Journal of Positive School Psychology
2022, Vol. 6, No. 2, 4411–4422 http://journalppw.com

Implementation of Multi agent and Multi source dynamic Resource allocation for IoT based cloud computing environment

J. Mahalakshmi[1], P. Venkata Krishna[2]

[1] *Department of Computer Science, Bharathiar University, Tamilnadu, India*
[2] *Senior Member IEEE, Department of Computing Science, Sri Padmavati Mahila Visvavidyalayam, Tirupati, India*
[1] *Mahalakshmi1203@gmail.com,* [2]*pvk@spmvv.ac.in*

Abstract

IoT based cloud computing system faces new challenges every day, due to the complex structure of system clusters and high volume of data processed by the systems. The ability of acquiring resources in an elastic manner is considered as the primary rationale for adopting IoT based cloud computing system. Elasticity mainly supports the facility to grow and shrink the virtual resources dynamically according to the requirement of IoT based cloud users. This article proposes a Multi-source framework using multisource QoS based Resource Allocation (QRA) and Multi agent Dynamic resource allocation (MADRA) Algorithm for increasing the flexibility and efficiency of resource allocation using virtualization. In the proposed framework, each source monitors and investigates all requests and processor availability before finding and allocating the resource. QRA algorithm is used to utilize the resources effectively and reduce the congestion by using vii multiple intermediate layers. On the average, the proposed framework approach provides 20.52% improvement in response time, 13.29% reduction in power consumption, and 1% error in prediction when compared to existing Efficient Resource Allocation (ERA) approaches. From simulation results, when the input load frequency is 200 Hz the percentage of resource allocation is 99.18% which is high when compared to that existing approaches. Experimental results indicate that the proposed approach is capable for more user requests and it improves QoS parameters such as completion time, response time and power consumption.

Keywords— Dynamic resource allocation, Multi agents, Multi sources.

INTRODUCTION

In IoT based cloud computing technique [1], the services are abstracted and provided to the users over the internet in a distributed manner and these services are accessed through the networks. Its essential goal is to serve all users with high reliability and better performance. This technique becomes a good choice for several business contexts [2]. Users in a IoT based cloud are allowed to attain the resources by initializing the QoS. IoT based cloud consumers request various services based on their dynamic needs in a IoT based cloud computing environment. Resources are used on a rental basis instead of owning the resources for their business. This saves the cost and reduces risks in managing the resources.

Nowadays any business or organizations [3] are using ―IoT based cloud for their websites. There are many more uses of IoT based cloud in current market for example like popular social networking websites: Facebook, YouTube, Twitter, LinkedIn, Google Plus, ClassMates and many more are hosted on IoT based cloud as well as email providers: Google Apps, Yahoo Zimbra, PanTerra Networks, Microsoft Exchange Online [4]. One can get their computed solution from anywhere at any time. When we surfing on any website or search engine we find IoT based cloud everywhere,

Each and every IT magazine describes the new technologies introduced for IoT based cloud. We can upload documents on IoT based cloud for sharing. Online games required very much space but IoT based cloud makes it easy for online games providers, they can host their online games on IoT based cloud with minimum cost. From the survey, nowadays many developers from Microsoft, Amazon, AT&T, Google, GoGrid, RightScale, NetSuite, Enomaly and many more are working on IoT based cloud services [5]. It is a type of computing system which is basically an internet-based that gives access and services that are not on local computers or datacenter but reside on remote location. These services [6] are provided as per the consumption of particular services by the consumer respect to pay-as-you-go. IoT based cloud should ensure that all requested services are available for the end users. Limited resource availability in a IoT based cloud makes IoT based cloud provider‘s job more difficult. Allocation of resources and satisfying the user QoS requirements are important issues in an IoT based cloud computing environment.

This article proposes a new approach in which Hierarchical Agglomerative clustering algorithm [7] is applied to group resources. The grouping of resources helps to optimize the time to search the required resources from a pool of resources. Since the resources are grouped, the required resource to meet the requirements of a request is identified and allocated in short time. Resource allocation [8] is one of the major challenging issues in Cloud environment. As very large numbers of heterogeneous resources are used by thousands of clients in real time, it is essential to allocate resources dynamically. With diverse applications of Cloud, allocation of various resources such as memory, server, and bandwidth to jobs is an extremely complex task. The previous article contains the discussion about the various resource allocation techniques and its QoS parameter[9] such as cost, completion time, and response time and energy consumption for resource allocation. It has also discussed about the merits and its demerits. From the above discussion, it is clear and noticed that most of the approaches are not improving the performance in terms of response time, energy consumption, accuracy and scalability. Also, all the existing approaches provide only partial solution to the entire Cloud environment [10].

Literature survey

In [11] authors suggested a systematic framework to monitor, analyze and improve the performance of system in the present research. Particularly, a radial basis neural network is formed in order to modify the simulated tasks with abstract specifications to particular resource needs in terms of their qualities and quantities.

