The Scientific Temper (2023) Vol. 14 (3): 576-581
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2023.14.3.03
https://scientifictemper.com/

RESEARCH ARTICLE

MARCR: Method of allocating resources based on cost of the resources in a heterogeneous cloud environment

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

Abstract

The cloud is an intelligent technology that provides requested services to users. It offers unlimited services for the users. Many small and medium-scale industries are startup their businesses to the next level using cloud computing. The services have been provided to the users by allocating the requested resources. Allocating resources without waste and with the finest allocation is a critical task in the cloud. This paper proposes a method for allocating resources using the cost of the resource. Resource allocation follows a priority system when allocating resources. The proposed method gives priority to low-cost resources. The cost denotes the service cost of the resource. The requested resource is assigned to the user by the CSP, who provides the specific resource at a low cost. This proposed method suggests a UHRAM for collecting and allocating the resources from the different CSPs. UHRAM is a centralized hub for delivering requested resources to users, and it maintains a repository of details about the resources from all CSPs in the heterogeneous cloud. The proposed method is implemented with the user's data. The results from the comparison show that the proposed cost-based resource allocation method is more efficient than existing methods.

Keywords: Cloud computing, Resource allocation, Cost-based allocation, Heterogeneous cloud,

Introduction

Cloud computing is an intelligent computing paradigm that is only getting started in the 21st century. The vast capacity of computing resources found in the cloud is made available to consumers as a service (Aishwarya S *et al.*, 2017). There are now available technologies that are more beneficial for online computing. However, they have some resource provisioning restrictions (Lokesh A *et al.*, 2022). The cloud was created to provide limitless on-demand provisioning of computing resources. The cloud is not a brand-new technology; it has developed from earlier technologies (R. *et al.*, 2022). The cloud is increasingly appealing since it offers services at a low cost (L. Yala *et al.*, 2018). Cloud computing is a type of business computing mostly utilized and supported by small and medium-sized businesses to grow their computer businesses. The computing source is made up of cloud service providers, each of whom offers a variety of resources and services. Users can connect with various service providers to meet their computing needs. Users who consume services are charged based on how much of those services they use (Imran *et al.*, 2017).

Several cloud service providers (CSPs) are in action to provide services like software, platforms, and infrastructure in a cloud environment. There is no centralized hub to consume all the cloud services (S. *et al.*, 2021). If the cloud may have a single or centralized hub of service, then it is more comfortable for all users to use and contact a centralized service hub to consume all their required computing resources (Xinping Xu *et al.*, 2020). This paper suggests a centralized hub to provide all cloud computing services in a single window. Hence, the users do not need to approach each cloud service to consume the services. The research work considered resource allocation in the heterogeneous cloud environment. Allocating cloud resources based on users' requirements is critical in a heterogeneous cloud environment (Sadia Ijaz *et al.*, 2020).

Department of Computer Science, Adaikalamatha College, Vallam, Thanjavur. Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India.

***Corresponding Author:** Sheena Edavalath, Department of Computer Science, Adaikalamatha College, Vallam, Thanjavur. Affiliated to Bharathidasan University, Tiruchirappalli, Tamil Nadu, India, E-Mail: sheenaachu20@gmail.com

How to cite this article: Edavalath, S., Sundaram, M.S. (2023). MARCR: Method of allocating resources based on cost of the resources in a heterogeneous cloud environment. The Scientific Temper, **14**(3): 576-581.

Doi: 10.58414/SCIENTIFICTEMPER.2023.14.3.03

Source of support: Nil

Conflict of interest: None.

