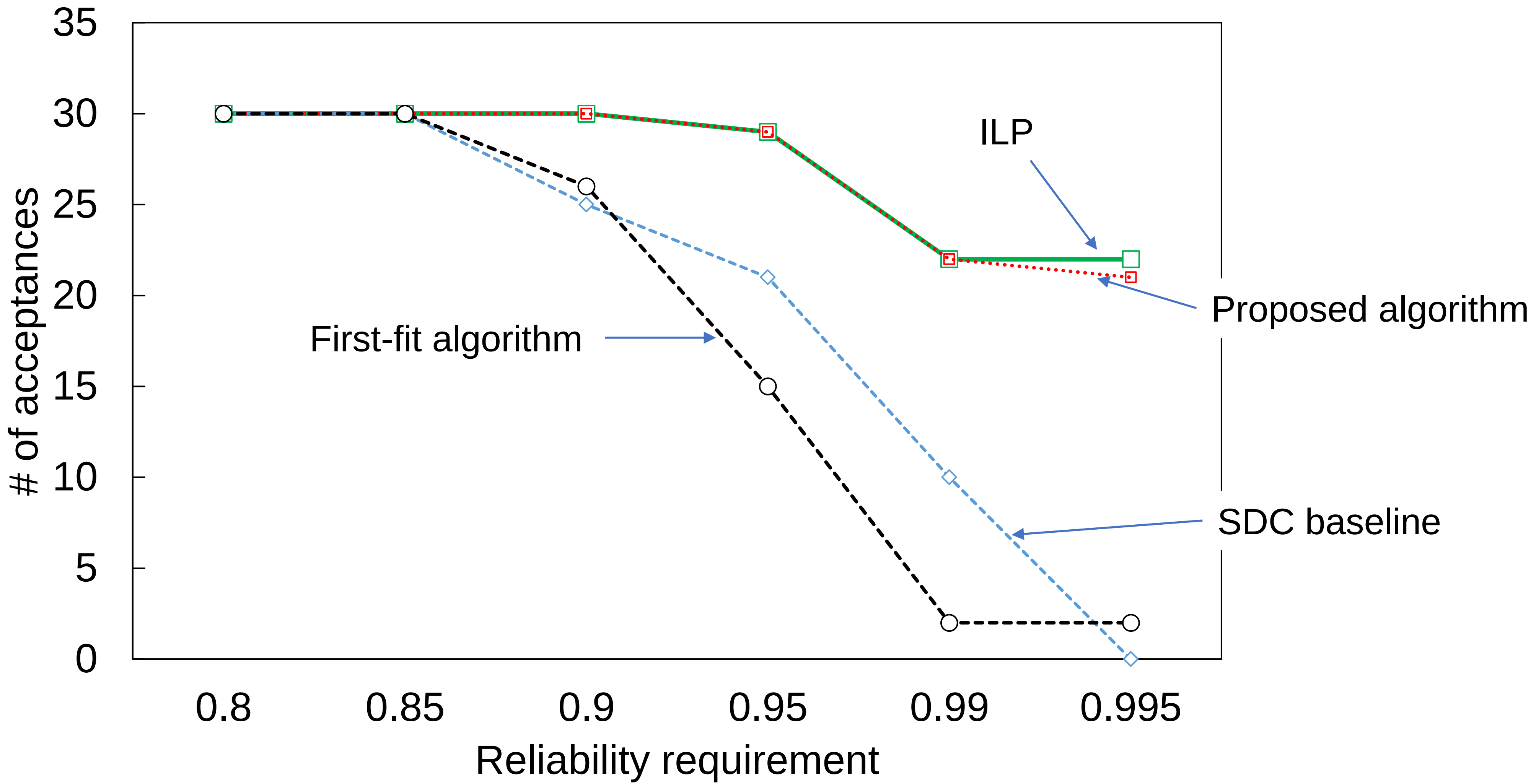
$$\max\left\{\underbrace{\sum_{i\in\boldsymbol{I}}\omega_i}_{\text{Number of accepted requests}}-\epsilon\cdot\underbrace{\sum_{i\in\boldsymbol{I}}\chi_i}_{\text{Number of requests provisioned with backups}}\right\}$$

- Constraint: Each request is provisioned with at most one backup
- Solution 1: ILP (Chapter 4.3)

Heuristic Method (Chapter 4.4)

- Heuristic method (Detailed in Chapter 4.4)
  - First try to satisfy the reliability requirement without backup
  - Try to satisfy the reliability requirement with backup if without backup cannot meet the requirement
  - Try to allocate modules to each request that is least reliable but can satisfy the requirement.