In [12] authors improvised a multi-objective dynamic programming technique to reduce the estimation cost when meeting the latency demands with good support possible. The simulated experimental result indicates that during the occurrence of disturbance, the adjustment of window size and settlement of offloading technique, the performance of ARHOS is considerably superior when compare with the static optimal offloading mechanism.

In [13] authors have observed the much professed and popularly used technology of the advanced era: the Internet of Things (IoT). In the case of IoT, different smart objects such as smart phones, embedded devices, smart sensors, and PDAs share the data with each other irrespective of their geographical positions with the help of the Internet. The quantity of data that is generated via these linked smart objects would be in the order of zetta bytes in future.

In [14] authors have offered a control structure which concentrates on the troubles of load balancing and allocation of resource of the combined web services in the infrastructure of cloud computing. The intended method aimed at succeeding the needs of the customer defined in a SLA whereas exploiting the usage of server. An ordered two layer controller was recognized.

In [15] authors have presented a computation algorithm based on a fuzzy analytic hierarchy procedure and offered an innovative vibrant permission alteration method. The experimental results revealed that their technique offered the base for assigning the resources dynamically. It also enhanced the exploitation ratio of the cloud computing resources and defends the safety of cloud computing environment.

In [16] authors have proposed another algorithm concerning the assignment of job to dissimilar VMs within a Cloud Data Center by means of ETC matrix, MTC matrix and classical scheduling strategy. It was a portion of that entire effort. The ETC and MTC matrices assist to plot the jobs to the suitable VM, which lessens the whole response time and waiting time of jobs.

In [17] authors have presented another approach which considers the scale, dynamics, and heterogeneity of network facilities in the dispersed cloud networks with the intention to suit online solutions. They improved the overall objectives such as high resource utilization through local communication based on the statistics among service requests.

In [18] authors have developed a dynamic scheduling enabled scheduler, called Cress, which examine the qualities of cloud applications and create suitable decisions in assuring the needs of performance and attaining more utilization concurrently. Cress utilizes a dynamic depiction language to define the multidimensional needs of the resource inhibited jobs.

In [19] authors have proposed another algorithm that is optimal and appropriate to the cloud data centers. They proposed a dynamic virtual network allotment algorithm that is appropriate to the cloud data center which used suboptimal solutions for allocation. The numerical results of their experiments revealed that the algorithm performs superior to the conventional technique in terms of total utility.

In [20] authors have proposed a resource allocation scheme for single cloud providers. It deals with the query on how to better harmonize the customer demand based upon the supply and price so as to enhance the revenue of the providers and customer satisfaction by decreasing the cost of energy.

Hierarchical Ordering of Resources

The hierarchical tree is built using structures computed from the resources. After the computation, the resources are then assigned to the requests in the hierarchical order. In the proposed model, Krill-Herd algorithm utilizes the computed average values of both requests and resources for optimal allocation of resources in hierarchical order. The resource allocation depends upon the energy p (x) / and memory (U) attributes of every individual resources. The energy and the memory parameters are computed for each resource in the initial source. Based on the values, resources are arranged in a hierarchical order. After that, the resources are allocated to the tasks according to their position in the hierarchy. Figure 1 shows how the resources are allocated in hierarchical manner

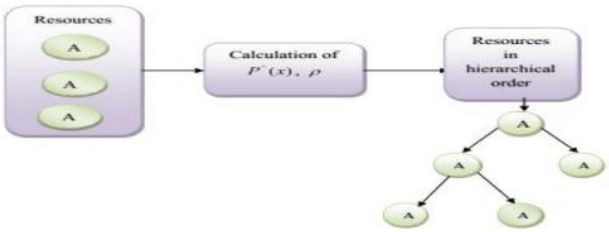

Figure 1: Hierarchical Arrangements of Resources

Measuring Network Bandwidth: It is termed as the ratio of time of request to the work load of the request. It influences how rapidly the information/ request is uploaded to the cloud. Bandwidth of the request determines the quantity of data pass through the network immediately. It influences the time it takes to download or upload the information into the cloud.

COST: It is the amount to be paid prior to the request to be processed.

Memory: Data centers include different kind of memory modules such as RAM and Virtualized memory. The monitoring systems will not provide information regarding how often the memory modules are accessed. The subsequent steps are followed to compute the values including U which is used to indicate the state.

Proposed Multi-agent based dynamic resource allocation

To increase the efficiency of the Cloud environment, the Cloud computing needs to satisfy the following concerns as: Reduce the energy consumption, Increase the process speed

and Increase the Cloud security. Cloud helps to migrate, share and compute the data from a remote location with less impact on system performance. Resource allocation is a challenging issue in Cloud computing since a remote user can access resources from anywhere at any time. It is impossible to access resources directly from remote, but it is possible by using various interfacing protocols such as SOAP, WEB-APIs, and virtualization. Resources can be allocated statically or dynamically for user requests. The dynamic resource allocation technique is used to reduce the time and energy. The proposed solution improves the energy by choosing best nearest resource and allocates it to the client. The aim of the resource allocation in Cloud is either optimized or improved QoS. Most of the resource allocation challenges are related to energy efficiency. Resource allocation depends on the work assigned to Cloud and it should be power consumption based, cost effective and time effective. The overall performance of the Cloud environment including clients and servers should be increased and it should be flexible to other applications.