Received: 20/07/2023 **Accepted:** 14/08/2023 **Published** : 25/09/2023

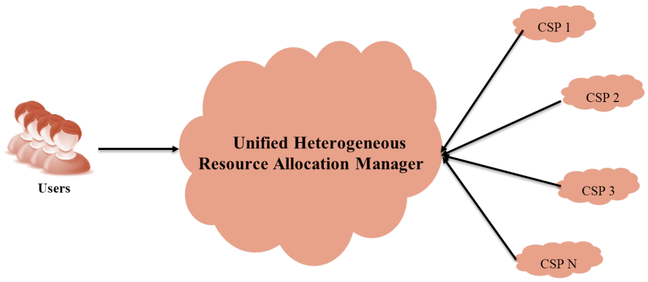

Figure 1: Unified resource allocation in heterogeneous cloud environment

Figure 1 shows the abstract representation of resource collection and allocation in the heterogeneous cloud environment design. There are N numbers of cloud service providers in the cloud environment. All are connected into one unified, integrated environment called unified heterogeneous resource allocation manager (UHRAM). UHRAM has metadata about the resource available and the cost of the resources from different CSPs. Resources from CSPs are collected and maintained in the UHRAM. UHRAM helps cloud resource users to get all the necessary resources from a single counter. The resource providers, connected with UHRAM, identify the cost of the resources, and the low-cost resources are scheduled and ready for allocation. This paper proposes a resource allocation method based on the scenario described in Figure 1. The paper considers the challenge of allocating resources efficiently based on the cost of the resources. The cost denotes the service cost of the resources. The proposed resource allocation method is based on a combined architecture for cloud services from a centralized system in heterogeneous cloud computing. Resources from different CSPs are cumulatively maintained in the heterogeneous cloud service.

Related Work

Cloud resources are allocated to the users based on their requests. The allocation is based on the priority of the resources. The priority may be cost, demand, energy, or auction; allocating the resource based on the cost is more efficient for the users. The researcher proposes different cost and price-based resource allocation methods, which this section discusses.

Manish P *et al.* (Manish Pandey *et al.*, 2017) proposed a cost-based resource allocation approach that migrates the virtual machine from one physical host (physical machine) to another machine to evade the work overloaded and under load situation of the host. A cost-based resource allocation strategy will allocate the VM first, paying the provider more for using the cloud resources. Hence, the cost metric is used to place the VM. First, they calculate the bill, which the customer pays for using the VM, and arrange all VM in descending order according to the cost. While assigning a VM to PM, they maintained a list that satisfied the VM requirements. Then they assigned VM to all available PMs one by one, calculated the utilization of PM before and after assignment, and maintained a list. This proposed approach is applied in the cloud simulator tool CloudSim to assess the quality of the proposed approach performance.

Duong T N *et al.* (Duong *et al.*, 2021) proposed a market-based resource allocation framework in heterogeneous cloud capacity-limited edge nodes (EN) to multiple competing services at the network edge. By adequately costing the geographically circulated ENs, the proposed framework produces a market equilibrium (ME) explanation that increases the edge computing resource consumption and allocates optimal resource packages to the services given their budget limitations. When the utility of service is well-defined as the extreme revenue that the service can reach from its resource allocation, the equilibrium can be computed centrally by solving the Eisenberg-Gale (EG) convex program. Further, it shows that the equilibrium allocation is Pareto-optimal and satisfies desired fairness properties, including sharing incentive, proportionality, and envy-freeness. Also, two distributed algorithms, which efficiently converge to an ME, are introduced. When each service aims to maximize its net profit (i.e., revenue minus cost) instead of the revenue, derive a new convex optimization problem and rigorously prove that its solution is exactly a market equilibrium.

Kanimozhi *et al.* (Kanimozhi S *et al.*, 2019) proposed a cost-effective resource allocation method. This method manages various double auction protocols devised. Multiple users and providers can access the resource honestly and securely through that protocol. A double auction protocol is designed for multiple consumers and providers to manage multiple task handling. A price formation, price matching, and price prediction mechanism is devised. A good system has been designed which satisfies both the provider and consumer. It also helps to remove the participant who is not truthful. Based on the price quoted by each participant, the winner can be determined with the paddy field algorithm.

The consumers consider heterogeneous demands, and only one resource is provided through an online auction mechanism. Cloud resource pricing is done with truthful and computationally efficient auctions. They are suppressing dishonest participants through confidential mechanisms. The cost for execution problems is highly reduced.

