

## AN IMPROVED RESOURCE ALLOCATION STRATEGY BASED ON PACKAGE LEVEL ACCESS IN CLOUD ENVIRONMENT

Rajveer Kaur <sup>(1)</sup>, Mr. Navtej Singh Ghumman <sup>(2)</sup>

<sup>(1)</sup> Research Scholar, Department of Computer Science & Engineering, SBSSTC, Ferozpur, Punjab.

<sup>(2)</sup> Assistant Professor, Department of Computer Science & Engineering, SBSSTC, Ferozpur, Punjab.

### ABSTRACT

Cloud computing is an emerging field of computer science, where heterogeneous services such as applications, servers and storage are delivered to an individual or organization's computer and devices through the Internet. In the proposed research work, a load balancing algorithm is designed, to provide proper utilization of all resources while processing the requests received from the users. We have implemented the 'package' level access at the Virtual Machines. We have proposed three types of packages like basic, medium and premium according to the requirements of the Cloudlets. In the basic package, we have the Virtual Machines with low capacity and in the premium package we are having the Virtual Machines with higher capacity, whereas the medium is of intermediate capacity of MIPS and RAM. In the proposed approach, the capacity, status and current load of every Virtual Machine is computed before allocating the new request of the user. Different parameters like waiting time, execution time and turnaround time of the Cloudlets are computed and analyzed. The proposed research provides the anticipated results with the implementation of the proposed algorithm. Compared with the other job scheduling algorithms, the proposed load balancing algorithm can outperform them in circumstances where the load and Virtual Machines are heterogeneous.

### Keywords

Cloud Computing, load balancing, virtual machine, data center broker, cloudlet, cloud sim

### INTRODUCTION

Cloud computing is an on demand service in which shared resources, information, software and other devices are provided according to the clients requirement at specific time. It's a term which is generally used in case of Internet. The whole Internet can be viewed as a cloud. Capital and operational costs can be cut using cloud computing [36]. With traditional desktop computing, we run copies of software programs on our personal computer. The documents we make are stored on our own pc. Although documents can be accessed from other computers on the network, they cannot be accessed by computers outside the network. This is PC-centric. By cloud computing, the software programs one use are not run from one's personal computer, but are quite stored on servers accessed via the Internet. If a computer crashes, the software is still available for others to use. Similar goes for the documents one create; they are stored on a collection of servers accessed through the Internet. Anyone with permission can not only access the documents, but can also edit and cooperate on those documents in real time. Unlike traditional computing, this cloud computing model is not PC-centric, it's document-centric. Cloud Computing may be a term accustomed define each a platform and kind of application. As a platform it provides, constructs and reconfigures servers, while the servers is physical or virtual machines. Cloud could be a term used as a trope for the wide space networks (like internet) or several such giant networked atmosphere. It came partially from the cloud like image wont to represent the complexities of the networks within the schematic diagrams. It represents all the complexities of the network which can hold everything from cables, routers, servers, knowledge centers and every one such alternate devices.

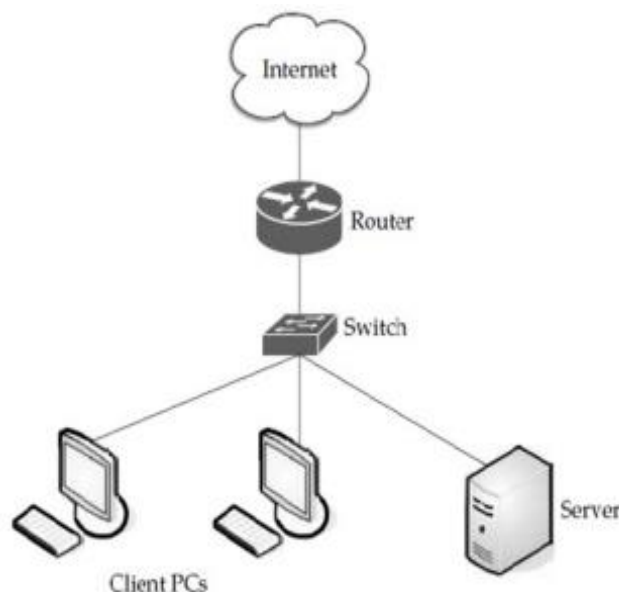


Figure 1. Model of Cloud Computing [8]