The main aim of this article is to provide a fast, efficient and dynamic resource allocation method for Cloud environment. Cloud users submit their requests to the server from anywhere in the world. Due to more requests coming to the server at a time, it is difficult to find out the appropriate resource in the Cloud for fulfilling all the requests. To analyze the requests and to allocate the resources, it is important to record the user requests, available resources, and request deployment details. Proposed MADRA acts as an interface between the Cloud environment and the Cloud users. The agents used for increasing the QoS regarding resource allocation are,Green Navigator Agent (GNA), Service Analyzer Agent (SAA), User Profile Agent (UPA), QoS Parameter Monitoring Agent (QPMA), Service Scheduling Agent (SSA) and Pricing Agent(PA).

The virtual machine manager maintains the availability of the virtual machines and the resources. It also takes care of the virtual migration across physical machines. Pricing Agent (PA) decides about the cost for incoming services based on the time, demand and supply needed for the computing resources with all facilities. PA assigns priority for the services and allocates cost for the resources efficiently. PA considers the number of request, size of the request, time taken for Completion and cost for resource allocation. The Figure 6.2 shows the model used in the proposed technique. According to the proposed approach, resources allocation is performed based on various attributes of requests and resources. For each request, its weight, cost, speed parameters are calculated independently. The average (mean) for every request is computed by means of the calculated structures. Similarly, energy and memory parameters are computed for each resource. This model uses the Request Queue which is used to track the time between the requests enters and exits the system. All the requests are time stamped as per the specification of the system. The time may include an actual queue that requests enter into the queue. The time can be used in many other functions such as load balancing or internal network latency.

Virtual Machines

VMs are used to handle the user requests. It provides more flexibility for partitioning the resources in a single machine into the different requirements for the service requests. Several virtual machines can be concurrently accessed by different OS based environments dynamically. This process saves energy, time and cost. In the proposed work, the entire processes are decomposed into several sub-processes and are assigned to several agents in the Cloud. Then, the resources are ranked in terms of time, energy, and availability. The chosen resource should be matched with user requirement. Finally, in terms of QoS, all the resources are evaluated by assigning a score. From which the best score is selected and allocated for the appropriate incoming request dynamically shown in Figure 2 and Figure 3.

