IC3IT 2026

2026/4/7

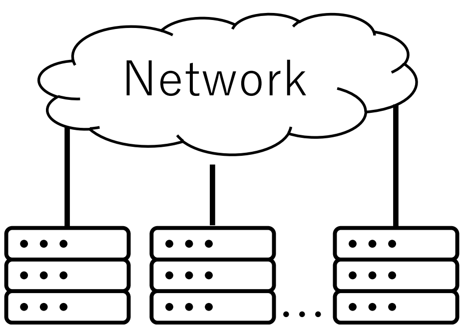
Optimal Resource Allocation for Disaggregated Data Centers under Uncertain Task Demands

CHEN HANZHANG, Yuichi Ohsita, Hideyuki Shimonishi

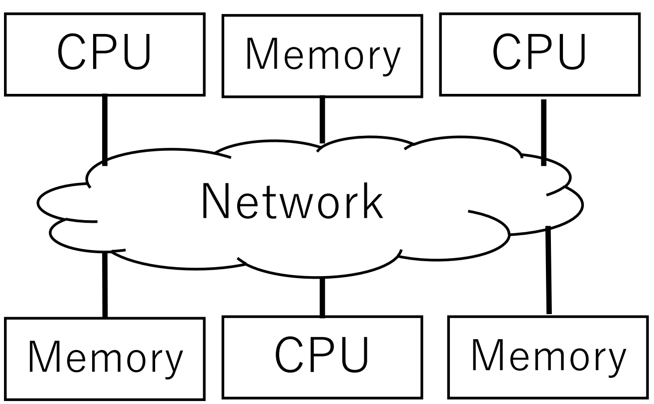
大阪大学
THE UNIVERSITY OF OSAKA

Disaggregated Data Center (DDC)

- Disaggregated Data Center:
  - CPU and memory are connected via the high-speed network
  - Enable only required resources are allocated
- Merits of DDC:
  - Flexible composition across servers
  - Improving overall resource utilization by reducing idle resources during operation



Traditional Data Center (Server-Centric)

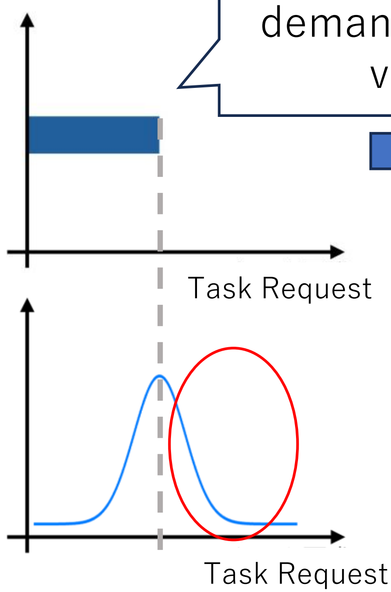


Disaggregated Data Center (DDC)

Resource Allocation Problem in DDC

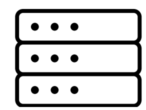
- Challenge in DDCs: Efficient resource allocation is required to execute tasks under limited resources.
- Previous Research of Resource Allocation Method

Treat resource demands as fixed values



Resource Allocation Algorithm

Task Allocation



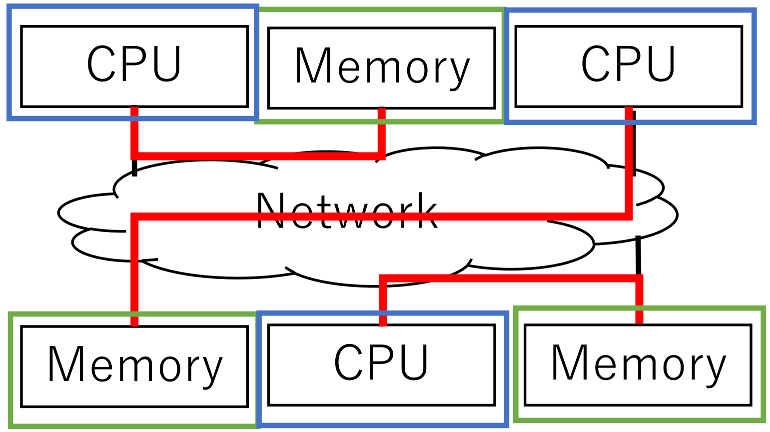
In practice, task demands are not fixed but uncertain

Difficult to handle cases where the required resources are larger than assumed fixes value.