## LOAD BALANCING

It is a process of reassigning the total load to the individual nodes of the collective system to make resource utilization effective and to improve the response time of the job, simultaneously removing a condition in which some of the nodes are over loaded while some others are under loaded. It is used to achieve a high user satisfaction and resource utilization ratio, hence improving the overall performance of the system. Proper load balancing can help in utilizing the available resources optimally, thereby minimizing the resource consumption. It also helps in implementing fail-over, enabling scalability, avoiding bottlenecks and over-provisioning, reducing response time etc. Load Balancing is a method to distribute workload across one or more servers, network interfaces, hard drives, or other computing resources [13]. Typical datacenter implementations rely on large, powerful (and expensive) computing hardware and network infrastructure, which are subject to the usual risks associated with any physical device, including hardware failure, power and/or network interruptions, and resource limitations in times of high demand. Load balancing in the cloud differs from classical thinking on load-balancing architecture and implementation by using commodity servers to perform the load balancing. This provides for new opportunities and economies-of-scale, as well as presenting its own unique set of challenges. Load balancing is used to make sure that none of your existing resources are idle while others are being utilized. Modern high-traffic websites must serve hundreds of thousands, if not millions, of concurrent requests from users or clients and return the correct text, images, video, or application data, all in a fast and reliable manner. To cost-effectively scale to meet these high volumes, modern computing best practice generally requires adding more servers [28]. A load balancer acts as the “traffic cop” sitting in front of your servers and routing client requests across all servers capable of fulfilling those requests in a manner that maximizes speed and capacity utilization and ensures that no one server is overworked, which could degrade performance. If a single server goes down, the load balancer redirects traffic to the remaining online servers. When a new server is added to the server group, the load balancer automatically starts to send requests to it.

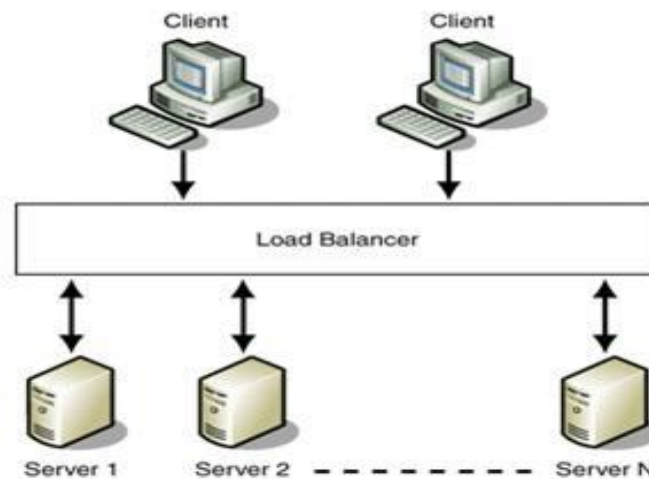


Figure 2. Model of Load Balancing [28]

## LOAD BALANCING CHALLENGES

Although cloud computing has been widely adopted. Research in cloud computing is still in its early stages, and some scientific challenges remain unsolved by the scientific community, particularly load balancing challenges [1].

- i. **Providing services automatically:** A main feature of cloud computing is elasticity; resources can be allocated or released automatically as per the end user's requirement. Here challenge is to how to allocate or release the resources by keeping the system performance same as traditional one.
- ii. **Virtual Machines Migration:** Virtualization is a mean to create a virtual or replicated form of a device or resource that can be a server, storage device, any network or an operating system to unload a physical machine that is currently heavily loaded. Here main challenge is to dynamically distribute the load from that overload physical machine to some other side by moving virtual machine to avoid bottlenecks in Cloud computing system.
- iii. **Stored Data Management:** Cloud computing includes large sum of data. Main challenge is to distribute data across the cloud that is stored optimally while maintaining fast access for users.
- iv. **Usage of small data centers:** Small datacenters can be more beneficial, cheaper and less energy consumer than large datacenter. Load balancing will become a problem then to ensure a maximum response time with an optimal distribution of resources.
- v. **Spatial Distribution of the Cloud Nodes:** Some algorithms are designed to be efficient only for closely located nodes where communication delays are comparatively small. Here challenge is to design a load balancing algorithm that can work for spatially distributed nodes. For such nodes there would be some other factors too to be noticed such as the speed of the network links among the nodes, the distance between the user and the processing nodes, and the distances between the nodes involved in providing the cloud services.

- vi. **Point of Failure:** Load balancing mechanism must be designed in a way to overcome from situations such as single point failure. For such situations there are some mechanisms that involve certain pattern that uses a controller for whole system. In such situation if that controller fails, it can cause failure to the whole system. It is also a challenge while making a mechanism for proper load balancing in clouds.
- vii. **Algorithm Complexity:** Load balancing algorithms are preferred to be less complex and must be easy to understand in terms of implementation and operations. High complexity will cause negative performance issues.