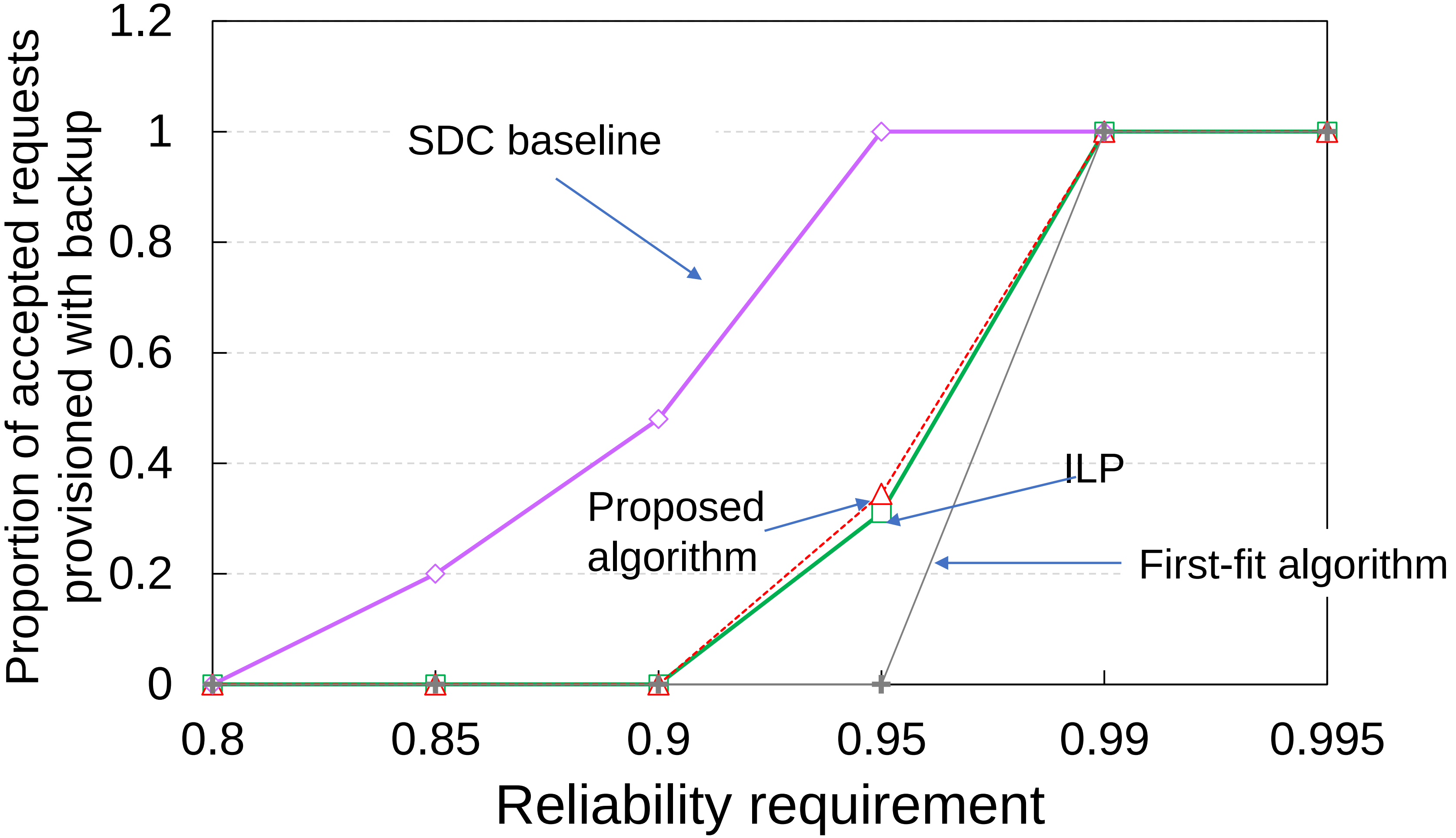
Number of accepted requests vs. reliability requirements



Proportion of Accepted Requests Provisioned with Backup Resources

It is more efficient to meet reliability requirement with no redundancy.