Consider resource uncertainty is necessary to avoid performance degradation in DDCs

Objective: Propose an optimal resource allocation method accounting uncertain resources request.

Modeling of the DDC



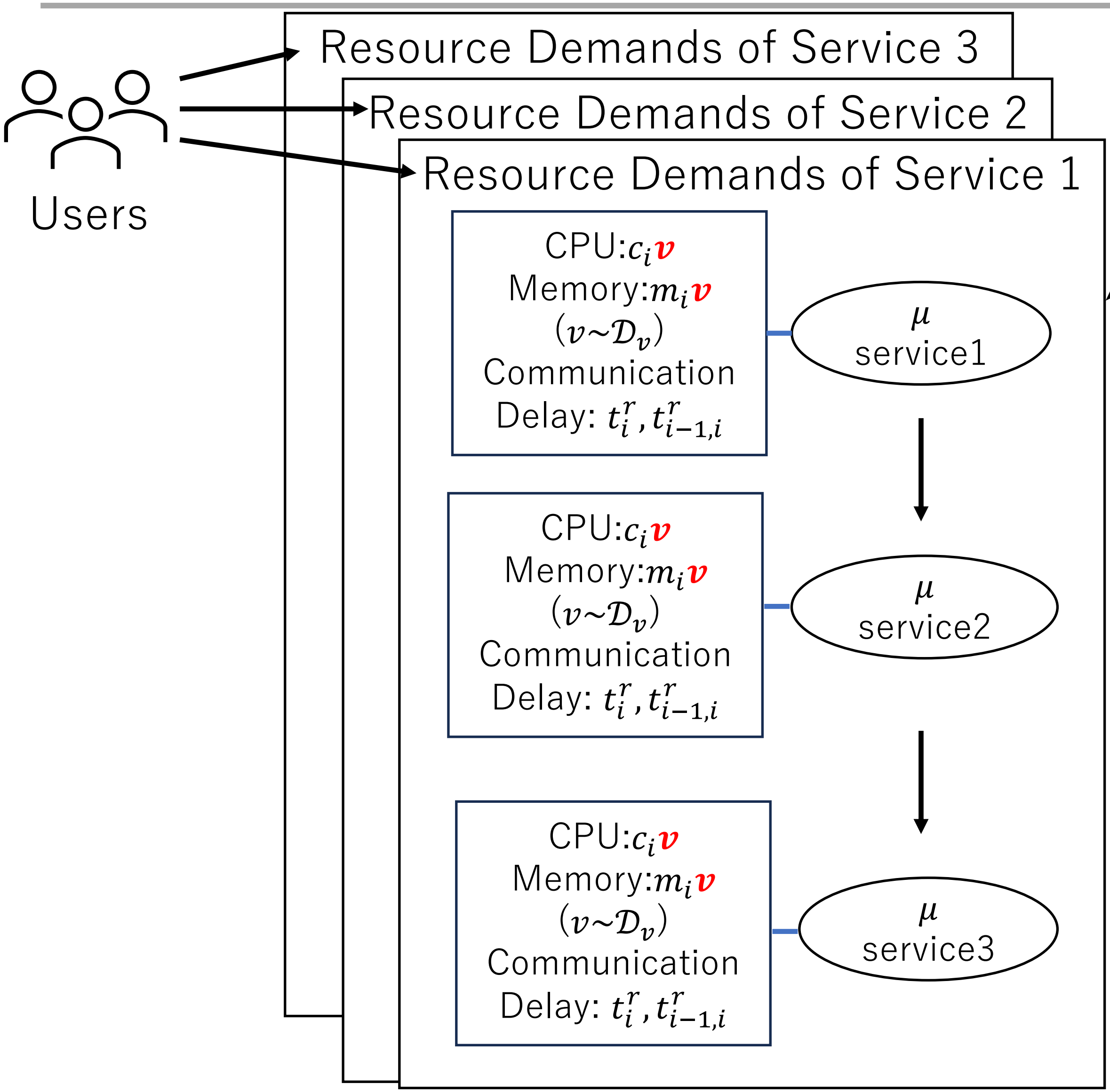
Disaggregated Data Center
(DDC)

