The Scientific Temper (2023) Vol. 14 (4): 1339-1344
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2023.14.4.41
https://scientifictemper.com/

RESEARCH ARTICLE

Cost-based resource allocation method for efficient allocation of resources in a heterogeneous cloud environment

Sheena Edavalath*, **Manikandasaran S. Sundaram**

Abstract

Cloud computing is appealing due to features like adaptability, portability, utility service and on-demand service. Cloud resource providers are a source of computing, and each provider delivers different types of resources. In an active cloud environment, timely resource allocation is more important. In order to increase the effectiveness and user-friendliness of resource allocation in the heterogeneous cloud, the paper suggests an efficient cost-based resource allocation (ECRA) method and framework. In the heterogeneous cloud, there is no centralized resource allocation manager (CRAM) to get all requested resources from a single counter. The proposed methodology for allocating resources divides them according to their cost. The paper's framework for allocating resources consists of various parts. The unified heterogeneous resource allocation manager (UHRAM) part of the framework collects and manages resources from several cloud resource providers. The resource identifier is one of the components in the framework, which is coupled to UHRAM to determine the cost of the resources. The low-cost resources are scheduled and to be in a ready state for allocation. The proposed ECRA is simulated and compared based on parameters like total computation time, response time and resource allocation percentage with existing resource allocation methods. The results prove that the proposed ECRA is efficient in allocating the resources in minimum response time and it allocates maximum resources for lower cost.

Keywords: Cloud computing, Resource allocation, Resource cost, Resource utilization, Heterogeneous cloud environment, Centralized resource allocation.

Introduction

The way that organizations and individuals use computer resources has been completely transformed by cloud computing. Without requiring an initial financial investment, it offers on-demand provisioning of computer resources, such as internet, memory, applications, and services. Because it is affordable, scalable, and adaptable, cloud computing is becoming more and more popular across a range of sectors, including e-commerce, banking, and healthcare (R.R.V.S.S. Barath Tej *et al*., 2023). However, in order to maximize resource utilization while upholding service-level agreements and improving energy efficiency, cloud computing necessitates efficient resource allocation (R. B. Bohn *et al*., 2022).

Allocating computing resources, such as a processor, storage, and backup of several items and programs, in a way that optimizes effectiveness and minimizes energy consumption is referred to as resource allocation in cloud computing (Sadia Ijaz *et al*., 2020). The dynamic nature of operations, the varied nature of resources, and the differing priorities of activities and apps make the allocation of resources in cloud computing a complicated problem (G. Senthilkumar *et al*., 2023). In order to tackle this issue, scholars have put up a number of strategies for allocating resources in cloud computing.

In cloud computing, resource allocation is the process of allocating suitable resources to complete activities requested by customers effectively (Muhammed F Manzoor *et al*., 2020). This entails assigning virtual machines with the attributes that the customers have selected. Users submit their jobs, which may take varying amounts of time to complete. Managing workloads and allocating them to virtualization are additional components of efficient

Department of Computer Science, Adaikalamatha College, Vallam, Thanjavur, Affiliated to Bharathidasan University, Trichy, India.

***Corresponding Author:** Sheena Edavalath, Department of Computer Science, Adaikalamatha College, Vallam, Thanjavur, Affiliated to Bharathidasan University, Trichy, India, E-Mail: sheenaachu@gmail.com

How to cite this article: Edavalath, S., Sundaram, M.S. (2023). Cost-based resource allocation method for efficient allocation of resources in a heterogeneous cloud environment. The Scientific Temper, **14**(4):1339-1344.

Doi: 10.58414/SCIENTIFICTEMPER.2023.14.4.41

Source of support: Nil

Conflict of interest: None.

Received: 09/10/2023 **Accepted:** 12/11/2023 **Published:**27/12/2023

utilization of resources in the cloud (Shriya Pinkulkar *et al*., 2023). The allocation of resources, the amount of time expended, the actions of successors, and the connections with earlier tasks are all important considerations when deciding when to begin or finish a computer activity (Satya Nagamani *et al*., 2019). Furthermore, resource allocation includes tasks like sharing available resources, deciding which resources are best, supplying them, organizing how they will be used, and managing resources as a whole (Sheena Edavalath *et al*., 2023).