Fuzan C *et al.* (Fuzan Chen *et al.*, 2022) proposed an economic pricing and resource allocation model for cloud services. The author proposed an economic model in which the provider adjusts both the price and resource allocation to maximize profit. They use queuing theory to model the user-service interaction and seek to protect servers from overloading and derive the provider's optimal pricing and resource allocation decisions. The impact of service guarantee on profit and service fulfillment rate is examined, and high compensation does not always hurt cloud service providers' profit. Their findings can guide cloud providers to develop virtuous profitability modes.

Sanjay K P *et al.* (Sanjay K Prasad *et al.*, 2022) presented an energy-based approach to ensure truthful price discovery and maximize revenue. Cloud resource providers must ensure efficient price discovery and allocate their capacity to maximize revenue. This proposed energy-aware heuristics to shut down the new servers to minimize the total energy cost for the resource provider. Additionally, this approach has been validated by conducting a simulation. The results demonstrate that the energy-based mechanism has immense potential, offering significantly better revenue realization.

Methodology

In the process of proposed resource allocation, the priority method is used for selecting requests and resources. The priority methods are Request-Priority (ReqPty) and Resource-Priority (ResPty). ReqPty is given to the user's requests, and ResPty is given to the resources from CSPs. Each request from users is for a resource from the cloud service provider. Hence a request is denoted as a resource request. The priority to the request ReqPty denotes that among the different resource requests, which is selected as a privileged resource request for allocation? The priority of resource ResPty denotes that among different CSPs, which CSP's resources are selected as a privileged resource for allocation? This proposed technique aims to minimize the utilization cost of resources allocated to the users. The proposed technique uses the ResPty resources priority method to select resources based on the service cost of the resources. The minimum service cost resources are given maximum priority to allocate first. The minimum cost of resources is calculated using Eq. (1)

ResPty = mincost[Res1,Res2,…ResN] …(Eq.1)

The proposed work mainly allocates cloud resources based on its service cost. Initially, the total number of resources with all service providers is calculated, and find the total number of resource requests from the users. The resources are analyzed for their availability with the CSPs. The CSPs with the minimum number of resources are selected for allocation. The resources for the request are selected based on lower cost. The resources are allocated based on the user's request. The resource selection first considers the cost and then considers resource providers with the minimum number of resources. CSPs with fewer resources are allocated first, and then the remaining requests for the same resources are allocated from the next service provider with minimum cost. In the same way, all resource requests are allocated by considering the cost and CSPs with a minimum or equal number of resources. Hence, all the resources from the CSPs are fully utilized and allocated.

MARCR System Design

The proposed MARCR mainly concentrates on allocating cloud resources efficiently based on the service cost. The procedure involved in the proposed resource allocation consists of different steps. Details of user requests, resources available, and cost of the resources are gathered from the CSPs. The proposed resource allocation technique is based on the resource priority method. It gives priority to the resources which are provided at a low cost. The MARCR aims to allocate the resources optimally to the users. The resources should not keep on futile, and also the cost of the service should be low. The proposed technique's result gives the providers income and minimum costs for the users. The proposed technique sorts out the available resources in ascending order based on cost and allocates the resources first from the providers with the minimum number of resources. In this process, the providers with minimum resources can allocate the services first to the required users.

The MARCR is defined mathematically. This section gives a mathematical representation of the proposed MARCR and presents the mathematical proof of the efficiency of the proposed MARCR, as per the procedure of the proposed technique. The users' request is calculated and analyzed first, and then resources are calculated for the availability of the resources.