## RELATED WORK

In this section literature survey regarding the previous work and related techniques about the load balancing techniques of cloud computing are presented. We have studied and analyze many load balancing techniques for cloud computing. After our detailed literature review we are able to discover theme of our work including the techniques we have incorporated in various phases, to implement it and to evaluate our results and conclusion. Reena et al. (2015) discussed an efficient Dynamic Load Management Algorithm where load is managed by the server by considering the present status of present VMs for request assignment sharply. That distribution of the entire incoming request among the virtual machines effectively. [34] Hiren et al. (2015) presented a Flexible load sharing algorithm (FLS) which introduce the third function. The third function makes partition the system in to domain. This function is helpful for the selection of other nodes which are present in the same domain. The third partition function will make the partition in to each domain so that using the remote location the information will be transferred to another domain. Such that the node will exchange the information from one node to another node. By applying the flexible load sharing to the particular domains in to the distribute system, the performance can be improved when any node is in overloaded situation. [20] B. Kruekaew et al. (2014) applies artificial Bee Colony (ABC) to optimize the scheduling of Virtual Machine (VM) on Cloud computing. The main contribution of work is to analyze the difference of VM load balancing algorithm and to reduce the make span of data processing time. There are three kinds of honey bee to search food sources, which include scout bees search for food source randomly, employed bees search around the food source and share food information to the onlooker bees, and onlooker bees calculate the fitness and select the best food source. [10] Nagamani H et al. (2014) proposed a new technique to achieve load balancing called Load Balancing with Optimal Cost Scheduling Algorithm. The user selects a list of services available from the service pack. The scheduler processes these tasks to the virtual machine (VM) based on the configuration and computing power of the VM. This task is achieved with minimum execution cost which is a profit for the service provider and minimum execution time which is an advantage for both service provider and the user. [30] M. Ajit et al. (2013) described Cloud load balancing helps to enhance the overall cloud performance. This paper presents the analysis of three contemporary algorithms in cloud analyst tool to resolve the issue of cloud load balancing as a preparation phase for new load balancing technique.. [29] Desai et al. (2013) discusses about the emerging technology i.e. a new standard of large scale distributed computing and parallel computing. It provides shared resources, information or other resources as per clients' requirements at specific times. For better management of available good load balancing techniques are required. And through better load balancing in cloud, performance is increased and user gets better services. So in this author has discussed many different load balancing techniques used to solve the issue in cloud computing environment. [39] Babu et al. (2013) proposed a Honey Bee Behavior inspired Load Balancing [HBB-LB] technique which helps to achieve even load balancing across virtual machine to maximize throughput. It considers the priority of task waiting in queue for execution in virtual machines. After that work load on VM calculated decides whether the system is overloaded, under loaded or balanced. And based on this VMs are grouped. [25] Al-Rayis et al. (2013) explains that basically, load balancers can be deployed based on three different architectures. The centralized load balancing architecture which includes a central load balancer to make the decision for the entire system regarding which cloud resource should take what workload and based on which algorithm(s). Upendra et al. (2013) discussed that in enhanced Max-Min Task Scheduling Algorithm in cloud computing helps in supplying a high performance computing based on protocols which allowed shared computation and storage over long distances. It depends upon expected execution time instead of completion time [41]

## RESEARCH GAP

Cloud computing thus involving distributed technologies to satisfy a variety of applications and user needs. Sharing resources, software, information via internet are the main functions of cloud computing with an objective to reduced capital and operational cost, better performance in terms of response time and data processing time, maintain the system stability and to accommodate future modification in the system .So there are various technical challenges that needs to be addressed like Virtual machine migration, server consolidation, fault tolerance, high availability and scalability but central issue is the load balancing , it is the mechanism of distributing the load among various nodes of a distributed system to improve both resource utilization and job response time while also avoiding a situation where some of the nodes are heavily loaded while other nodes are idle or doing very little work. It also ensures that all the processor in the system or every node in the network does approximately the equal amount of work at any instant of time.

## PROBLEM FORMULATION

- The current algorithm will work only in the homogeneous cloud system where all the resources are of same configuration.
- Cloudlets are not assigned to the virtual machine according to their capacities. There may be a scenario where a cloudlet with highest priority is assigned to the machine with lowest capacity in the host.
- The processing capacity (No of processors / MIPS) is not considered for assigning the VM to a job.



- The status or the current load of the VM is not computed before allocating the new request to it.

## OBJECTIVES

The primary purpose of the cloud system is that its client can utilize the resources to have economic benefits. A resource allocation management process is required to avoid underutilization or overutilization of the resources which may affect the services of the cloud.

- To implement and study the performance of existing load balancing algorithms.
- To design the improved load balancing algorithm with heterogeneous Virtual Machines with package level access in datacenter with different hosts.
- To improve the different parameters by compute the capacity, status and current load of the virtual machine before allocating the new request of the user.
- To develop the proposed algorithm and compare the performance of proposed algorithm with existing algorithm.