Allocating resources to the live process of computing demands is a vital task. The paper proposes a framework for allocating the resources optimally and also proposes a cost-based resource allocation method for the heterogeneous cloud environment.

Related Work

Cloud tasks are allocated to users based on their demands. The distribution of resources is based on the level of demand. It is more efficient for customers to distribute resources based on the cost of the resources. The many cost-based resource allocation strategies that the researcher proposed are examined in this section.

In order to avoid the host becoming overburdened or underloaded with work, M Pandey *et al*. (2017) offered a method for allocating resources based on cost that transfers the computing infrastructure from one host server to another. A market-based resource allocation paradigm was presented by D T Nguyen *et al*. (2021) to multiple rival services at the network's edge in distributed cloud strong edge nodes (EN). In 2019, Sharmila D. *et al*. (2019) proposed a resource allocation strategy that is both cost-effective and efficient.

For cloud services, F Chen. *et al*. (2022) offered a model of resource requirements and pricing that works well. In the article's growth ideology, the provider adjusts the rate and distribution of resources to maximize profit. An energy-based strategy was provided by S K Prasad *et al*. (2023) to maximize income and ensure precise price discovery. Cloud resource providers must spread their potency and ensure that cost is found efficiently in order to maximize revenue.

As an effective and economical framework for resource allocation with pricing, Sujata Swain *et al*. (2023) have developed resource allocation with economical strategy (RAES). They have contrasted their strategy with the greedy algorithm and ECON. For simulation purposes, they alter the average number of CPUs needed by users while maintaining a constant number of users and system capacity. It is found that our approach performs noticeably better in terms of the number of allocations than both the greedy and the simple ECON scheduling algorithms. An effective way to handle the user's deadline violations regarding the second auction is to assign them to a lesser priority cloud, which will guarantee that they can at least access the cloud structures to finish their duties.

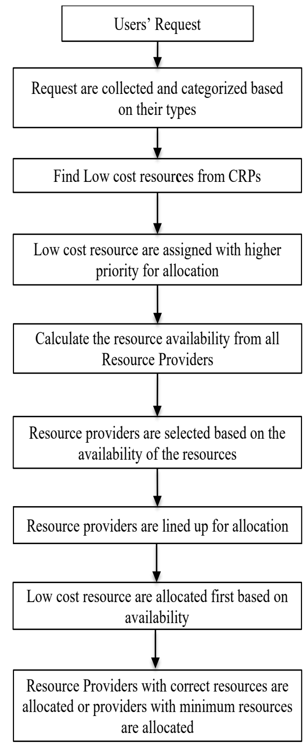

Figure 1: The methodogial diagram of proposed work

Even though many cost and price-efficient resource allocation methods have been proposed over a while, efficient resource allocation is still lacking in the cloud environment. The proposed method emphasizes how much it costs to allocate resources to the intended customer.

Methodology

Figure 1 outlines the proposed ECRA's methodology. There are N distinct cloud resource providers (CRPs) in the context of clouds. All of them are merged into a single, cohesive setting known as a unified heterogeneous resource allocation manager (UHRAM). UHRAM has metadata about the resources that are available and how much they cost from different CSPs. The UHRAM is where CSP resources are gathered and kept up to date. Users of cloud resources can obtain all the resources they require from a single point with the aid of UHRAM. Through their connection to UHRAM, the resource providers determine the cost of the resources and schedule the low-cost resources so they are prepared for allocation. The cost denotes the service cost of the resources. In heterogeneous cloud computing, the suggested resource allocation technique is predicated on a combined architecture for cloud services from a centralized system. Resources from many CSPs are aggregated and maintained by the heterogeneous cloud service.

Framework Design

The cloud environment can provision resources from UHRAM in the proposed framework. The proposed scheme's conceptual diagram is shown in Figure 2. In the cloud environment, there are N cloud resource service providers (CRSPs). They are all interconnected to form UHRAM, a single, integrated environment. Metadata regarding the resources that are available and their costs from various CRPs are stored in UHRAM. The primary goal of the proposed study is to allocate cloud resources according to service costs efficiently. Various steps in the process go into the proposed resource allocation. The resource pooler component of the system collects information about user requests, available resources, and resource costs.