Mathematical Procedure for Request Calculation

The mathematical notation used in the proposed MARCR for calculating user requests is as follows,

Usri → denotes users

TltUsr → denotes the total number of users

The total users are calculated using Eq. (2)

TltUsr = count[Usri] …(Eq.2)

Reqi→ denotes requests

Resi→ denotes resources

tyReq1, tyReq2, tyReq3…tyReqn →denotes types of the request for the resources

tyRes1, tyRes2, tyRes3…tyResn → denotes the type of Resources available in the CSPs

The total request count of a request is calculated by Eq. (3)

tyReqi = count [Reqi] …(Eq.3)

The total resource count of a resource is calculated by Eq. (4)

tyResi = count [Reqi] …(Eq. 4)

The number of requests for the same resource from each Usri is calculated; it denotes that each user Usri can request the same resource number of times. The below Eq. (5) denotes the total number of requests for each resource from all users.

$$Usri\left[Resj\left[\sum_{k=1}^{i} tyReqk\right]\right] \quad \ldots(\text{Eq. }5)$$

i→denotes number of users

j→ denotes the types of resource request

k→ denotes the number of requests from users

Resource requests are calculated from UHRAM suggested; UHRAM maintains the metadata of requests received and resource availability.

The total number of requests by an Usri is calculated by Eq. (6),

$$Usri\left[\sum_{k=1}^{l} tyReqj\right] \quad \text{...(Eq. 6)}$$

The total number of requests by all the users is calculated by Eq. (7); it denotes all requests from all the users irrespective of resources. It can be found by adding the total number of request by Usri.

$$\sum_{i=1}^{n} Usri\left[\sum_{k=1}^{l} tyReqj\right] \quad \text{...(Eq. 7)}$$

The above notations denote users, requests, and resources in the cloud environment. It also denotes the calculation of requests for each resource and calculation requests by all users. The following section describes the notation used for calculating the resources.

Mathematical Procedure for Resource Calculation

The mathematical notation used in the proposed MARCR for calculating CSPs with available resources is as follows,

CSPi → denotes cloud service providers

TCSP → denotes the total number of CSPs

The total CSPs are calculated using Eq. (8)

TCSP=count [CSPi] ...(Eq.8)

CiRsj← Types of resources available with CSPi

CiRsNj← Array of the name of the Resources

TyCiRs ←Total types of resources available with a single CSP

CiRs ← Counting of CiRsj

The total available resources from all CSPs are calculated by Eq. (9),

$$CSPi\left[Resj\left[\sum_{j=1}^{x} tyResj\right]\right] \quad \text{...(Eq.9)}$$

i→denotes numbers of CRPs

j→ denotes types of resources

Cst_{kj}← denotes the cost of resources available with different CSPs, k denotes different types of resources, and j denotes the cost of each k^{th} resource. The low resource cost among different resources is calculated by Eq. (10),

Low resource cost= min({f(x,y):x=1…j;y=1…k}) …(Eq.10)

The equation used to find the provider who provides the requested resource with the lowest cost, which is the minimum of a set of resource costs. Each value in the set results from a function f(x,y); the x range is between 1 and j, and y is between 1 and k.

The total number of all resources available with a CSP is calculated using Eq. (11),

$$CSPi\left[\sum_{j=1}^{x} tyResj\right] \quad \text{...(Eq.11)}$$

The CSPs fix the cost of each resource. The proposed system does not calculate the cost of the system, and it allocates the resources based on the service cost.

Procedure of* MARCR *Resource Allocation

The resource allocation begins after the calculation of the requests from the user and available resources with the CSPs. The procedures involved in resource allocation of the proposed MARCR are as follows:-

- Determine the resource service cost of each resource from all CSPs
- Determine the total number of resources available with CSPs
- Determine the total number of requests requested by the users
- Check whether the resource available is greater than the resource request received from the users.
- Calculate the cost of resources from different CSP
- Assemble the CSP in an order based on the availability and cost of the resources.
- Consider the minimum number of resources and the lowest cost.
- Allocates the low-cost resource from the CSPs with the minimum number of resources.
- If more requests are needed for the same resource, the next CSPs with the lowest cost are selected for allocation.
- The process is repeated up to allocating resources for completing all user requests.