## RESEARCH METHODOLOGY

Cloud services provide computing on demand in real time. Number of users accessing cloud environment are always more than that were using it on previous day. Cloud has application areas for developing applications, providing and managing infrastructure, patching applications. Users and their requests for accessing cloud infrastructure are highly dynamic and loading servers running in data center. We need efficient strategy to balance load on these servers so that the servers don't get crash and they can persist long. Precisely Objective is to achieve accuracy, performance of servers and the cloud environment can be maintained. The lease types associated with the jobs are:

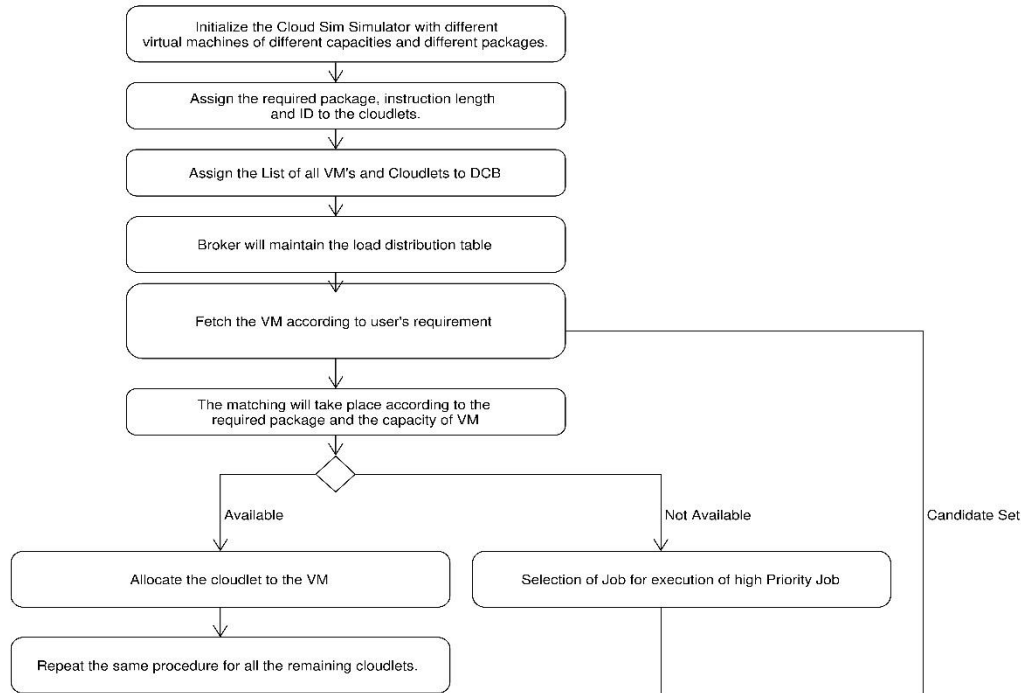
- Cancellable: These requests can be scheduled at any time after their arrival time. It need not be resumed later. Cancellable leases do not guarantee the deadline.
- Suspendable: Leases of this type can be suspended at any time but should be resumed later. This type of lease guarantees the execution but not in a specific deadline. Suspendable leases are flexible in start time and can be scheduled at any time after their ready time. In the case of preemption, these leases should be rescheduled to find another free time-slot for the remainder of their execution.
- Non-Preemptable: The leases associated with such requests cannot be preempted at all.

The proposed algorithm is implemented through the simulation software like cloud sim used. For the implementation of the application, java language is used. This research work considers Datacenter, VM, host and Cloudlet components from CloudSim for implementation of a proposed algorithm.

- Initialize the Cloud Sim in Java by creating different data centers having multiple number of hosts of different configurations.
- Each Host in the datacenter will have the different numbers of Virtual machines of different capacities.
- Then we will create the Cloudlets of varying length and size of different users.
- The virtual machines are having service level packages with different resources inside it.
- Create the multiple number of user cloudlets of different requirements and size.
- The Datacenter broker inside the cloud simulator will retrieve the number of cloudlets and number of virtual machines created by the user.
- The broker will maintain the LOAD distribution table in which the current status, current package, VM Capacity and the number of cloudlets submitted to each and every Virtual machine will be stored.
- The status will specify whether the virtual machine is busy or idle. The allocation entry in the table will define the total number of tasks submitted to this particular virtual machine.
- Submit the job of the user to the datacenter broker.
- DCB will first find the suitable VM according to the requirements of the cloudlet and will match and find the most suitable VM according to its availability or the machine with least load in the LOAD distribution table.
- Fetch that virtual machine which is having more power than another virtual machines and is under loaded.
- If the multiple virtual machines are satisfying the requirement, then the machine which is available at the first index will be used for executing the given task.
- Change the status of that virtual machine from idle to busy.
- Dispatch our cloudlet to that virtual machine and we will modify the rating of that particular virtual machine.
- Repeat the same procedure for all the remaining cloudlets. .

## SIMULATION RESULTS

Our proposed work, will start its process by arranging multiple resources inside the virtual machines. The virtual machines are having service level packages with different resources inside it. The broker will maintain the load distribution table in which the current status, current package, VM Capacity and the number of cloudlets submitted to each and every Virtual machine will be stored. The status will specify whether the virtual machine is busy or idle. The allocation entry in the table will define the total number of tasks submitted to this particular virtual machine. DCB will first find the suitable VM according to the requirements of the cloudlet and will match and find the most suitable VM according to its availability or the machine with least load in the load distribution table. Once each of the virtual machines has finished their current execution, each of them will update their status and ready to be re-assigned again. To conclude the results parameters like total processing time, total processing cost and total waiting time has been evaluated for both the existing work and proposed work.