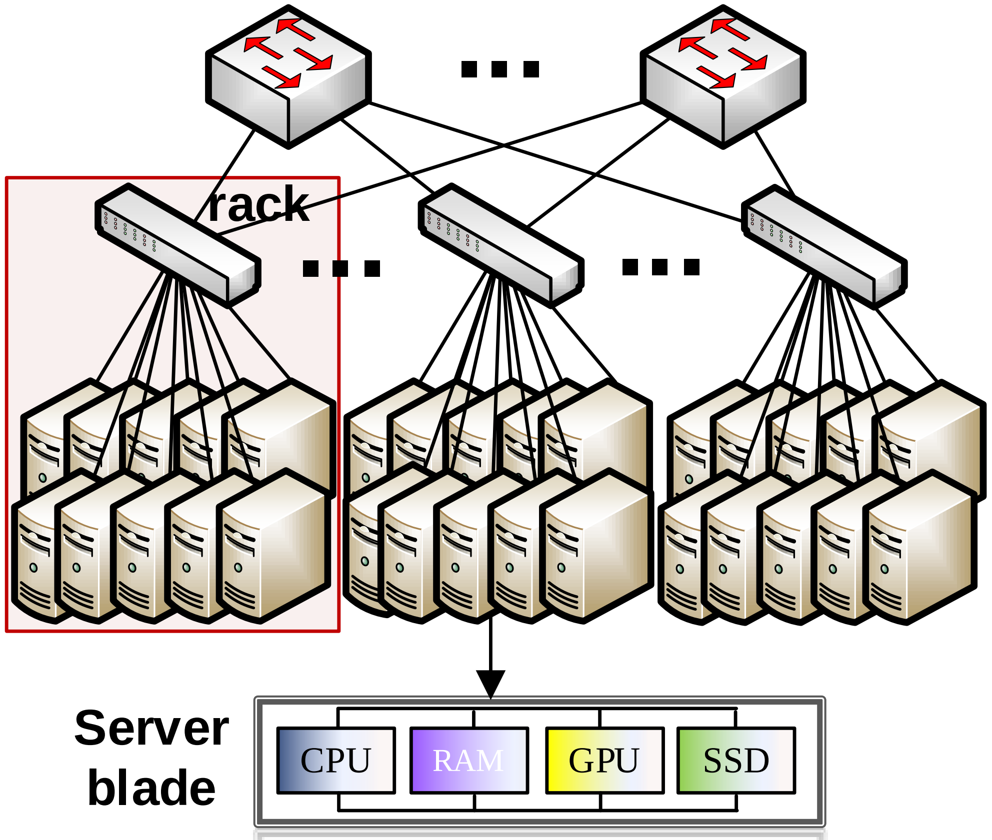
Therefore, the lower the proportion, the more efficient it is.



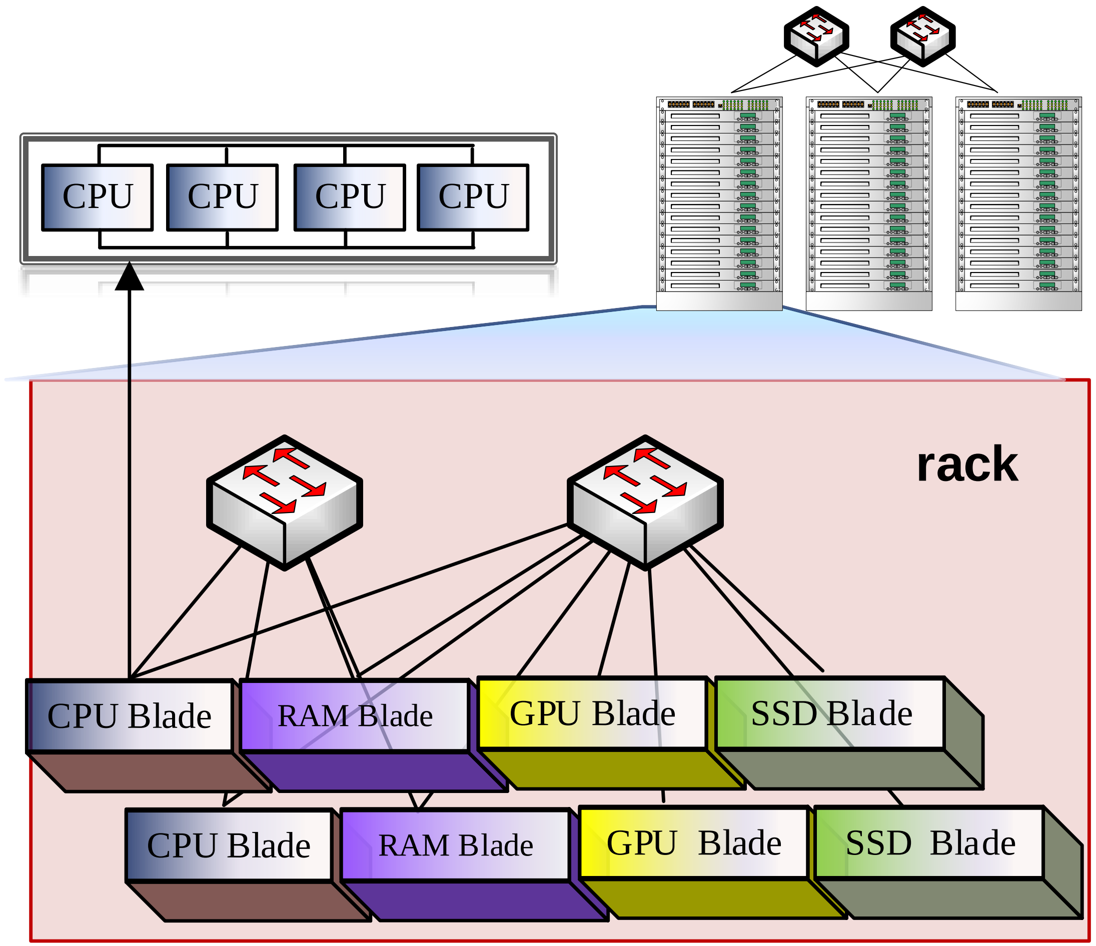
Study 3: Reliable Resource Allocation for DDCs Considering ***Network Effects***