Components of Proposed Framework

- Users make requests in accordance with their needs. Requests may be made to access many services, including infrastructure, software, and platforms.
- Request analyzer examines the many requests that users submit. The request is categorized according to its type and nature.
- Request pooler is an organized list of requests with a schedule, and allocation is called a request pooler. Based on the requests that users create, the request pooler is regularly updated.
- Request repository includes every request that was previously assigned. It keeps track of all the information from the whole report of requests that are sent to and granted by users.
- Task scheduler plans the resources according to the cloud service provider's resource availability and the user's request.
- Resource pooler includes a reserve of ready-to-use resources that are slated for distribution. The demand forecaster's predictions provide the basis for scheduling.
- Resource collector, the resource identifier is linked to or registered with CRSPs in cloud environments. The details of the registered CRSP are kept up to date and regularly checked for current resource availability. The resource manager oversees the resources based on the information contained in the resource identifier.
- Resource manager handles the cloud data center's resources. The resource collector provides it with a comprehensive report that details the current state of the resources that are available within each cloud service provider. In accordance with requests and needs, the resource manager is in charge of adding and withdrawing resources.
- Resource allocator allocates the resources based on which ones have the highest priority. Depending on the demand, resources are ready.

According to the proposed framework, the resource allocation is done by the services of all the components in the framework. Before allocating the resources, resource costs are identified, and low-cost resources. The below pseudo-code describes the procedure of finding low-cost resources.

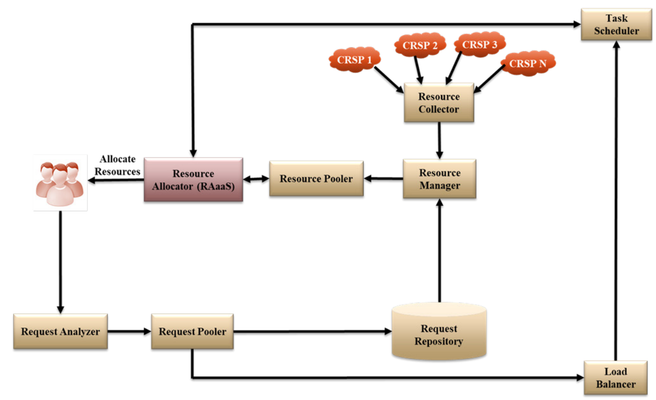

Figure 2: Proposed resource allocation framework design

Pseudo-code: Find the low-cost resource

```
Sub lcstfinder(Rs, TNRsRq, TNRs, Cst, s, p,i)
    /* find low-cost resource*/
Declarations
    Rs: Resources
    TNRsRq: Total number of requests for each resource from all users
    TNRs: Total number of resources
    p: denotes the total number of providers
    s: denotes the number of available resources
    i: iterating loop up to n times
Cst: Cost
    if(TNRsRq[i]<TNRs[i]) then
     lcst←Cst[1][1]
      for j  [1,2,…s] do
       for k  [2,3,…p] do
        if(Rs[k][j]!=0) then
         if (lcst>Cst[k][j]) then
              lcst←Cst[k][j]
              crp←k
              rs←j
     end if
     end if
     next k
     next j
    end if
end sub
```

The low-cost resources are found, and they are scheduled to allocate for demanded requests from the customers. The proposed cost-based allocation ECBA steps are explained as follows. The pseudo-code for efficient cost-based resource allocation is given below.