**Figure 3. Flow Chart of Proposed Methodology**

In the following tables, we have taken multiple cloudlets ranging from 5 to 100000 for both of the base work and proposed work and are compared with each other. To compute the results of proposed work we are taking virtual machines with MIPS = 100, RAM = 256 MB and Bandwidth = 300 MB/second to calculate the parameters. In table 1, different types of cloudlets have been shown that are taken for evaluation of results in the existing and present work.

**Table 1. Different types of Cloudlets**

Sno.	Total No of Cloudlets	Cancellable Cloudlets	Suspendable Cloudlets	Non Pretable Cloudlets
1	5	2	2	1
2	10	3	4	3
3	40	15	15	10
4	60	20	20	20
5	100	30	35	35
6	150	50	50	50
7	200	70	60	70
8	300	100	100	100
9	400	150	100	150



10	500	180	140	180
11	700	250	250	200
12	1000	300	400	300
13	2000	600	700	700
14	3000	1000	1000	1000
15	5000	1400	2000	1600
16	7000	2000	2500	2500
17	8000	2000	3500	2500
18	9000	3000	2500	3500
19	10000	3000	3000	4000
20	20000	6000	7000	7000
21	30000	10000	10000	10000
22	40000	15000	10000	15000
23	50000	15000	17000	18000
24	60000	20000	20000	20000
25	70000	25000	25000	20000

Multiple number of experiments have been conducted inside cloud sim simulator using different number of cloudlets. The results of the proposed work have been mentioned in table 2. Same set of virtual machines have been used for the present work and the proposed work.

**Table 2. Results of the present work**

S. No.	Total Processing Time in seconds	Total Waiting Time in seconds	Total Processing Cost in dollars.
1	0	0	0
2	0	0	1
3	0	1	4
4	0	2	5
5	1	5	9
6	1	10	13
7	1	19	18
8	2	43	26
9	2	77	35
10	3	120	44
11	4	238	62
12	6	486	88
13	12	1950	176
14	18	4395	264
15	29	12219	441
16	41	23967	617
17	47	31301	705

18	53	39624	793
19	59	48927	881
20	117	195751	1762
21	176	440514	2644
22	235	783202	3525
23	294	1223746	4406
24	352	1762272	5287
25	411	2398722	6169

### TOTAL PROCESSING TIME

It is defined as the time interval between the request sent and the response received by the cloud user/ consumer. Overall processing time is calculated as given as follow:-

$$\text{Response Time (RT)} = \text{FT} - \text{ST}$$

Where, FT=finish time of execution

ST=start time of execution

The data, thus collected was used to plot a graph consisting of number of jobs of the existing work and the proposed work.