Computational resource: $c \in C$.
Computing capacity per time of $c \in C$: S_c

Memory resource: $m \in M$
Processing capacity of $m \in M$: S_m

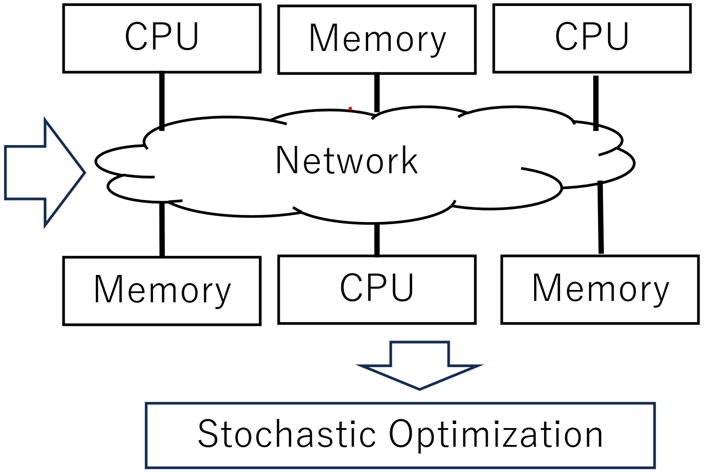
Communication path (delay follows probability distribution $\mathcal{D}_{a,b}^{delay}$)

Resource Allocation Request



In each service:

- Multiple μ services are sequentially executed
- Its resource demand is uncertain, so we represent it as a random variable



Resource Allocation Formulation

- Maximize allocated tasks while limiting resource shortage and delay violations
- Objective Function:
  - $\max \sum_{r \in R} x_r$

Binary variable indicating whether a request is allocated

Maximize accepted requests

- Constraints
  1. $\forall r \in R, \forall i \in P: \sum_{c \in C} u_{r,i,c} \geq x_r$
  2. $\forall r \in R, \forall i \in P: \sum_{m \in M} u_{r,i,m} \geq x_r$
  3. $\forall c \in C: P\left(\sum_{r \in R} \sum_{i \in P} u_{r,i,c} c_i v \leq S_c\right) \geq T_h$
  4. $\forall m \in M: P\left(\sum_{r \in R} \sum_{i \in P} u_{r,i,m} m_i v \leq S_m\right) \geq T_h$
  5. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M:$
     $P\left(p_{r,i,c_i,m_i} T_{c_i,m_i} \leq t_i^r\right) \geq T_h, P\left(p_{r,i,c_i,m_{i-1}} T_{c_i,m_{i-1}} \leq t_{i-1,i}^r\right) \geq T_h$
  6. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M:$
     $$p_{r,i,c,m} \geq z_{r,i,c} + z_{r,i,m} - 1$$
  7. Value of variable

Resource Allocation Formulation

- Maximize allocated tasks while limiting resource shortage and delay violations
- Objective Function:
  - $\max \sum_{r \in R} x_r$
- Constraints:
  1. $\forall r \in R, \forall i \in P: \sum_{c \in C} u_{r,i,c} \geq x_r$
  2. $\forall r \in R, \forall i \in P: \sum_{m \in M} u_{r,i,m} \geq x_r$
  3. $\ldots \geq T_h$
  4. $\ldots S_m) \geq T_h$
  5. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M$:
     $P(p_{r,i,c_i,m_i} T_{c_i,m_i} \leq t_i^r) \geq T_h, P(p_{r,i,c_i,m_{i-1}} T_{c_i,m_{i-1}} \leq t_{i-1,i}^r) \geq T_h$
  6. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M$:
     $$p_{r,i,c,m} \geq z_{r,i,c} + z_{r,i,m} - 1$$
  7. Value of variable