Network Challenge & Disaggregation Scale

- Disaggregation & pooling being constrained by network capability



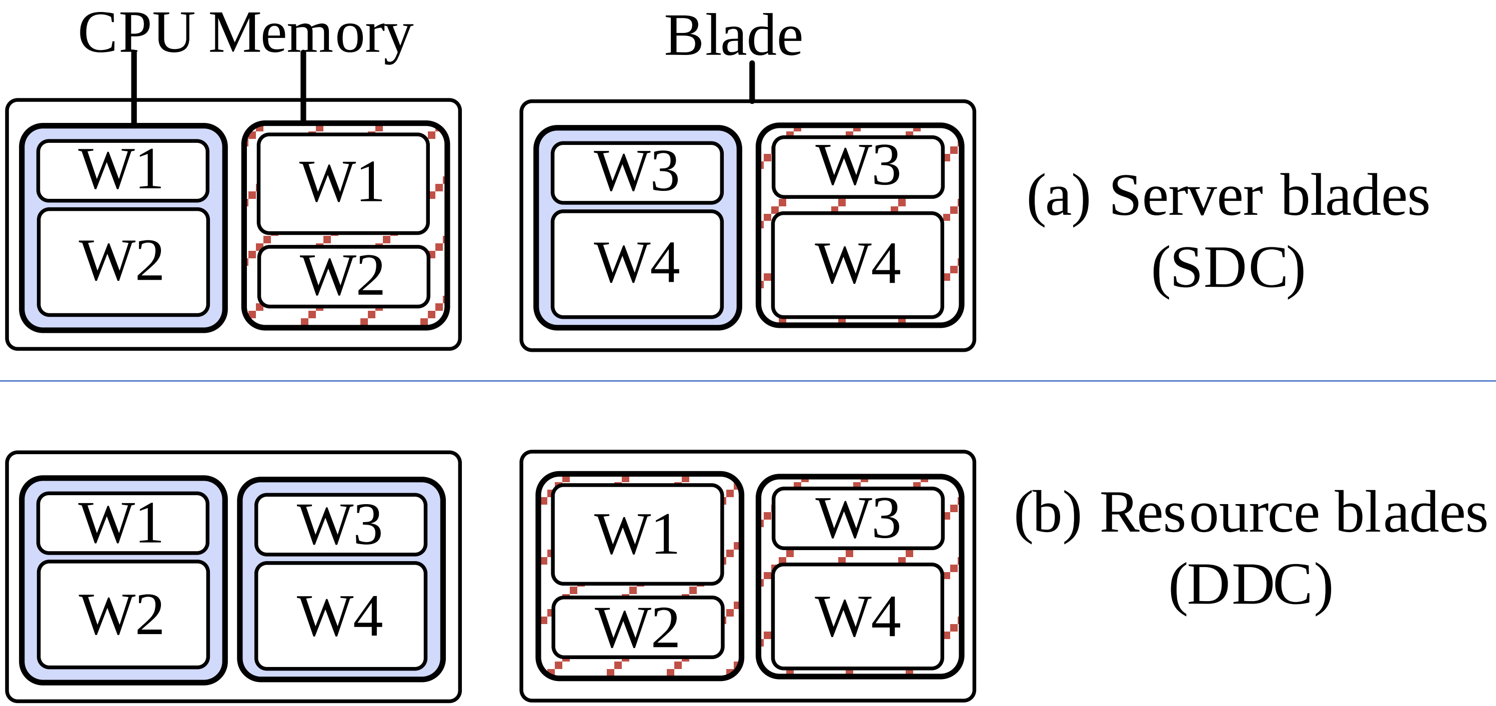
(a) Server-based DC architecture



(b) Rack-scale disaggregated DC architecture

Reliability Challenges of Resource Pooling

- Shared network – shared failures



Study Problem

- Given
  - $G(V,E)$
  - Each blade with multiple resource modules
  - Hardware parameters: 1) capacity, 2) reliability, 3) bandwidth, delay
  - Requests: resource demand, bandwidth and latency requirements
- Objective
  - Max: 1) Acceptance ratio; 2) Minimum request reliability.
- MILP (Chapter 5.3)
  - Weighted sum approach