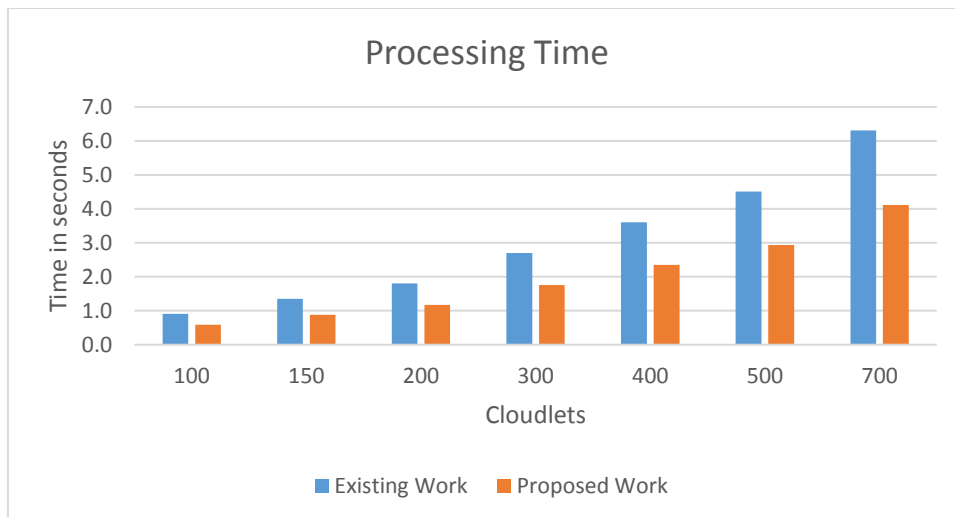


Figure 4. Total Processing Time

For the above bar chart, it is clear that total processing time has been reduced

### TOTAL PROCESSING COST

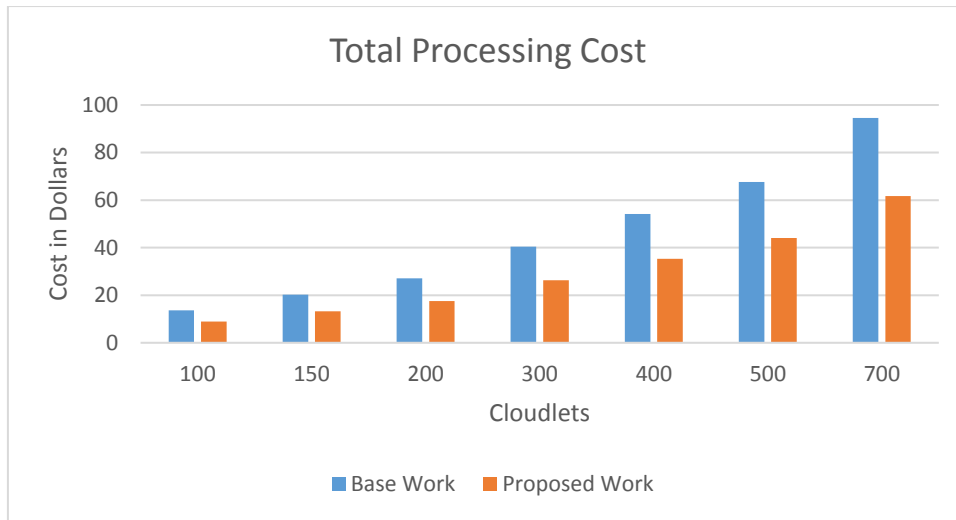
It is obtained by addition of cost per storage, cost per memory, cost per processor \* Processing time.

Processing cost=RT\* unit cost.

Where,

RT= response time

Unit cost = cost per unit time



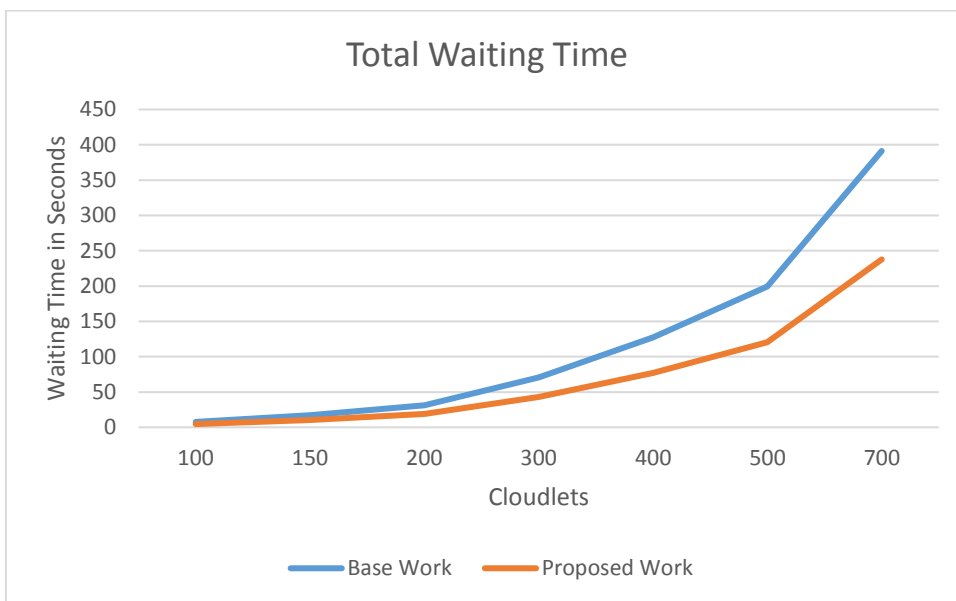
**Figure 5. Total Processing cost**

For the above bar chart, it is clear that total processing cost has been reduced.

## TOTAL WAITING TIME

Waiting time=Allocation Time-generation Time

Figure 6 illustrates that waiting time for the base work and proposed work



**Figure 6. Total waiting Time**

For the above bar chart, it is clear that total waiting time has been reduced.

## CONCLUSION

The performance of improved load balancing algorithm has been studied in the research work. The request time for the policies applied are same which means there is no effect on data centers request time after changing the algorithms. In order to evaluate performance of the proposed model simulation study has been put through various test conditions. It has been found that the model works well in ensuring an even distribution of the workload. In the current work, it has been assumed that all the incoming requests are independent to each other. The experiment conducted is compared with previous algorithm. The result indicates that the approaches outperform to previous algorithm in terms of total processing time, total processing cost and waiting time. To obtain a better solution, the model should be made more realistic by considering issues related to load balancing such as data locality, communication cost and flow time and results can be tested in real Cloud environment. However, within a computational cloud environment high throughput is of great interest rather than the load balancing. To achieve this, we have proposed the new dynamic load balancing algorithm. The experimental results obtained by applying the new proposed algorithm in the Cloud Sim Simulator, shows that the new work has outperformed the existing scheduling algorithms in large scale distributed systems.