If request r is accepted ($x_r = 1$), every process must receive both CPU and memory resources

Binary variable indicating whether a request is allocated (x_r)

Fraction of resource c/m usage for each process ($u_{r,i,c}$, $u_{r,i,m}$)

Resource Allocation Formulation

- Maximize allocated tasks while limiting resource shortage and delay violations
- Objective Function:
  - $\max \sum_{r \in R} x_r$
- Constraints:

Fraction of resource c/m usage for each process

1. $\forall r \in R, \forall i \in P: \sum_{c \in C} u_{r,i,c} \geq x_r$
2. $\forall r \in R, \forall i \in P: \sum_{m \in M} u_{r,i,m} \geq x_r$
3. $\forall c \in C: P\left(\sum_{r \in R} \sum_{i \in P} u_{r,i,c} c_i v \leq S_c\right) \geq T_h$
4. $\forall m \in M: P\left(\sum_{r \in R} \sum_{i \in P} u_{r,i,m} m_i v \leq S_m\right) \geq T_h$

For every resource, the probability that the load does not exceed its capacity must be at least T_h

5. $^{r}_{i-1,i}) \geq T_h$
6. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M:$
   $$p_{r,i,c,m} \geq z_{r,i,c} + z_{r,i,m} - 1$$
7. Value of variable

Resource Allocation Formulation

- Maximize allocated tasks while limiting resource shortage and delay violations
- Objective Function:
  - $\max \sum_{r \in R} x_r$
- Constraints:
  1. $\forall r \in R, \forall i \in P: \sum_{c \in C} u_{r,i,c} \geq x_r$
  2. $\forall r \in R, \forall i \in P: \sum_{m \in M} u_{r,i,m} \geq x_r$
  3. $\forall c \in C: P(\sum \quad \sum \quad u \quad c\, r \leq S \,) \geq T$

For each task, both intra-process and inter-process delays must satisfy their delay limits with probability at least T_h

  5. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M$:
     $P\left(p_{r,i,c_i,m_i} T_{c_i,m_i} \leq t_i^r\right) \geq T_h, P\left(p_{r,i,c_i,m_{i-1}} T_{c_i,m_{i-1}} \leq t_{i-1,i}^r\right) \geq T_h$
  6. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M$:
     $p_{r,i,c,m} \geq z_{r,i,c} + z_{r,i,m} - 1$
  7. Value of variable

Variable indicating whether communication occurs between allocated resource nodes

Communication delay is considered only between used resource nodes

Resource Allocation Formulation

- Maximize allocated tasks while limiting resource shortage and delay violations
- Objective Function:
  - $\max \sum_{r \in R} x_r$
- Constraints:

1. $\forall r \in R, \forall i \in P: \sum_{c \in C} u_{r,i,c} \geq x_r$
2. $\forall r \in R, \forall i \in P: \sum_{m \in M} u_{r,i,m} \geq x_r$
3. $\forall c \in C: P\left(\sum_{r \in R} \sum_{i \in P} u_{r,i,c} c_i v \leq S_c\right) \geq T_h$
4. $\forall m \in M: P\left(\sum_{r \in R} \sum_{i \in P} u_{r,i,m} m_i v \leq S_m\right) \geq T_h$
5. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M:$
   $P\left(p_{r,i,c_i,m_i} T_{c_i,m_i} \leq t_i^r\right) \geq T_h, P\left(p_{r,i,c_i,m_{i-1}} T_{c_i,m_{i-1}} \leq t_{i-1,i}^r\right) \geq T_h$
6. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M:$
   $$p_{r,i,c,m} \geq z_{r,i,c} + z_{r,i,m} - 1$$
7. Value of variable