The described procedures are executed to allocate the resources efficiently. The proposed resource allocation technique is implemented in the cloud platform.

Implementation Setup

The proposed resource allocation is implemented in the cloud environment. An implementation environment is created for the research work. Windows Azure platform is used for experiment proposed cloud research. Initially, three cloud servers are deployed on the Azure platform. A cloud service is developed with the help of C# coding and hosted in the cloud Azure platform. The three hosted cloud servers are connected to the hosted cloud service application. The cloud application allows users to generate requests for the resources in the deployed servers. The deployed servers are implemented remotely with different programs when a request is raised from the cloud application. The installed cloud application effectively works with the proposed functionality. Sample data are taken as input and execute the proposed cost-based resource allocation.

Experiment with Sample Data

The proposed MARCR is executed with sample data. Table 1 shows data that are considered for the experiment. Table data are considered resource requests for different resources.

Table 1: User's resource requests

Request types	*Usr1*	*Usr2*	*Usr3*	*Usr4*	*Total*
Rq1	3	4	3	3	13
Rq2	2	0	2	2	6
Rq3	0	0	2	4	6
Total	5	4	7	9	25

Table 2: Cost and available resources in CSPs

Resources and Availability	CSP1			CSP2			CSP3			CSP4			Total
	Res1	Res2	Res3	Res1	Res2	Res3	Res1	Res2	Res3	Res1	Res2	Res3	
Resources Available	3	3	3	3	2	0	4	0	2	4	2	3	29
Cost/Service	7	15	20	5	12	-	6	-	21	8	11	22	

Table 3: Resource allocation based on the proposed cost-based method

Users	Request Types	Number of Requests	Resource Allocation	Remaining Resource	Cost/ Service	Total Cost
Usr1	Req1	3	CSP2(3)	0	5	15
	Req2	2	CSP4(2)	0	12	24
	Req3	0	-	0	-	-
Usr2	Req1	4	CSP3(4)	0	6	24
	Req2	0	-	0	-	-
	Req3	0	-	0	-	-
Usr3	Req1	3	CSP1(3)	0	7	21
	Req2	2	CSP2(2)	0	15	30
	Req3	2	CSP1(2)	1*	20	40
Usr4	Req1	3	CSP4(3)	1	8	24
	Req2	2	CSP1(2)	1	11	22
	Req3	4	CSP1(1), CSP3(2), CSP4(1)	2	22	88
Total		25		4		288

*-Remaining 1 resource from CSP1 is allocated later for Usr4. So CSP1 has no remaining resource for Req3 request for resource3.

The sampling considered three resource requests named Req1, Req2, and Req3. The requests are generated from four users called Usr1, Usr2, Usr3, and Usr4. Based on the mathematical procedure, the total request for each resource from all users is calculated, and the total request from each user for all resources is calculated and shown in Table 1.

The resource availability is shown in Table 2. It shows the number of CSPs with available resources and also shows the cost of the resources. The service cost is denoted in Rupee.

The proposed cost-based resource allocation is executed for allocating the requested resources based on Table 1. The resource availability and cost of each resource are shown in Table 2. Table 3 shows the resource allocation based on the proposed cost-based method.

Consider the user Usr1, Usr1 requests Req1, Req2, and Req3 at 3, 2, and 0 number of requests, respectively. For the resource request Req1, it allocates resource Res1 from the CSP2 because compared to other CSPs' cost for Res1, CSP2 provides the Res1 in minimum cost at Rs. 5. However, CSP1, CSP3, and CSP4 provide the same Res1 at Rs. 7, 6, and 8, respectively. In this same method, another resource request is allocated based on the cost of the resources.

Implementation Results

The proposed MARCR is implemented and compared with similar cost-based resource allocation methods. A comparison of the proposed and existing methods is based on the percentage of the resource allocation and response time for the resource request. Table 4 shows the comparison results of the proposed and existing resource allocation methods concerning the percentages of resource allocation.