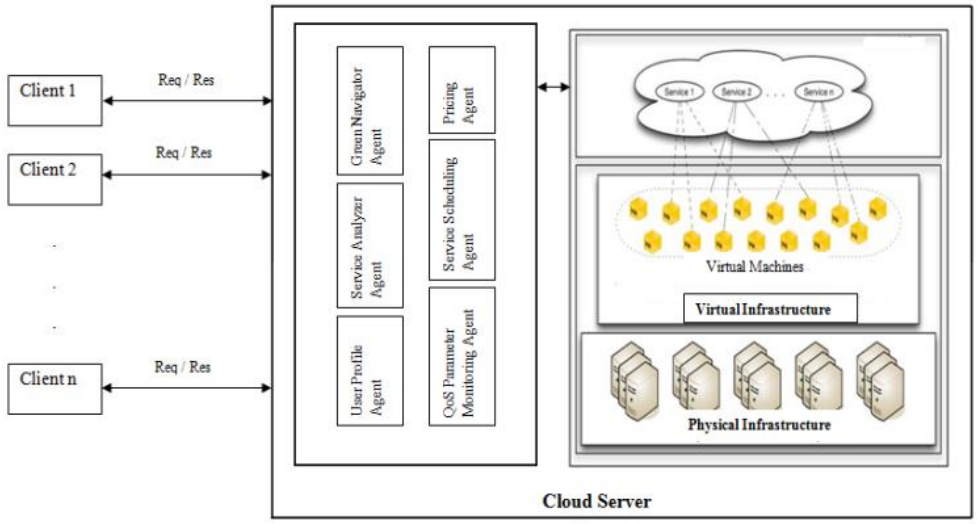

Figure 2: Proposed Multi-Agent based Model for Resource Allocation

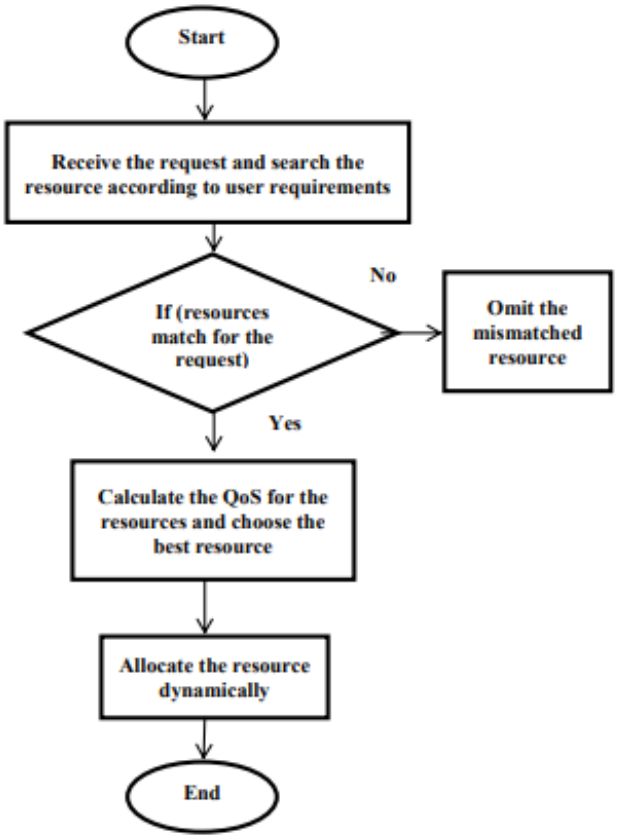

Figure 3: Steps Involved in MADRA

Here the considered QoS parameters are resource availability, resource allocation time and energy. Maximum_QS is the highest score which shows the best quality among various resources. The highest QS value is selected in term of high resource availability, less response time and energy for resource allocation. For all the QoS parameter, a threshold value is assigned to check the quality based on their values.

Proposed Multi source resource allocation

Cloud environment becomes complex with multiple heterogeneous resources. Monitoring, investigating and finding the appropriate resources become difficult due to scalability and complexity. Existing approaches provide solution in resource allocation, but the percentage of improvement in Quality of Service

(QoS) parameters can still be increased. This Article presents a multi-source framework for improving QoS during resource allocation in Cloud. Customer satisfaction is increased by enhancing the performance of the entire system. It can be obtained by server virtualization which is part of resource allocation. Cloud servers are distributed geographically and provide low latency, location awareness, and mobility. The proposed work focuses on increasing the efficiency of resource allocation algorithm and its applications in the Cloud environment. This work analyses the existing algorithms for resource allocation and then designs a framework for optimum resource allocation.

The proposed framework, shown in Figure 4, is designed and developed for finding solution to problems related to fault tolerance, overflow and underflow. The proposed framework is arranged into multiple sources monitoring and investigation of all requests, processors available, finding resources and allocating them. In the resource layer, all the resources are arranged in a typical manner. Each server manager comprises of data servers which verify the accessibility of the processor and has the responsibility of managing the virtual machines. In case if the client did not receive any response according to their request within the stipulated time interval, then, the client is assigned a wait status for the process, then, client requests are forwarded to next nearest middle layer i.e. server manager. The data server allocates the processor to client request and forwards available processor information to respective server manager. The proposed QRA algorithm is implemented and deployed in the middle of each source of the Cloud architecture. The aim of this algorithm is to utilize the resources effectively and reduce congestion by using intermediate multiple layers deployed in a Cloud environment.

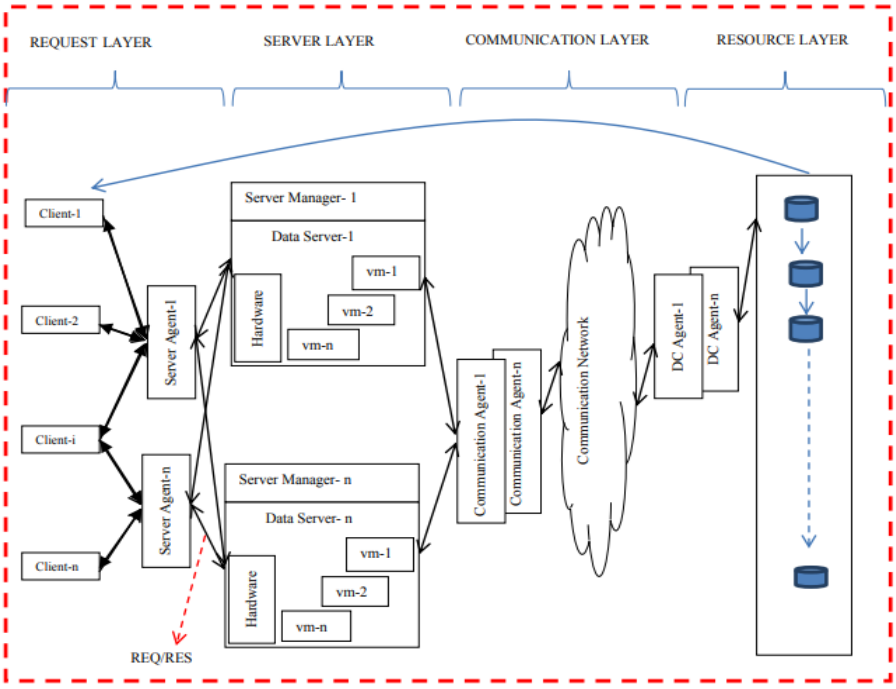

Figure 4: Multi-source Framework for Resource Allocation in Cloud

If the request processing time exceeds the assigned time, the request is forwarded to the next middle layer i.e. server manager. For exchanging and transmitting requests and responses, an energy threshold value is assigned to all the devices. If the present energy of the devices is greater than or equal to the energy threshold value then the REQRES is transmitted, or else other energy devices are elected for transmission. This method increases the energy efficiency in Cloud. QRA algorithm notices the power consumption and the resource scheduling to reduce energy, completion time and response time. Power consumption depends on the location and distance of the servers. If the location of the server is too far, then, the amount of power consumed by the servers is high. The scheduling is applied when one resource is

allocated to two requests and it is done according based on the priority of the requests. Scheduling purely depends on the workload of the data servers and complexity of the request and time taken to process the requests. Because of the dynamic nature of the resource allocation in data server, it should not take more time for processing the request.

