International Journal of Creative and Open Research in Engineering and Management
ISSN: **3108-1754 (Online)**
Volume 02 Issue 08 August-2026 | Impact Factor: 3.5

Cost Aware Resource Allocation in Cloud Computing: A Comparison of Heuristic and Greedy Approaches

Dhanush.V
PG Student
Department of Computer Science
Bharathiar University
Coimbatore – 641 046
Email id : dhanushvinfo@gmail.com

How to Cite this Article:

Dhanush.V, (2026). Cost Aware Resource Allocation in Cloud Computing: A Comparison of Heuristic and Greedy Approaches. International Journal of Creative and Open Research in Engineering and Management, 02(8), 1-9. https://doi.org/10.55041/ijcope.v2i8.186

https://doi.org/10.55041/ijcope.v2i8.186

Abstract-Cloud computing has become an essential platform for delivering scalable and on-demand computing resources. However, inefficient resource allocation often leads to increased operational costs and poor utilization of available resources. This paper focuses on addressing this issue by proposing a cost-aware resource allocation approach using a heuristic method and comparing its performance with a greedy allocation strategy. The heuristic approach assigns resources based on the actual requirements of tasks, aiming to minimize resource wastage and reduce overall cost. In contrast, the greedy method makes quick allocation decisions without considering future needs, which can result in over-provisioning. Experimental results demonstrate that the heuristic approach achieves better cost efficiency and improved resource utilization compared to the greedy method. The findings highlight that simple, rule-based allocation strategies can significantly enhance performance in cloud environments while maintaining low computational complexity.

Keywords: Cloud Computing, Resource Allocation, Heuristic Method, Greedy Algorithm, Cost Optimization, Virtualization, Task Scheduling, Resource Utilization

I. INTRODUCTION

Cloud computing has transformed the way computing resources are accessed and utilized by providing scalable, on-demand services over the internet. Platforms such as Amazon Web Services and Microsoft Azure allow users to dynamically allocate computing resources based on their needs. This flexibility has significantly reduced infrastructure costs and improved system efficiency.

Despite these advantages, resource allocation remains a critical challenge in cloud environments. Inefficient allocation can lead to over-provisioning, underutilization, and increased operational costs. Therefore, designing efficient resource allocation strategies is essential to ensure optimal performance and cost-effectiveness.

This paper presents a cost-aware resource allocation approach using a heuristic method and compares it with a greedy allocation strategy. The main objective is to analyze which approach provides better cost efficiency and resource utilization.

II. LITERATURE REVIEW

Resource allocation in cloud computing has been widely studied, with various techniques proposed to improve system performance and reduce operational cost. Peter Mell and Timothy Grance defined cloud computing and emphasized the importance of efficient resource management in large-scale cloud systems [1].

Rajkumar Buyya et al. presented fundamental concepts of cloud computing and highlighted resource provisioning and scheduling as key challenges in dynamic environments [2]. Similarly, Michael Armbrust et al. discussed the opportunities and challenges of cloud computing, including scalability, performance, and cost optimization [3].

Modern cloud data centers are designed to efficiently manage large-scale workloads. Luiz André Barroso et al. described data centers as warehouse-scale computers and emphasized the need for efficient resource utilization [4]. In addition, Singh and Chana analyzed quality-of-service (QoS)-aware resource management techniques and highlighted the importance of adaptive allocation strategies [5].

Heuristic-based approaches are widely used for resource allocation due to their simplicity and ability to provide near-optimal solutions with low computational complexity. Liu et

al. proposed a heuristic resource allocation model that improves efficiency in cloud environments [6]. Similarly, Li discussed optimal load distribution techniques for heterogeneous cloud systems [7].

Dynamic resource allocation has also been explored using distributed decision-making techniques. Xu and Li proposed a distributed approach to allocate resources efficiently in cloud environments [8]. Furthermore, Dean and Barroso highlighted performance challenges in large-scale systems and the need for efficient task scheduling mechanisms [9].

Auto-scaling techniques have been introduced to dynamically adjust resources based on workload demand. Lorido-Botran et al. discussed various auto-scaling strategies to improve resource utilization and reduce operational cost in cloud environments [10].

Although significant progress has been made, there is still a need for simple and practical approaches that can efficiently balance cost and resource utilization. This motivates the use of heuristic-based methods and their comparison with greedy approaches in this study.

III. METHODOLOGY

This section describes the proposed cost-aware resource allocation framework and the implementation of heuristic and greedy approaches for efficient resource management in cloud computing environments.