Table 4: Comparison of percentage of the resource allocation

S.No	Resource Allocation Methods	Percentage (%)
1.	(Sadia Ijaz *et al.*, 2020)	76
2.	(Duong Tung Nguyen *et al.*, 2021)	82
3.	(R. B. Bohn *et al.*, 2022)	74
4.	MARCR	85

Table 5: Response time comparison

Methods	Total Resources				
	10	20	30	40	50
	Milliseconds				
(Sadia Ijaz *et al.*, 2020)	15	30	40	55	67
(Duong Tung Nguyen *et al.*, 2021)	22	41	64	83	101
(R. B. Bohn *et al.*, 2022)	25	51	75	96	121
MARCR	8	16	14	33	42

The results shown in Table 4 show that the proposed MARCR efficiently allocates the resources based on the cost of the resource. Table 5 shows the response time of the existing and proposed techniques.

The comparison of the proposed and existing techniques concerning the response time is shown in Table 5. The results show that the proposed method takes a minimum response time to allocate the resources. Compared to the existing methods, the proposed cost-based resource allocation has maximum resource utilization and consumes minimum response time with minimal resource cost.

Conclusion

Cloud has much attraction in providing service to the users. Providing services to users is allocating all requested resources with the minimum waiting time. Allocating resources is a critical job in a cloud environment. This paper considered the cost of the resources for allocation. All resource providers are connected with heterogeneous cloud systems, and the availability of the resources is maintained for efficient allocation. The proposed method is a resource priority-based allocation method. The low-cost resources are given maximum priority for allocation, and those resources are scheduled for allocation. The mathematical notations are described for calculating the number of user requests, available resources with CSPs, and allocation procedures to allocate the resources. The proposed technique is implemented using the windows azure platform cloud in a real-time cloud setup. The proposed method efficiently allocates resources by considering the cost of the resources. The results comparison shows that the MARCR efficiently allocates resources in the heterogeneous cloud environment.

References

Aishwarya Soni, Muzammil Hasan. (2017). Time and cost-based Resource Provisioning Mechanism in Cloud Computing. *International Journal of Advanced Research in Computer Science*, *8*(5), 288–292.

Dr. Lokesh A, Dr. Mohamed Saleem, Bhupendra Kumar Swar, Saurav Kumar, Rahul Sharma. (2022). Cloud Computing: The Emerging Technology. *International Research Journal of Modernization in Engineering Technology and Science*, *4*(10), 1503–1510.

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. DOI: 10.1109/ICTC55196.2022.9952660.

L. Yala, P. A. Frangoudis, G. Lucarelli and A. Ksentini. (2018). Cost and Availability Aware Resource Allocation and Virtual Function Placement for CDNaaS Provision. *IEEE Transactions on Network and Service Management*, *15*(4), 1334-1348.

Imran Raza Khan and Mahfooz Alam. (2017). Cloud Computing: Issues and Future Direction. *International Conference on Global Sci-Tech*, *9*, 1-8.

S. De Capitani di Vimercati, S. Foresti, G. Livraga, V. Piuri and P. Samarati. (2021). Security-Aware Data Allocation in Multicloud Scenarios. *IEEE Transactions on Dependable and Secure Computing*, *18*(5), 2456-2468.

Xinping Xu, Wenxin Li, Heng Qi, Junxiao Wang, & Keqiu Li. (2020). Latency-Constrained Cost-Minimized Request Allocation for Geo-Distributed Cloud Services. *IEEE Open Journal of Communication Society*, *1*, 125-132.

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

Manish 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.

D. T. Nguyen, L. B. Le & V. Bhargava. (2021). Price-Based Resource Allocation for Edge Computing: A Market Equilibrium Approach. *IEEE Transactions on Cloud Computing*, *9*(1), 302–317.

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

Fuzan 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.

Sanjay k. Prasad, Puja S. Prasad & Nupur Neti. (2022). Energy-Efficient Resource Allocation With A Combinatorial Auction Pricing Mechanism. *International Journal of Information Technology*, *15*, 1–16.