Experimental results and discussion

Multi agent

The proposed MADRA, strategy, is simulated in CloudSim. The intention of the CloudSim tool is to provide a simulation framework which enables modeling, simulation and experimentation of emerging Cloud computing Infrastructure as a Service (IaaS), Software as a Service (SaaS), and Platform as a Service (PaaS) based applications and application services. It allows the users to focus on a specific process and resource can be investigated without any programming issues. The existing algorithm considered in this paper is, Improved Differential Evolution Algorithm (IDEA) [17]. Table 1 shows the simulation settings. In the simulation, a data center with 50 heterogeneous physical nodes having CPU with performance equivalent to 1000, 2000 or more than 2000 million instructions per second, 8-GB RAM and 1 TB of storage is created. When the user submits a request it is processed and the capacity of the resource is simulated under the data center. Different kinds of web applications are running on each VM with different workloads. This can be used as a model for creating the utilization of CPU according to the requirement randomly. Each experiment is simulated more than 10 times and the results are analyzed. Table-6.1 shows the parameters used in the simulation environment to simulate the proposed method. The XEN server is simulation software for Cloud simulation.

Table 6.1 Simulation Settings

Entity	**Tool/Software**
Operating System	Windows 8
Simulation Engine	CloudSim 3.0
Front-end IDE	NetBeans 8.0.3
Programming Language	Java
Performance Monitor	Cloud Analyst
Results Analysis	Cloud Report
Test-Bed	Xen-Server Based Cloud
VM	VM scheduler

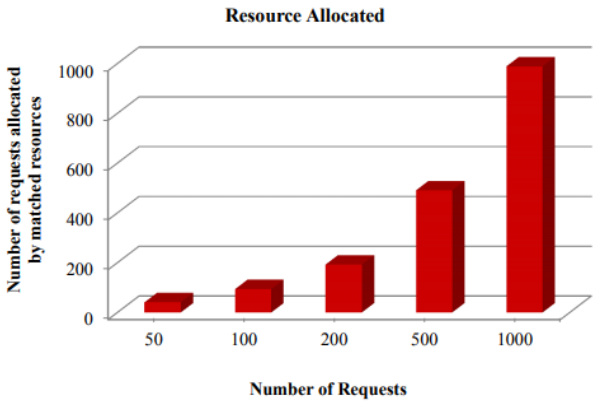

Figure 5 Numbers of Requests Allocated by Matched Resources

Figure 5 shows the number of best resources chosen for the user requests. Since the numbers of users requests are more at the same time, choosing the resources is also more in terms of quality score computed by the agents. The number of matching (selected) resources is not constant for all the iterations. It depends on the availability of the resources and quality score. MADRA utilizes the agents in various sources to reduce the request response time. Also, one of the agents is investigating the resources, which resource is matched accurately for the user request. The quality (quality-score) of the resource can be calculated by investigating the resource name, time is taken to allocate the resource, energy, resource availability etc. Using this quality score the appropriate resource can be chosen, which can provide better performance in the Cloud. By this way, MADRA is decided as a better approach for resource allocation effectively.

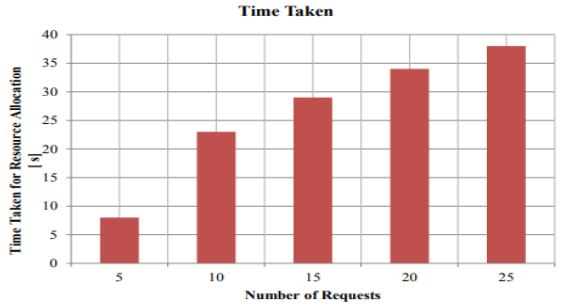

Figure 6: Number of Requests vs. Time

The proposed methodology MADRA allocates relevant, best resources to the user request even if more number of requests. Figure 6 illustrates the time taken for resource allocation. A user request is not unique and does not require the same resource by all the users. Thus, investigating and selecting a matched resource for one request in terms of QoS takes some time interval. From the obtained results, it is clear that when request increases the time taken for resource allocation also increases. The time taken to allocate the resources depends on the resource availability, distance between the resources and the client (request place). The proposed approach takes 8 s, 23 s, 29 s, 34 s, and 38 s for responding when the number of user requests of 5, 10, 15, 20, and 25 respectively. Hence MADRA is efficient in terms of response time.