## REFERENCES

- [1] A. Khiyaita, "Load Balancing Cloud Computing: State of Art", 9778- 1-4673-1053-6, 2012 IEEE.
- [2] Aameek, MadhukarKorupolu, DushmantaMohapatra, "Server-Storage Virtualization: Integration and Load Balancing in Data Centers", International Conference for high Performance Computing, Networking, Storage and Analysis, pp.1-12, ISBN 978-1-4244-2834-2, Nov 2008
- [3] Abhay , Sanjay Chaudhary, "Performance evaluation of web servers using central load balancing policy over virtual machines on cloud", COMPUTE '10, Proceedings of the Third Annual ACM Bangalore Conference, Article no. 16, ISBN 978-1-4503-0001-8, 2010.
- [4] Adeeb P.A. "A Seminar Report on security in cloud computing".
- [5] Al-Rayis, Ektemal, Heba Kurdi, "Performance Analysis of load balancing architectures in cloud computing", IEEE, European Modeling Symposium, pp. 520-524, ISBN 978-1-4799-2577-3, November 2013.
- [6] Ali M. Alakeel, A Guide to Dynamic Load Balancing in Distributed Computer Systems, IJCSNS International Journal of Computer Science and Network Security, VOL.10 No.6, June 2010.
- [7] Amandeep, SupriyaKinger"Analysis of Load Balancing Techniques in Cloud Computing", International Journal of Computers & Technology, Volume 4 No. 2, March-April, 2013, ISSN 2277-3061.
- [8] Anthony T.Velte, TobyJ.Velte, Robert Elsenpeter, "Cloud Computing A Practical Approach", TATA McGRAW-HILL Edition 2010.
- [9] Bendiab A.Randles M., Lamb D., "A comparative study into distributed load balancing algorithms for cloud computing", IEEE, 24th International Conference on Advanced Information Networking and Applications Workshops (WAINA), pp. 551-556, April 2010.
- [10] B. Kruekaew and W. Kimpan, "Virtual Machine Scheduling Management on Cloud Computing Using Artificial Bee Colony", Proceedings of the International Multi Conference of Engineers and Computer Scientists, ISBN: 978-988-19252-5-1 ,vol. 1 2014.
- [11] Calum Benson, Matthias, Müller-Prove Professional Usability in Open Source Projects: GNOME, OpenOffice.org, NetBeans.
- [12] C.C Rao, Mogasala, "Cloud: Computing Services and Deployment Models", International Journal of Engineering and Computer Science ISSN: 2319-7242 Volume 2 Issue 12, Dec.2013.
- [13] David, Andrew J. Korty, "Cloud Services: Policy and Assessment", EDUCAUSE Review, vol. 46, no. 4 (July/August 2011).
- [14] Dharmesh, JaydeepViradiya, "A Survey of Various Load Balancing Algorithms In Cloud Computing", International Journal of Scientific & Technology Research, Vol. 3, Issue 11, November 2014.
- [15] Dialogic (white paper), Introduction to cloud computing, 2010.
- [16] Fahringer T, Nae V., Prodan R., " Efficient management of data center resources for massively multiplayer online games" ACM/IEEE conference on Supercomputing, Article no. 10, ISBN: 978-1-4244-2835-9, 2008.
- [17] Fang Y., Wang F. and Ge J., "A Task Scheduling Algorithm Based on Load Balancing in Cloud Computing", Springer Berlin Heidelberg, Proceedings of International Conference, pp. 271-277, ISBN 978-3-642-16514-6, October 2010.
- [18] Haozheng, YihuaLan, and Chao Yin. "The load balancing algorithm in cloud computing environment." Computer Science and Network Technology (ICCSNT), 2012 2nd International Conference on. IEEE, 2012.
- [19] H. Deldari and M. Salehi , "Grid Load Balancing using an Echo System of Intelligent Ants", Proceedings of the 24th IASTED International Conference on Parallel and Distributed Computing and Networks, pp. 47-52, 2006.
- [20] Hiren, A. Bheda, "Enhance Load Balancing using Flexible Load Sharing in Cloud Computing", 2015 1st International Conference on Next Generation Computing Technologies (NGCT-2015) 4-5 September 2015.
- [21] Jianhua Gu, Guofei Sun, Tianhai Zhao, "A Scheduling Strategy on Load Balancing of Virtual Machine Resources in Cloud Computing Environment", IEEE, Third International Symposium on Parallel Architectures, Algorithms and Programming (PAAP), pp. 89-96, ISBN 978-1-4244-9482-8, December 2010.
- [22] K.Främling Introduction to Java Programming April 3rd, 2003.
- [23] Khiyaita.A, M. Zbakh, H. E Bakkali and DafirKettani, "Load Balancing Cloud Computing: State of Art", 9778- 1-4673-1053-6/12/\$31.00, 2012 IEEE.
- [24] Kumar Nishant, Pratik Sharma, Vishal Krishna," Load Balancing of Nodes in Cloud Using Ant Colony Optimization", 14th International Conference on Modelling and Simulation,2012.
- [25] L.D. Dhinesh Babu, P. Venkata Krishna "Honey bee behavior inspired load balancing of tasks in cloud computing environments", Elsevier, Applied Soft Computing Journal, ISSN: 1568-4946, 2013.