$\forall r \in R, i \in P, c \in C: u_{r,i,c} \in \{0, \delta, 2\delta, \ldots, 1\}$	$\forall r \in R: x_r \in \{0,1\}$
$\forall r \in R, i \in P, m \in M: u_{r,i,m} \in \{0, \delta, 2\delta, \ldots, 1\}$	$\forall r \in R, \forall i \in P, \forall c \in C: z_{r,c} \in \{0,1\}$
$\forall r \in R, i \in P, c \in C, m \in M: p_{r,i,c,m} \in \{0,1\}$	$\forall r \in R, \forall i \in P, \forall m \in M: z_{r,m} \in \{0,1\}$

Solution Approach

- Convert to a mixed-integer programming (MIP) problem
- Constraints:

3. $\forall c \in C, \forall \omega \in \Omega: \sum_{r \in R} \sum_{i \in P} u_{r,i,c}\, c_i v^{\omega} - L z_c^{\omega} \leq S_c,$

$$\forall c \in C: \frac{1}{|\Omega|} \sum_{\omega \in \Omega} z_c^{\omega} \leq 1 - T_h$$

4. $\forall m \in M, \forall \omega \in \Omega: \sum_{r \in R} \sum_{i \in P} u_{r,i,m}\, m_i v^{\omega} - L z_m^{\omega} \leq S_m,$

$$\forall m \in M: \frac{1}{|\Omega|} \sum_{\omega \in \Omega} z_m^{\omega} \leq 1 - T_h$$

5. $\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M, \forall \omega \in \Omega:$

$$p_{r,i.c_i,m_i} T_{c_i,m_i}^{\omega} - L z_{r,i,c_i,m_i}^{\omega} \leq t_i^r,\ p_{r,i.c_i,m_{i-1}} T_{c_i,m_{i-1}}^{\omega} - L z_{r,i,c_i,m_{i-1}}^{\omega} \leq t_{i-1,i}^r$$

$\forall r \in R, \forall i \in P, \forall c \in C, \forall m \in M:$

$$\frac{1}{|\Omega|} \sum_{\omega \in \Omega} z_{r,i,c_i,m_i}^{\omega} \leq 1 - T_h,\ \frac{1}{|\Omega|} \sum_{\omega \in \Omega} z_{r,i,c_i,m_{i-1}}^{\omega} \leq 1 - T_h,$$

Replace random variables using sampled scenarios $\omega \in \Omega$, The violation ratio across all samples must be $\leq 1 - T_h$

Red box: Scenario sampled from the probability distribution

Blue box: Binary variable indicating resource constraint violation (violation indicator)

L: Large Constant

Experimental Setup

DDC Configuration

Number of computational nodes	3
Number of memory nodes	3
CPU frequency per node(GHz)	3.4
Memory processing capacity per node(packet/ms)	200
Communication delay	$X \sim \mathcal{N}(\mu = 0.3, \sigma = 0.5)$

The number of users follows:
$X \sim \mathcal{N}(\mu = 10, \sigma = 6)$

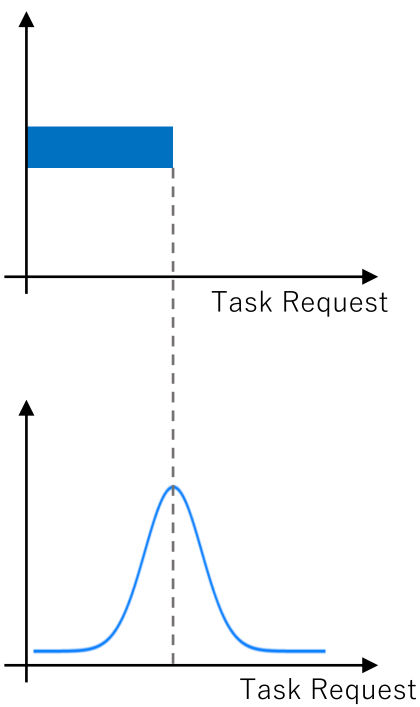
Information of services provided

	Segmentation	3D Object Detection	Bird's Eye View Perception
μ service 1 (process 1)			
Clock Cycles	68199102	49289698	181852
Write/Read Packet Arrival Rate	2.58/4.38	1.74/0.0	0.72/2.61
μ service 2 (process 2)			
Clock Cycles	33095402	52219370	1477624780
Write/Read Packet Arrival Rate	3.4/3.99	0.77/5.43	8.36/10.96
μ service 3 (process)			
Clock Cycles	79239	59155594	399211
Write/Read Packet Arrival Rate	1.39/0.0	1.09/0.0	3.72/13.13