A. System Architecture

The proposed system consists of the following components:

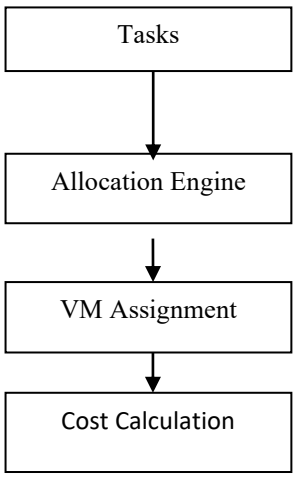

Fig. 1. Proposed System Architecture

The proposed system architecture consists of task input, resource allocation engine, and cost evaluation module. The architecture is designed to efficiently allocate resources using heuristic and greedy approaches.

The system operates in a virtualized environment using technologies such as VMware, where physical resources are divided into multiple virtual machines.

B. Task Model

Each task is defined as:

$$T_i = CPU_i, RAM_i, Time_i \quad (1)$$

Where:

- CPU_I= CPU requirement of task
- RAM_i= Memory requirement
- $Time_i$= Execution time

Tasks are assumed to be independent and non-preemptive.

C. Resource Model

Cloud resources are represented as virtual machines (VMs) with predefined configurations:

TABLE I. Configurations

VM Type	CPU	RAM
Small	2	2 GB
Medium	4	4 GB
Large	6	8 GB
X-Large	8	8 GB

D. Heuristic-Based Allocation Algorithm

The heuristic approach allocates resources based on predefined rules considering task requirements.

Algorithm Steps:

1. Input task list
2. For each task:
   - If CPU ≤ 2 → assign Small VM
   - If CPU ≤ 4 → assign Medium VM
   - If CPU ≤ 6 → assign Large VM
   - Else → assign X-Large VM
3. Allocate minimum sufficient resources
4. Repeat for all tasks

The heuristic-based allocation reduces resource wastage, improves utilization, and minimizes cost.

E. Greedy-Based Allocation Algorithm

The greedy approach allocates resources based on immediate availability without considering optimization.

Algorithm Steps:

1. Input task list
2. Maintain a list of available VMs
3. For each task: Assign the next available VM
4. Repeat until all tasks are assigned

The greedy-based allocation is suitable for faster execution but may lead to over-allocation due to lack of optimization.

F. Cost Model

The total cost of resource allocation is calculated using

$$\text{Cost} = \sum_{i-1}^{n} [(\text{CPU}_i \times C_{cpu}) + (\text{RAM}_i \times C_{ram})] \times \text{Time}_i \quad (2)$$

This model reflects real-world cloud pricing strategies.

G. Performance Metrics

Total Cost : Measures overall expense of resource allocation

Resource Utilization :

$$Utilization = \frac{Used\ Resourses}{Allocated\ Resourses} \quad (3)$$

Execution Efficiency : Measures how effectively tasks are completed

IV. RESULTS AND EVALUATION

This section presents the experimental results of the proposed resource allocation approaches and evaluates their performance based on cost and resource utilization.

A. Experimental Setup

The proposed system is implemented using Python in a simulation environment. A set of tasks with varying CPU and memory requirements is considered for the experiment. Both heuristic and greedy allocation methods are applied to the same set of tasks to ensure a fair comparison.The cost of resource allocation is calculated using a predefined cost model, where CPU and memory usage are associated with specific cost values.

B. Input Dataset

The input consists of multiple tasks with different resource requirements, as shown in Table 2.

TABLE II. Input Dataset

Task	CPU Required	RAM Required
T1	2	2 GB
T2	5	6 GB
T3	3	4 GB
T4	6	8 GB

C. Resource Allocation Results

The allocation results obtained from both approaches are shown below.

TABLE III. Heuristic Allocation

Task	**Allocated CPU**	**Allocated RAM**
T1	2	2 GB
T2	6	8 GB
T3	4	4 GB
T4	6	8 GB

TABLE IV. Greedy Allocation

Task	**Allocated CPU**	**Allocated RAM**
T1	4	4 GB
T2	8	8 GB
T3	4	4 GB
T4	8	8 GB

The total cost for both methods is calculated based on CPU and RAM usage.

TABLE V. Total Cost

Method	**Total Resource Usage**	**Total Cost**
Heuristic	Lower	Lower
Greedy	Higher	Higher

The results clearly show that the heuristic approach reduces resource usage and overall cost compared to the greedy method.

D. Graphical Analysis

A graphical comparison is performed to visualize the cost difference between the two approaches. The cost comparison between heuristic and greedy methods is illustrated in Fig. 2.