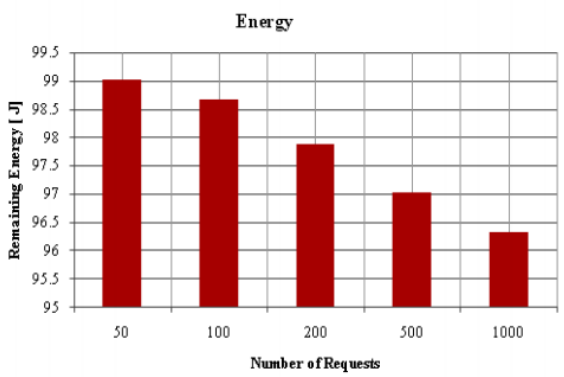

Figure 7: Number of Requests vs. Energy Consumption

Energy consumption increasing is directly proportional to increasing number of requests. Figure 7 shows the energy consumption in joules. To allocate the resources, Cloud entity needs some amount of energy. Figure 7illustrates the energy consumption for resource allocation obtained in the simulation based experiment. A resource needs a certain amount of energy for its each state like active, service, and live etc. When a resource is investigated, it should be active and involved in the job, so some amount of energy is spent on the resources. While the number of resources and time increases, the energy consumption also increases. By reducing the resource active state, processing speed of the energy can be saved.

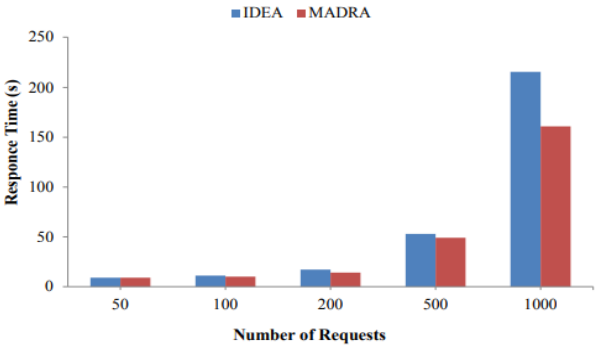

Figure 8: Comparison based on Response Time

It is also simulated for large sized problems such as 50, 100, 200, 500 and 1000 requests and the performance is evaluated and presented in figure 8. The energy savings are in a range from 96% to 99% when the user requests are between 50-1000. When a request increases the number of resources matched for the requests also increases. The resources availability can be verified dynamically while evaluating the Cloud user request by comparing the request based resources by the multiple agents. Since multiple agents are working simultaneously the time complexity is not affecting the Cloud computing based resource allocation. To evaluate the performance of the proposed approach it is compared with the existing algorithms.

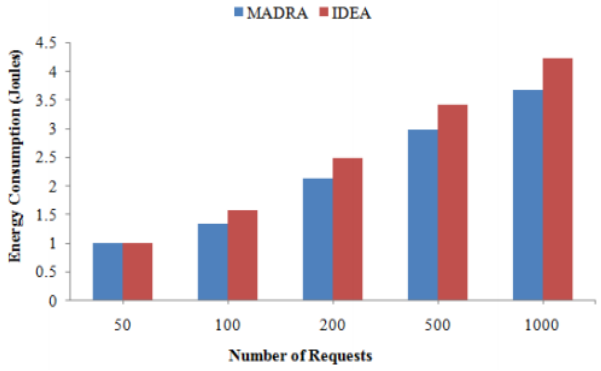

Figure 9: Comparison based on Energy Consumption

From the experimental simulation result figure 9, on the average, the proposed strategy, MADRA, improves the response time and energy consumption by 20.39% and 12.55%

respectively when compare to existing approach IDEA. The simulation results illustrate that the proposed approach is efficient and potential for the large data center in Cloud. Any CPU deployment in Cloud does not affect the migration with the Cloud server. The multi-agents smoothly verify and distribute resources to user requests.

Multi source

The proposed QoS-based Resource Allocation technique (QRA) is implemented and simulated in Green cloud Simulator tool and the results are discussed in this section. Results are measured and computed in terms of response time, resources allocated and energy consumption. The time taken for responding the requests is shown in Figure 10. When compared with existing ERA, on the average, the proposed approach QRA has shown 20.52% improvement in response time. When task increases, the time taken to process the task is also increased. The proposed methodology QRA requires 48 seconds to process 500 requests, whereas, ERA requires 60 seconds to process 500 requests. The incoming requests and responses are validated initially during resource allocation. The error is calculated to predict the request and response validation. Less percentage of error shows that the incoming requests are valid. The server manager processes the requests if and only if the incoming requests are valid. The error is predicted from the simulation and is tabulated in Table 2. In this work, the total number of requests is 500 for computing the error percentage.