Pseudo-code: Cost-based Resource Allocation

Global Declaration

lcst: denotes the low cost of resource

crp: denotes the provider who provides low-cost resource

rs: denotes low-cost resource type
tcst: denotes the total cost of the allocated resources
ctcst: denotes cumulative cost of all CRP resources
Sub ecra(Rq,Rs,Cst)

Declarations

u: denotes the number of users
q: denotes the number of types of request
p: denotes the total number of providers
s: denotes the number of available resources
TNRsRq[]: total number of requests for each resource from all users.
TNRs[]: total number of resources available with all CRPs

Input

Rq[u][q]: denotes an array of details about the number of requests from users
Rs[p][s] denotes an array of details about the number of available resources with resource providers.
Cst[p][s]: denotes an array of details about the cost of resources from different resource providers

```
/*find the total number of requests for each resource from all users*/
    for j  [1,2,…q] do
        for k  [1,2,…u] do
        TNRsRq[j]←TNRsRq[j]+Rq[k][j]
        next k
  next j
  /*find the total number of resources available from all C.R.P.s*/
  for j  [1,2,…s] do
        for k  [1,2,…p] do
        TNRs[j]←TNRs[j]+Rs[k][j]
        next k
  next j
                /*Resource Allocation*/
  for i  [1,2,…q] do //iterated for each request
   TRq←TNRsRq[i] // Total numbers of request to be allocated
  while(TNRsRq[i]>0)
      /*Calling cost finder sub-function*/
      lcstfinder(Rs, TNRsRq, TNRs, Cst, s, p,i)
      If (TNRsRq[i]<=Rs[crp][rs])
  TNRsRq[i] ← Number of requests are allocated by number of resources in Rs[crp][rs]
  TNRs[i]← TNRs[i]- TNRsRq[i]
  Rs[crp][rs]← Rs[crp][rs]- TNRsRq[i]
  tcst=tcst+lcst*TNRsRq[rdp]
  TNRsRq[i]← 0
else
  TNRsRq[i] ← Number of requests allocated by available number of resources in Rs[crp][rs]
  TNRs[i] ← TNRs[i]- TNRsRq[i]
  TNRsRq[i] ← TNRsRq TNRsRq [i]- Rs[crp][rs]
  tcst=tcst+lcst*Rs[crp][rs]
  Rs[crp][rs] ← 0
end if
ctcst ←   ctcst+tcst
  end while
  The total cost of the allocated TRq number of requests is ctcst
  next i
end sub
```

Implementation Setup

The cloud environment is used to accomplish the proposed resource allocation. The Microsoft Azure platform is utilized for proposed cloud setup experiments. On the Azure platform, three virtual cloud servers are set up. C# code is used to create a cloud service that is then hosted on the Azure cloud platform. The cloud-hosting service application is linked to the three cloud-hosted servers. Users can create demand for the resources on the deployed servers using the hosted cloud application. When a request is made from a cloud application, the hosted servers act accordingly based on the framework and start allocating the resources. The resources are inbuilt programs created as web services. The proposed ECRA's efficiency is calculated based on different parameters and compared with other techniques.

Result and Discussion

The deployed cloud serves, and the cloud application effectively works based on the proposed framework functionality. Sample requests are generated from the application and execute the proposed cost-based resource allocation. The results are derived based on the total computation time, response time, and percentage of optimal resource allocation. The proposed ECRA is compared with similar cost-based resource allocation methods such as (Sharmila D *et al*., 2019, D T Nguyen *et al*., 2021, F Chen *et al*., 2022).

Figure 3 shows the total computation time consumed for allocating resources. The total computation time denotes the time consumed from the request to the response; consider that the computational time calculation starts from the request received for a resource and the time taken up to the allocation done. The comparison is repeated for different numbers of resources like 10, 20, 30, 40 and 50. The time calculation is in milliseconds. From Figure 3, it is observed that the proposed ECRA efficiently allocates the resources in minimum computational time. Other methods consume more computational time than the proposed ECRA method. Figure 4 shows the response time of the existing and proposed methods. The response time denotes the time taken to allocate a resource. It considers the time of running of the proposed ECRA resource allocation procedure. The results from Figure 4 depict the proposed ECRA consumes a lower response time for allocating the resources. The