Heuristic Method

- Rack Selection
  - Single rack for rack-scale DDC
- Blade Selection
  - **|R|** blades, one for each resource type
- Module Selection
  - Use multiple modules to allocate one type of resource

Heuristic Method

- Blade/Rack Selection: Select a blade/rack with high blade/rack index (η):

$$\eta = \varepsilon \cdot \eta^{rel} + (1 - \varepsilon) \cdot \eta^{eff}$$

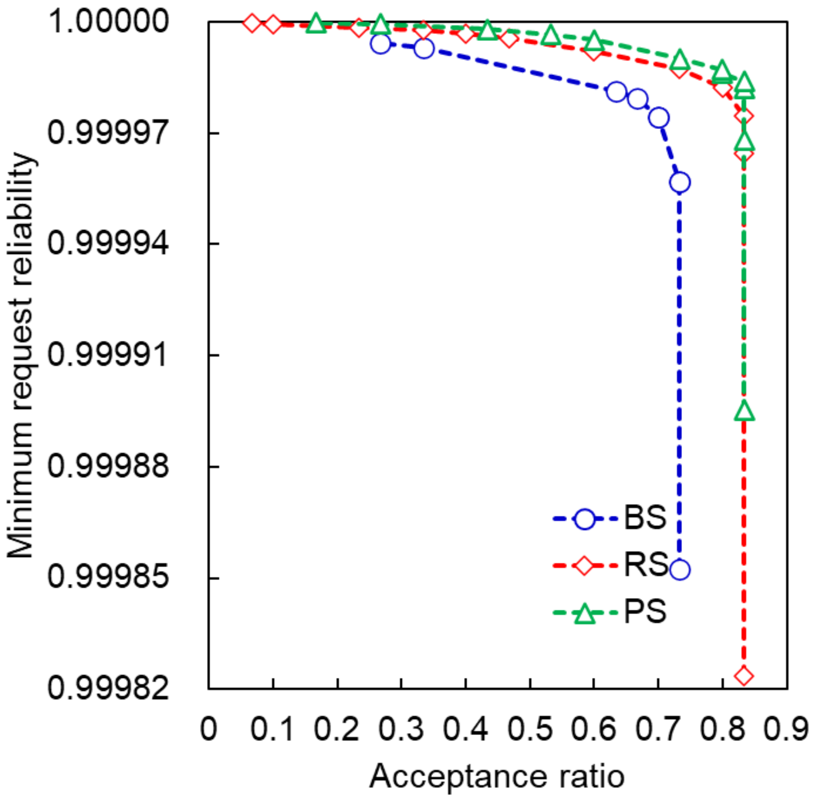
η^{eff}: efficiency index, defined as the (average) utilization of the blade/rack

η^{rel} : reliability index, defined as the product of the probability that the used hardware does not fail during the service time of a request

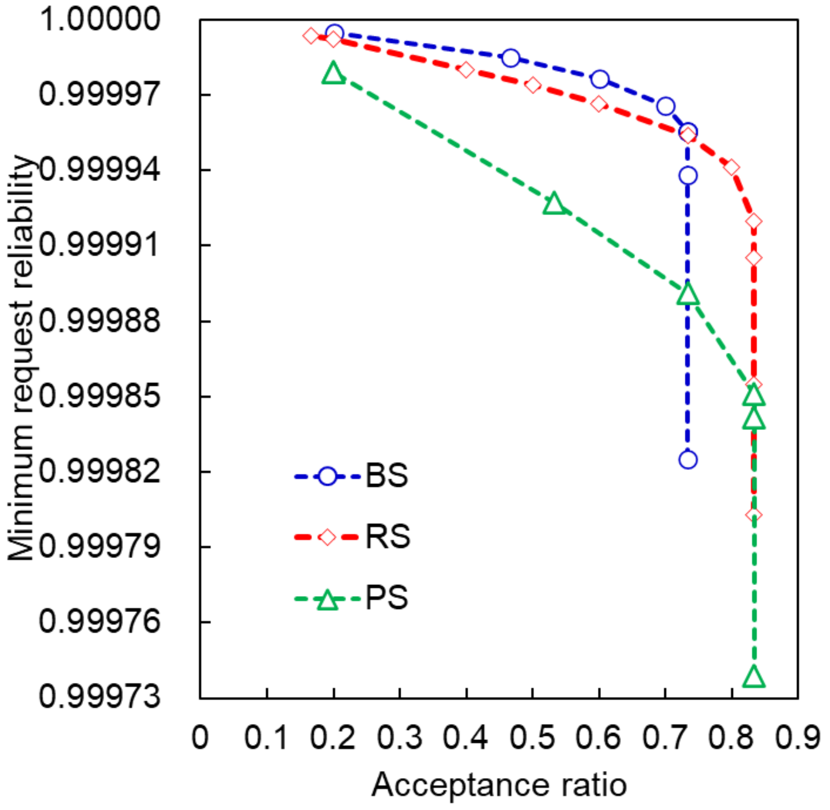
ε: weighting coefficient, $\varepsilon \in [0,1]$