Table 6.2 Estimation of Average Error using QRA and ERA

S. No	Number of Requests	Valid Requests in QRA	Error %	Shahin Vakilinia et al. 2016	Error %
1.	50	50	0	50	0
2.	100	100	0	99	1
3.	150	149	0.7	147	2
4.	200	198	1	195	2.5
5.	250	248	0.8	244	2.4
6.	300	297	1.0	292	2.7
7.	350	348	0.6	339	3.1
8.	400	397	0.8	387	3.3
9.	450	446	0.9	434	3.6
10.	500	495	1	480	4

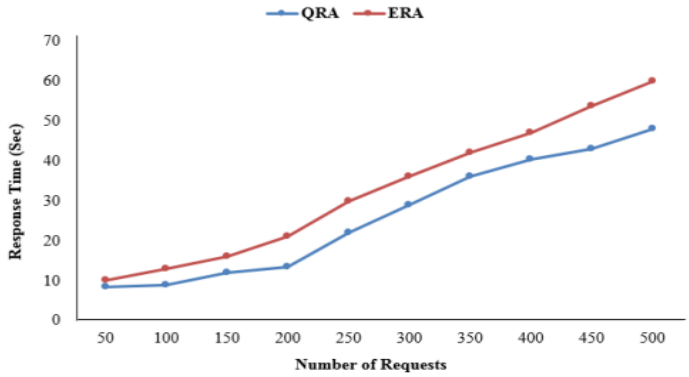

Figure 10: Comparison based on Response Time

After validating the requests, the power efficiency of the entire framework is calculated in order to transfer the data. The proposed QRA encounters 1% error among 200 requests; ERA encounters 2.5% of error for the same number of requests. Comparison based on power consumption using the proposed QRA and the existing ERA is shown in Figure 10. When compared with existing ERA, on the average, the proposed approach QRA has shown 13.29% power reduction.

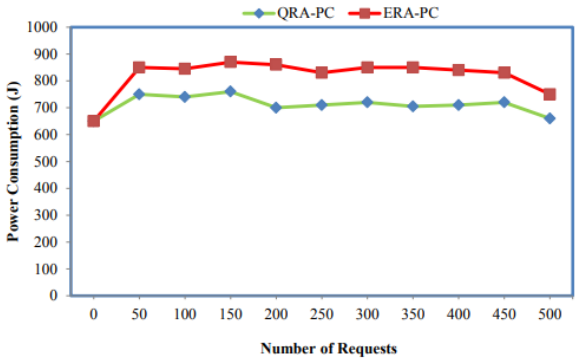

Figure 11: Comparison based on Power Consumption

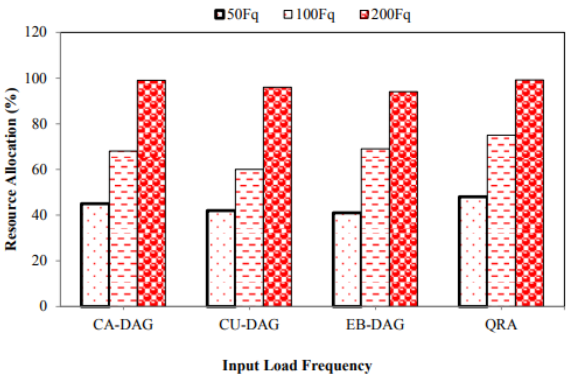

Figure 12: Number of Requests Allocated by Resources

The number of valid requests allocated by appropriate resources is shown in Figure 12. The resource allocation efficiency is shown using the load frequency. When load frequency increases, generally, the percentage of resource allocation also increases. For example, the load frequency considered in this simulation is 50Hz, 100Hz and 200Hz and the respective resource allocation obtained using the proposed QRA is 48%, 75% and 99.18% which is higher than the existing approaches such as Edges Based Directed Acyclic Graph (EB-DAG)[11], Communication Unaware-DAG (CU-DAG) [13]and Communication Aware DAG (CA-DAG) [15].From the experimental results, it can be observed that the QoS parameters such as time and energy of the proposed QRA are efficient when compared to ERA.

Conclusion

The aim objective of the Article is to allocate the resource speedily and dynamically. It is concerned that the QoS parameters should be improved during resource allocation. For this, MADRA is proposed using multiple agents involved in the resource investigation and allocation process on the server side. Each agent has its own assigned job and works simultaneously and individually, but the agents are correlated with one another. In simulation results, the performance evaluation of the proposed MADRA strategy is evaluated on various factors such as resource allocation, response time, and energy consumption. By using the proposed strategy, it is evident that the average of response time has seen a steep decline by 20.39% and improves the average of energy consumption by 12.55% when comparisons with existing approach IDEA. It is proved that MADRA is efficient for resource allocations dynamically even when the numbers of user requests are high and also improves QoS parameter in term of response time and energy consumption. The proposed multi-source framework, QRA, which is a QoS-based resource allocation to improve the efficiency of Cloud, is presented in this article. A Multi-source framework which employs QRA algorithm is designed and implemented on Green cloud Simulator tool. On the average, the proposed QRA approach has 20.52% improvement in response time, 13.29%, reduction in power consumption and error prediction of 1% when compared to existing ERA approach. From simulation results, it is evident that when the input load frequency is 200Hz, the percentage of resource allocation is 99.18% which is high when compared to that of other approaches such as EB-DAG, CU-DAG, and CA-DAG models. It is observed that the proposed Multi-Source Framework with QRA algorithm is efficient for resource allocation even with large number of requests and is scalable for more user requests and also improves QoS parameters such as completion time, response time and energy.