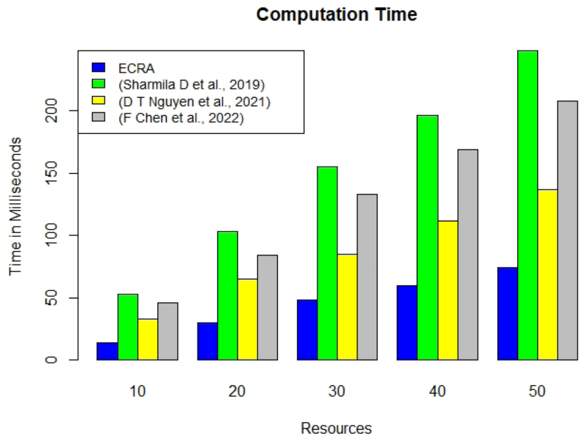

Figure 3: Comparison of computation time of resource allocation

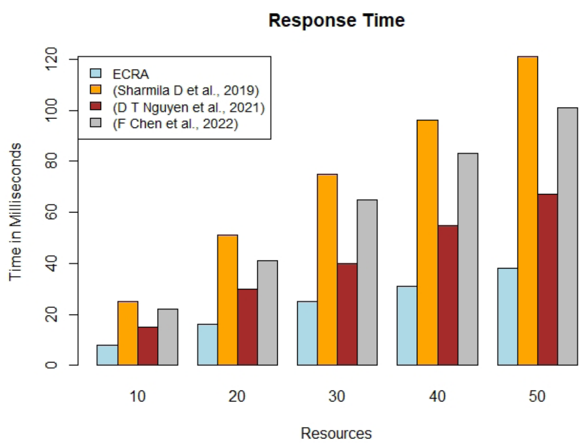

Figure 4: Comparison of response time of resource allocation

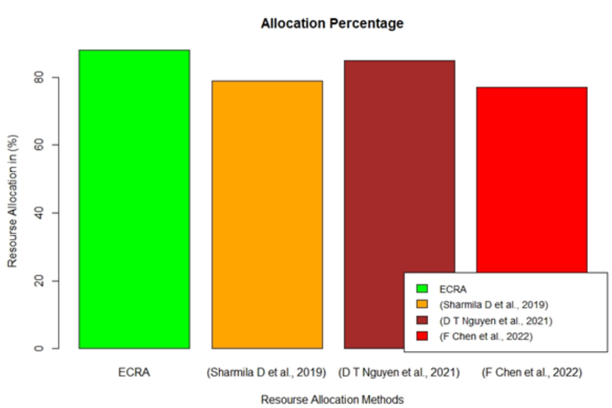

Figure 5: Comparison of percentage of the resource allocation

comparison of other counts of resources also shows that the ECRA has taken a minimum response time than other resource allocation methods such as (Sharmila D *et al*., 2019, D T Nguyen *et al*., 2021, F Chen *et al*., 2022).

Figure 5 shows the comparison of the percentage of resource allocation completed from the available resources. The percentage of resource allocation is calculated as how much resources are allocated from the total available resources. The percentage is calculated by the formula (1) given below.

$$Percentage\ of\ Reource\ Allocation = \frac{Allocated\ Resources}{Total\ Available\ Resources} * 100$$

From the results of Figure 5, it is noted that the proposed method allocates the maximum number of resources from available resources with different cloud resource providers using UHRAM in the proposed framework. Compared to other methods, the ECRA secures a higher percentage of allocation.

Based on the results, the proposed ECRA with the proposed framework, is more efficient in allocating the resource based on the cost of the resources. It is more suitable for the heterogeneous cloud environment. By adopting this framework, cloud resource providers can improve their revenue and increase their customers.

Conclusion

The most significant difficulty in the cloud is resource allocation in a heterogeneous cloud. The proposed framework with ECRA aims to allocate resources that could maximize the efficiency of allocation while minimizing response time. ECRA is a method of cost-based resource allocation. The resource providers declare the cost of resources. In order to integrate into the proposed framework, every provider has registered with the framework and provided all of their available resource details. The low-cost resources are allocated first to improve the full utilization of resources. The results showed that the proposed ECRA method is more efficient in allocating resources in maximum utilization and minimum response time.

Acknowledgment

We sincerely acknowledge the Head of the department, Dr L. Nagarajan, and Dr N. Sumathi, Principal of the institution, for providing the facility to complete this paper successfully.