- [26] M.Dorigo, and T. Stutzle, "Ant Colony Optimization-Artificial Ants as a Computational Intelligence Technique", IEEE, Computational Intelligence Magazine, Vol. 1, Issue 4, pp. 28-39, November 2006.
- [27] Mayanka, Atul Mishra, "A Comparative Study of Load Balancing Algorithms in Cloud Computing Environment" ISSN 2319-7080 International Journal of Computer Science and Communication Engineering Volume 3 issue 1 (February 2014 issue).
- [28] Miss. Nikita, Prof. ManjushaDeshmukh, "Dynamic Load Balancing Algorithms For Cloud Computing", International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169 Volume: 3.
- [29] Mr. M. Ajit, Ms. G. Vidya, VM Level Load Balancing in Cloud Environment, Computing, Communications and Networking Technologies (ICCCNT), 2013 Fourth International Conference, July 2013.
- [30] Mrs. Nagamani H. Shahapure, Dr. JayarekhaP, "Load balancing with optimal cost scheduling algorithm", ICCPEIC, IEEE, 2014.
- [31] O.M. Elzeki, M. Z. Reshad, M. A. Elsoud "Improved Max-Min Algorithm in Cloud Computing", International Journal of Computer Applications, Vol. 50, No. 12, ISSN 0975-8887, July 2012.
- [32] Perez, R., van Doom, L., Sailer, R. IBM T.J. Watson Res. Center, Yorktown Heights, NY: Visualization and Hardware-Based Security -October 2008.
- [33] Peter Mell Timothy Grance, "National Institute of Standards and Technology" Special Publication 800-145 7 pages (September 2011).
- [34] Reena, Prof. Dr. BhawnaMallick, "Load Balancing in Cloud Computing Using Dynamic Load Management Algorithm", IEEE 2015.
- [35] Richhariya V, Dubey R, Siddiqui R., "Hybrid Approach for Load Balancing in Cloud Computing", Orient.J. Comp. Sci. and Techno, ISSN: 0974-6471.
- [36] RP Padhy, Rao "load balancing in cloud computing systems", may 2011.
- [37] Ruay, Chang, Jih-Sheng Chang, Shin-Yi Lin, "An ant algorithm for balanced job scheduling in grids"Elsevier. Future Generation Computer Systems,pp.20-27,June 2009.
- [38] Shilpa , Tanuj Wala "Study of Cloud Simulation Tools based on Cloud Sim" International Journal of Computer Science and Mobile Computing, Vol.4 Issue.5, May- 2015.
- [39] Tushar, JigneshPrajapati, "A survey of various load balancing techniques and challenges in cloud computing", International Journals of scientific and technology research, Vol. 2, Issue11, ISSN 2277-8616, November 2013.
- [40] Uhlig, R., Neiger, G. Rodgers, D. S.M. Kagi, A.Leung, F.H. Smith: Intel Corp., USA: Intel visualization technology IEEE Computer Society: May 2005.
- [41] Upendra, Purvi N. Ramanuj, "Enhanced max-min Task scheduling Algorithm in cloud computing", International Journal of Application or Innovation in Engineering & Management (IJAIEM), Vol. 2, Issue 4, ISSN 2319 - 4847, April 2013.
- [42] Vikas, Shiva, "A Load Balancing Based Cloud Computing Techniques and Challenges", International Journal of scientific research and management (IJSRM) Volume2 Issue5 2014.
- [43] Yaohang Li "A Bio-inspired Adaptive Job Scheduling Mechanism on a Computational Grid" International Journal of Computer Science and Network Security, VOL.6 No.3B, March 2006.
- [44] Yang Xu, Lei Wu, LiyingGuo, ZhengChen, Lai Yang, "An Intelligent Load Balancing Algorithm Towards Efficient Cloud Computing", in Proc. of AI for Data Center Management and Cloud Computing: Papers, from the 2011 AAAI Workshop.
- [45] Yi Lu, QiaominXie, Gabriel Kliot, Alan Geller, James R. Larus, Albert Greenberg, "Join-Idle-Queue: A Novel Load Balancing Algorithm for Dynamically Scalable Web Services", International Journal on Performance evaluation, 2011.
- [46] Zhang and Xuejie Zhang, "A Load Balancing Mechanism Based on Ant Colony and Complex Network Theory in Open Cloud Computing Federation" by research gate, 2nd International Conference on Industrial Mechatronics and Automation, 2010.



This work is licensed under a [Creative Commons Attribution 4.0 International License](https://creativecommons.org/licenses/by/4.0/).

DOI: 10.24297/ijct.v16i7.6440