Comparison Method

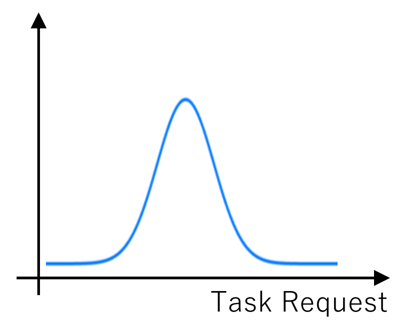
Comparison Method:

- Treat demands in fixed value
- The fixed value corresponds to p% in probability distribution of proposed method



Proposed Method:

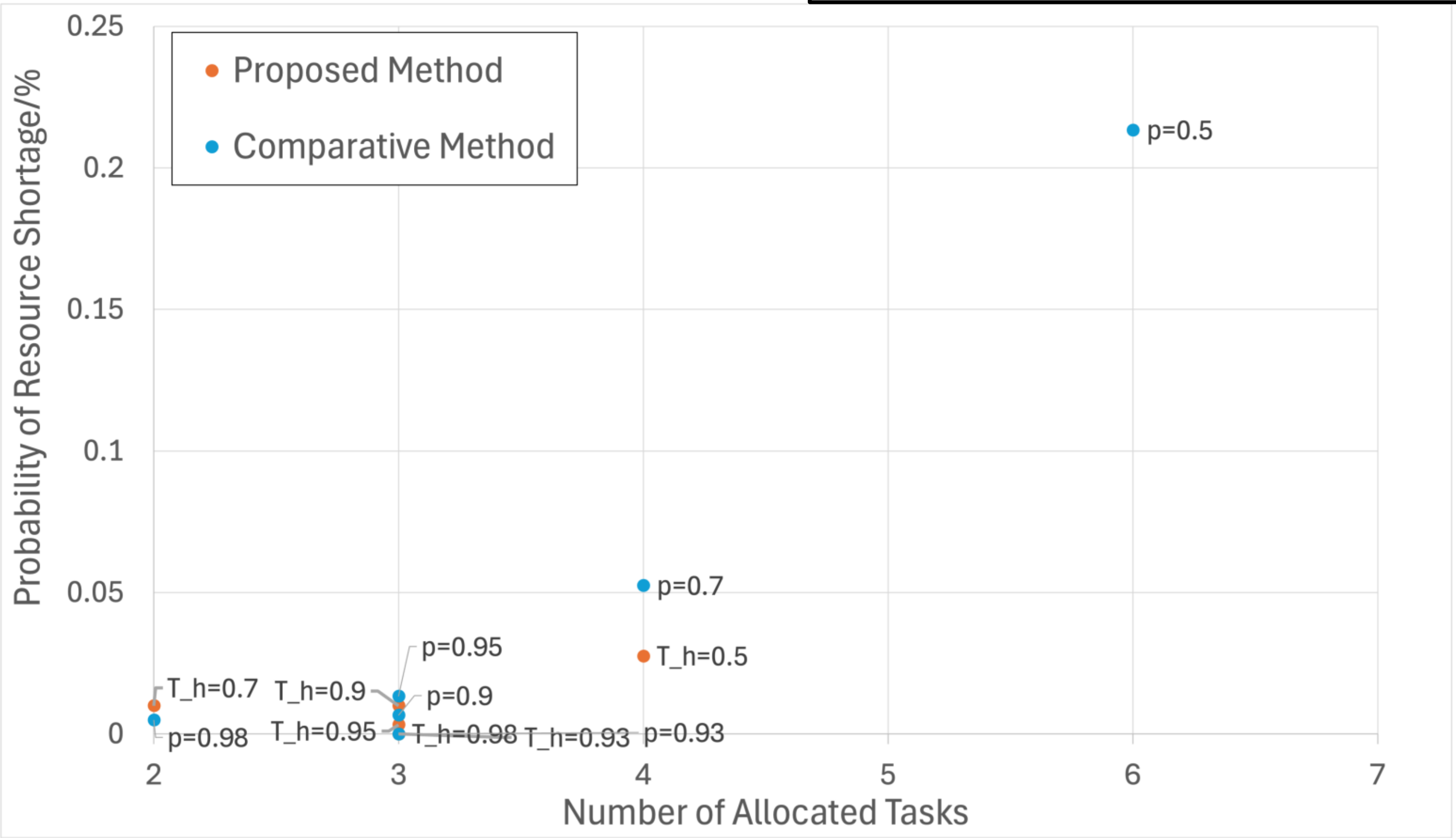
- Treat demands in probability distribution.
- Ensure that resource overload won’t occur above a threshold.