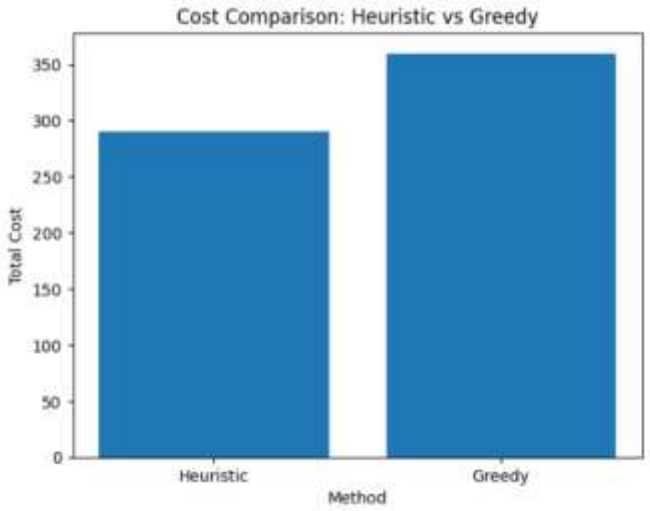

Fig. 2. Cost Comparison of Two Algorithms.

E. Performance Evaluation

The performance of both methods is evaluated based on the following metrics:

Cost Efficiency: The heuristic method achieves lower cost due to optimized allocation.

Resource Utilization: Better utilization is observed in the heuristic approach.

Execution Efficiency: The greedy method is faster but less efficient in resource usage.

F. Summary of Results

The experimental analysis confirms that:

- The heuristic approach minimizes cost by allocating only required resources.
- The greedy method leads to over-allocation and increased cost.
- The heuristic method provides a better balance between performance and cost.

V. DISCUSSION

The experimental results demonstrate that the heuristic-based resource allocation approach performs more efficiently compared to the greedy method. The heuristic method allocates resources based on task requirements, which reduces unnecessary resource usage and minimizes overall cost.

In contrast, the greedy approach assigns resources without considering optimization, leading to over-allocation in several cases. This results in higher cost and lower resource utilization.

Although the greedy method provides faster allocation, it lacks the ability to make intelligent decisions. The heuristic approach achieves a better balance between performance and cost, making it more suitable for cloud environments where cost efficiency is critical.

VI. CONCLUSION

This paper presented a cost-aware resource allocation strategy in cloud computing using a heuristic approach and compared it with a greedy allocation method. The results show that the heuristic method significantly improves resource utilization and reduces overall cost compared to the greedy approach.

The study highlights that simple rule-based techniques can effectively address resource allocation challenges without introducing high computational complexity. The heuristic approach proves to be a practical and efficient solution for cloud environments.

VII. FUTURE WORK

Future research can explore the integration of machine learning techniques for predictive resource allocation. Additionally, advanced optimization methods such as genetic algorithms and swarm intelligence can be used to further enhance performance and reduce cost in dynamic cloud environments.

References

[1] P. Mell and T. Grance, “The NIST Definition of Cloud Computing,” National Institute of Standards and Technology (NIST), 2011.

[2] R. Buyya, J. Broberg, and A. Goscinski, “Cloud Computing: Principles and Paradigms,” Wiley, 2011.

[3] M. Armbrust et al., “A View of Cloud Computing,” Communications of the ACM, vol. 53, no. 4, pp. 50–58, 2010.

[4] L. A. Barroso, J. Clidaras, and U. Hölzle, “The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines,” Morgan & Claypool, 2013.

[5] S. Singh and I. Chana, “QoS-aware autonomic resource management in cloud computing: A systematic review,” ACM Computing Surveys, vol. 48, no. 3, 2016.

[6] X. Liu, C. Wang, B. Zhou, and J. Chen, “A Heuristic Resource Allocation Approach for Cloud Computing,” IEEE International Conference, 2019.

[7] K. Li, “Optimal Load Distribution in Non-Dedicated Heterogeneous Cloud Computing Systems,” IEEE Transactions on Computers, vol. 64, no. 3, 2015.

[8] H. Xu and B. Li, “Dynamic Cloud Resource Allocation via Distributed Decisions,” IEEE Journal on Selected Areas in Communications, vol. 31, no. 12, 2013.

[9] J. Dean and L. A. Barroso, “The Tail at Scale,” Communications of the ACM, vol. 56, no. 2, 2013.

[10] T. Lorido-Botran, J. Miguel-Alonso, and J. A. Lozano, “Auto-scaling Techniques for Elastic Applications in Cloud Environments,” Department of Computer Architecture, University of Basque Country, 2014.