References

1. Krishnamoorthy, R, & SreedharKumar ,S,2012, ‘New optimized agglomerative clustering algorithm using multilevel threshold for finding optimum number of clusters on large data set’, IEEE International Conference on Emerging Trends in Science, Engineering and Technology, pp. 121-129.
2. Krishnaveni, N., & G. Sivakumar, 2013,‘Survey on dynamic resource allocation strategy in cloud computing environment’, International Journal of Computer Applications Technology and Research,vol. 2, no. 6, pp. 731-737.
3. Kumar, Deepesh, & Ajay Shanker Singh, 2015, ‘A survey on resource allocation techniques in cloud computing.’ Proceedings of 116 International Conference

on Computing, Communication & Automation (ICCCA), pp. 655-660.
4. Kumbhare, AlokGautam, YogeshSimmhan, Marc Frincu, & Viktor K. Prasanna,2015, 'Reactive resource provisioning heuristics for dynamic dataflows on cloud infrastructure.', IEEE Transactions on Cloud Computing, vol. 3, no. 2 ,pp. 105-118.
5. Laili, Yuanjun, Fei Tao, Lin Zhang, & Lei Ren,2011, 'The optimal allocation model of computing resources in cloud manufacturing system.' ,Proceedings of Seventh International Conference In Natural Computation (ICNC), vol. 4, pp. 2322-2326.
6. Lee, Young Choon, Chen Wang, Albert Y. Zomaya, & Bing Bing Zhou,2012,'Profit-driven scheduling for cloud services with data access awareness.' ,Journal of Parallel and Distributed Computing,vol. 72, no. 4 ,pp. 591-602.
7. Leontiou, Nikolaos, DimitriosDechouniotis, NikolaosAthanasopoulos, & Spyros Denazis,2014, 'On load balancing and resource allocation in cloud services.' ,Proceedings of Control and Automation (MED), pp. 773-778.
8. Li, Jian, Kai Shuang, Sen Su, Qingjia Huang, PengXu, Xiang Cheng, & Jie Wang,2012, 'Reducing operational costs through consolidation with resource prediction in the cloud.' , Proceedings of the 2012 12th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing , pp. 793-798.
9. Li, Yong, Jizhong Han, & Wei Zhou,2014, 'Cress: Dynamic Scheduling for Resource Constrained Jobs.' Proceedings of 17th International Conference on Computational Science and Engineering , pp. 1945-1952.
10. Liu, Hongzhe, Hong BAO, Jun WANG, & De XU, 2010, 'A novel vector space model for tree based concept similarity measurement', IEEE International Conference on Information Management and Engineering.
11. Liu, Kaiyang, Jun Peng, Weirong Liu, Pingping Yao, & Zhiwu Huang,2014, 'Dynamic resource reservation via broker federation in cloud service: A fine-grained heuristic-based approach.', In Global Communications Conference (GLOBECOM), pp. 2338-2343. 117
12. Liu, Xi, Xiaolu Zhang, Xuejie Zhang, & Weidong Li,2015, 'Dynamic fair division of multiple resources with satiable agents in cloud computing systems.',Proceedings of Big Data and Cloud Computing (BDCloud), pp. 131-136.
13. Logeswaran, Lajanugen, HMN DilumBandara, & Bhathiya,2016,'Performance, Resource, and Cost Aware Resource Provisioning in the Cloud.' Proceedings of 9th International Conference on Cloud Computing (CLOUD), pp. 913-916.
14. Ma, & Richard, TB,2016, 'Efficient Resource Allocation and Consolidation with Selfish Agents: An Adaptive Auction Approach.',Proceedings of 36th International Conference on Distributed Computing Systems, pp. 497-508.
15. Magedanz, Thomas, & Florian Schreiner,2014, 'QoS-aware multicloud brokering for NGN services: Tangible benefits of elastic resource allocation mechanisms.', Fifth International Conference on Communications and Electronics , pp. 168-173.
16. Malatras, Apostolos,2015, 'State-of-the-art survey on P2P overlay networks in pervasive computing environments.' Journal of Network and Computer Applications, vol. 55, pp. 1-23.
17. Manjur Kolhar, Saied M. Abd El-atty & Mohammed Rahmath,2017, 'Storage allocation scheme for virtual instances of cloud computing', Neural Computing and Applications, vol. 28, no. 6, pp. 1397-1404.
18. Mathapati, Basavaraj S., Siddarama R. Patil, & Mytri,VD,2012,'Energy Efficient Reliable Data Aggregation Technique for Wireless Sensor Networks', IEEE International Conference on Computing Sciences, pp. 155-158.
19. Meenakshi, A., H. Sirmathi, & J. Anitha Ruth,2016, 'Quality Assured Optimal

Resource Provisioning and Scheduling Technique Based on Improved Hierarchical Agglomerative Clustering Algorithm (IHAC)', International Journal of Engineering and Technology (IJET), vol. 8, no. 4.

20. Michael Cardosa & Abhishek Chandra,2009, 'HiDRA: Statistical Multi-dimensional Resource Discovery for Large-scale Systems', International Workshop on Quality of Service, Quality of Service, pp. 1-9. 118