References

D. T. Nguyen, L. B. Le and V. Bhargava. (2021). Price-Based Resource Allocation for Edge Computing: A Market Equilibrium Approach. *IEEE Transactions on Cloud Computing*, **9(1)**: 302-317. https://doi.org/10.1109/TCC.2018.2844379

F. Chen, Aijun Lu, Harris Wu, Runliang Dou, Xiangyun Wang (2022). Optimal Strategies On Pricing And Resource Allocation For Cloud Services With Service Guarantees. *Elsevier Computers & Industrial Engineering*, **165**: 1-22. https://doi.org/10.1016/j.cie.2022.107957

G. Senthilkumar, K. Tamilarasi, N. Velmurugan, and J. K. Periasamy. (2023). Resource Allocation in Cloud Computing. *Journal of Advances in Information Technology*, **14(5)**: 1063-1073. https://doi.org/10.12720/jait.14.5.1063-1072

M. Pandey, Sachin Kumar Verma. (2017). Cost Based Resource Allocation Strategy for The Cloud Computing Environment, *IEEE International Conference On Computing, Communication And Networking Technologies*: 1-7. https://doi.org/10.1109/ICCCNT.2017.8204170

Muhammad Faraz Manzoor , Adnan Abid , Muhammad Shoaib Farooq , Naeem A. Nawaz , Uzma Farooq. (2020). Resource Allocation Techniques in Cloud Computing: A Review and Future Directions. *ELEKTRONIKA IR ELEKTROTECHNIKA*, **26(6)**: 40-51. http://dx.doi.org/10.5755/j01.eie.26.6.25865

R. B. Bohn, R. Chaparadza, M. Elkotob and T. Choi. (2022). The Path to Cloud Federation through Standardization. *International Conference on Information and Communication Technology Convergence*, Korea: 942-946. https://doi.org/10.1109/ICTC55196.2022.9952660

R.R.V.S.S. Barath Tej. (2023). A Comprehensive Study of Resource Allocation in Cloud Computing Environments. *International Journal of Research Publication and Reviews*, **4(5)**: 2035-2038.

S. k. Prasad, Puja S. Prasad & Nupur Neti. (2023). Energy-Efficient Resource Allocation With A Combinatorial Auction Pricing Mechanism. *Springer International Journal of Information Technology*, **15**: 511-518. https://doi.org/10.1007/s41870-022-01110-9

Sadia Ijaz, Tauqeer Safdar, and Amanullah Khan. (2020). Challenges and Limitations of Resource Allocation in Cloud Computing. *Springer International Conference on Intelligent Technologies and Applications, Intelligent Technologies and Applications*: 723-737. https://doi.org/10.1007/978-981-15-5232-8_62

Sharmila D., Kanimozhi S (2019). Cost-Effective Approach for Vivacious Resource Allocation in Cloud Environment. *International Journal of Engineering and Advanced Technology*, **8(3S)**: 318-320. https://doi.org/10.3390/electronics11213639

Sheena Edavalath, Manikandasaran S. Sundaram. (2023). M.A.R.C.R.: Method of allocating resources based on the cost of the resources in a heterogeneous cloud environment. *The Scientific Temper*, **14(3)**: 576-581. https://doi.org/10.58414/SCIENTIFICTEMPER.2023.14.3.03

Shriya Pingulkar, Aryaman Tiwary, Shruti Tyagi. (2023). Resource Allocation Techniques in Cloud Computing: A Comprehensive Review. *International Journal of Engineering Research & Technology*, **12(7)**: 350-357.

Sujata Swain, Bandyopadhyay Anjan, Dipti Dash and Arup Roy. (2023). Resource Allocation in Cloud Computing with Economical Strategic Setting. *International Journal of Computing and Digital Systems*, Preprint: 189-198.

T Satya Nagamani, K N V S K Vijaya Lakshmi and B Lalitha Bhavani. (2019). A new dynamic and enhanced resource allocation algorithm in cloud computing. *International conference on computer vision and machine learning, I.O.P. Conf. Series: Journal of Physics, **1228***: 1-8. https://doi.org/10.1088/1742-6596/1228/1/012033