Approximate Pareto Fronts Comparison

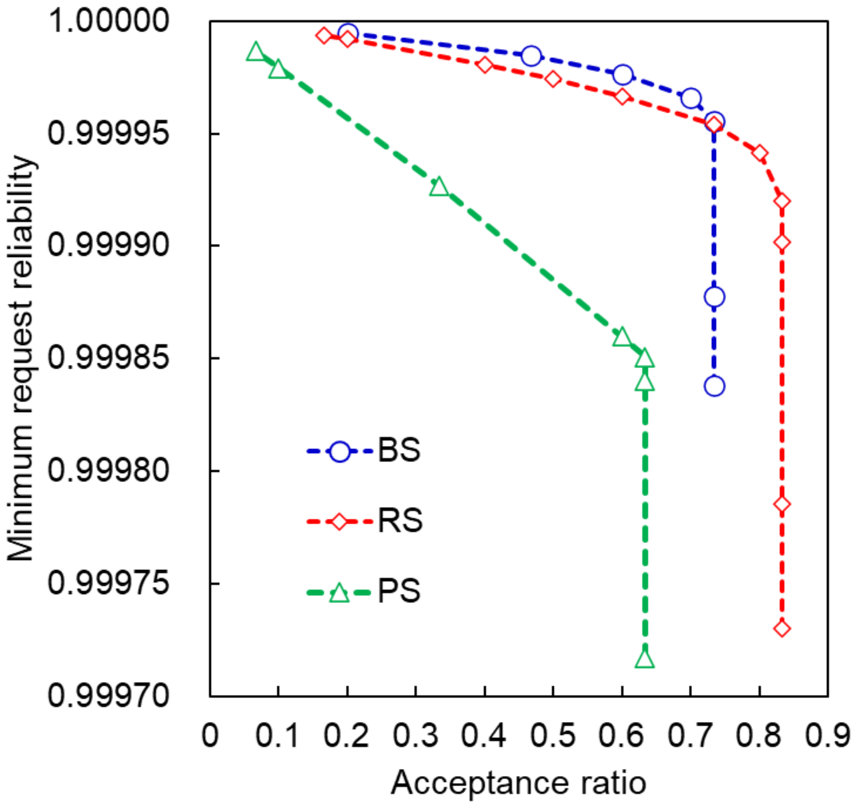
“Perfect” network



Not resilient network

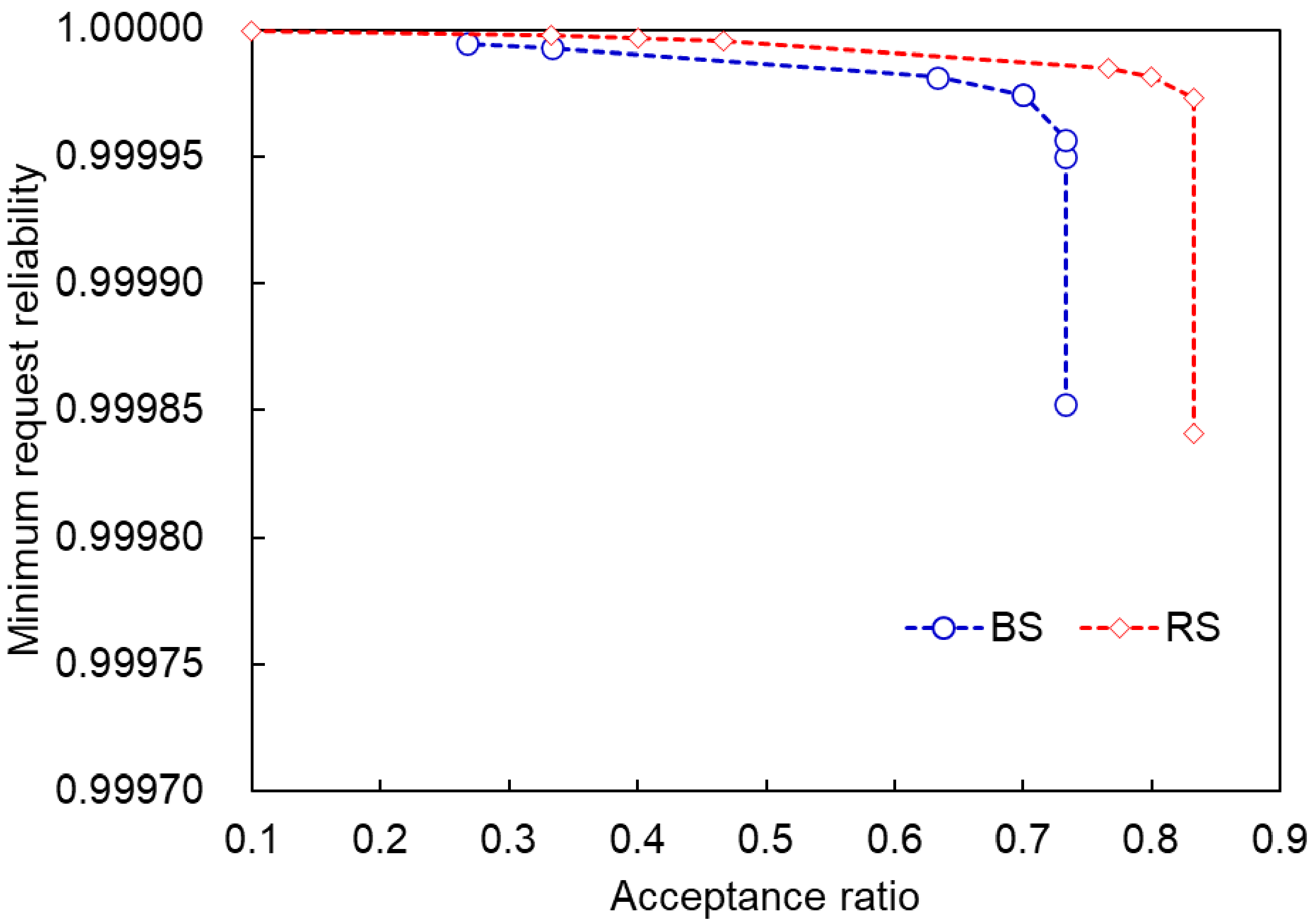


Not resilient network with not sufficiently low latency



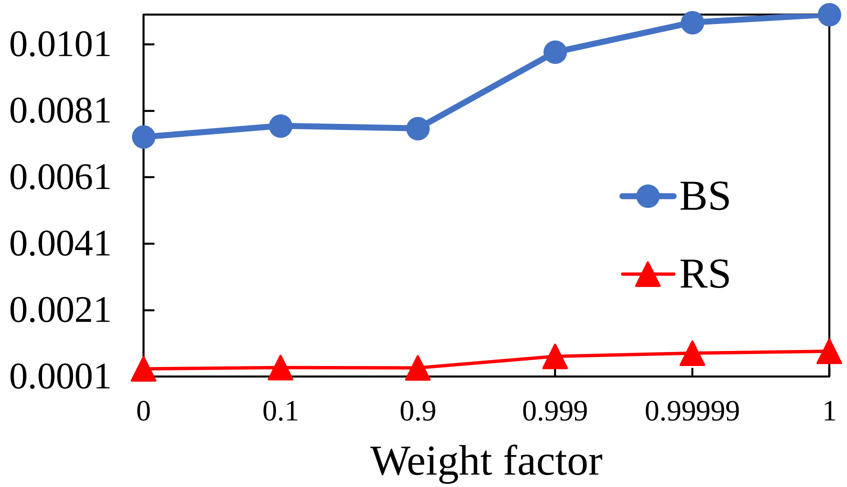
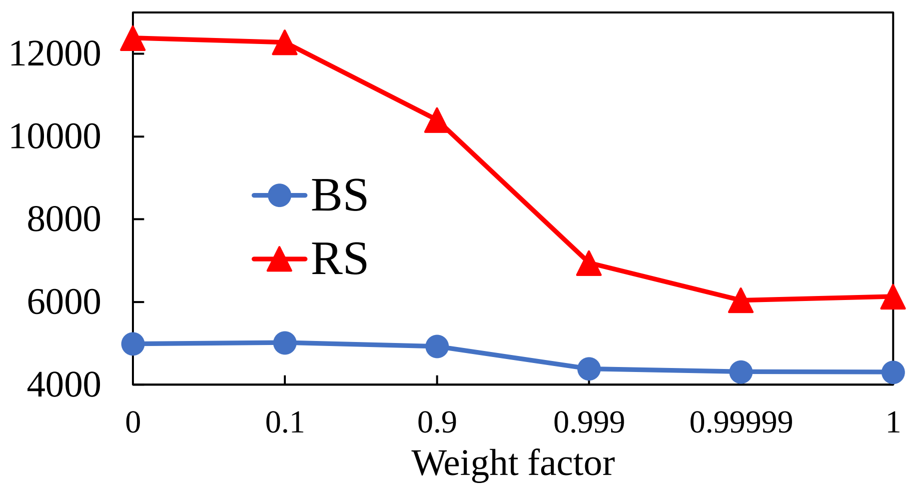
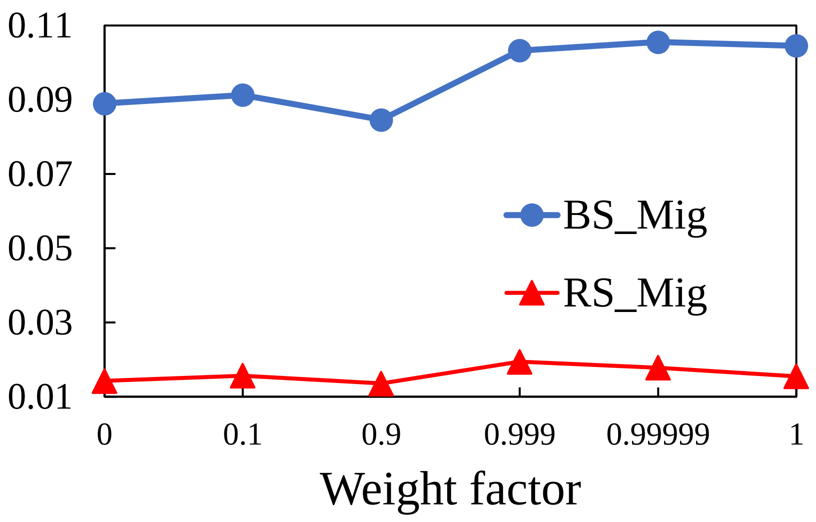
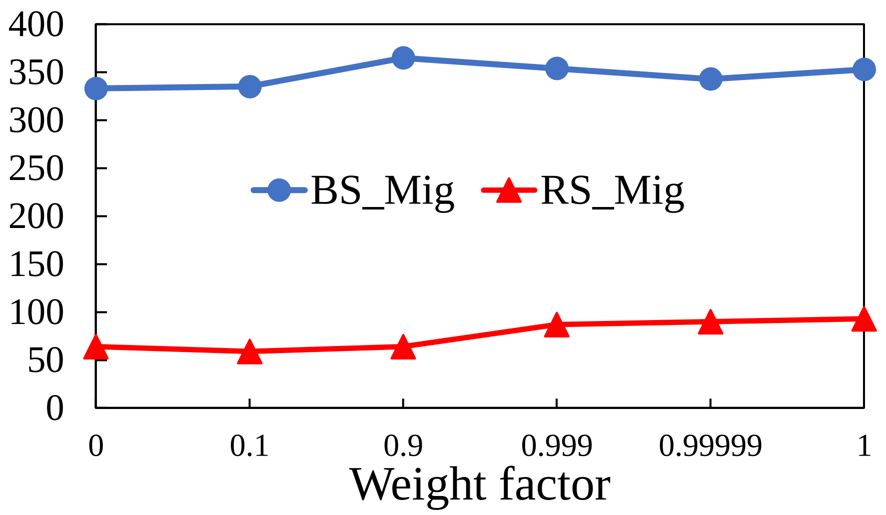
Applying Backup

PS has been removed “OUT”



Migration-Based Restoration

- Principle:
  - Migrate interrupted requests from failed hardware elsewhere, to restore the service.
- Simulation
  - Request arrival (Poisson)
  - Request departure (Service time: Exponential)
  - Hardware failure (Weibull)
  - Hardware repair (Exponential)

	Blocking ratio	Number of (accepted) requests failing to complete services
No restoration		
Migration based restoration		

CONCLUSION

- We design a temperature-aware VDC embedding scheme which can not only proactively balance the inlet temperatures and avoid hot spots but also achieve high energy-efficiency.
- We design a reliability-aware resource allocation method for a DDC which can achieve a high number of acceptances with guaranteed reliability requirement.
- We design a resource allocation method for a DDC considering network effects and different disaggregation scales, where we find the reliability benefit is possible to be offset by an imperfect network. We propose a migration scheme to overcome such issue.