Evaluation Results

- 3D Object Detection

X-axis: Number of allocate tasks
Y-axis: Probability of resource shortage

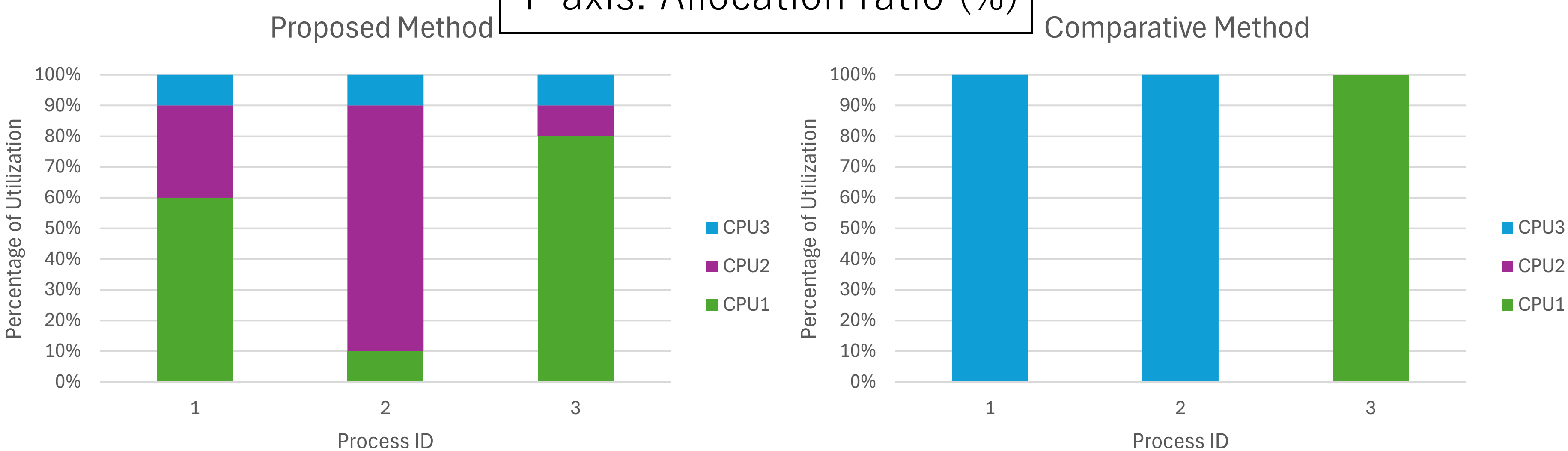


The proposed method achieves a better trade-off :
For the same name of allocated tasks, it reduces the probability of resource shortage

Evaluation Results

- Examples of resource allocation for 3D Objective Detection (93% demand satisfaction)：

X-axis: Process ID
Y-axis: Allocation ratio (%)



- The proposed method shows more distributed allocation across nodes.

Conclusion and Future Works

- Conclusion
  - We proposed a resource allocation method for DDC under uncertain task demands
  - The proposed method reduces resource shortages and allocates more tasks
- Future Works
  - Evaluation on larger-scale DDC systems
  - Integrate prediction methods to update demand models using observed resource usage