



## ENHANCED CREDIT BASED LOAD BALANCING IN CLOUD ENVIRONMENT

Jagdeep Singh <sup>(1)</sup>, Mr. Pawan Luthra <sup>(2)</sup>

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

jagdeep9417@gmail.com

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

pawanluthra81@gmail.com

### ABSTRACT

Cloud computing is one of the latest and upcoming paradigm that offers huge benefits such as reduced time to market, unlimited computing power and flexible computing capabilities. It is a model that provides an on-demand network access to a shared pool of computing resources. It comprises a large number of concepts primarily Load Balancing, Scheduling, etc. This paper discusses load balancing as a mechanism to distribute the workload evenly to all nodes in the system to achieve a higher resource utilization and user satisfaction. It helps in allocation and de-allocation of instances of applications without failure. This paper reports a new load balancing technique using modified credit based system using task length, task priority and its cost. The proposed algorithm has been implemented in cloudsims toolkit and its comparison with existing algorithm has been discussed in the paper.

### Keywords

Cloud Computing, Load Balancing, Virtual Machine, Data Center, Data Center Broker.

### INTRODUCTION

Cloud computing is a term we hear quite often, but there are very few people who understand what it's all about. You'd argued that whatever the technology this is, it's probably out of your world or too complex. In reality, Cloud Computing is a simple technology that is being used by us for while, without even knowing what it is [Yubo Tan, Xinlei 2012]. In simple terms, cloud computing entails running computer network applications that are on other person's servers using a simple user interface or application format. It's that simple. If this language still sounds strange, going back to basics will tell you about what cloud computing is all about. In old days of the networking, before the companies like Google and Yahoo were born, companies ran e-mails as an application whose data was stored in a safe, dark room on the company's premises. This sound familiar therefore you were not allowed to enter that room due to security concerns. Moving forward into 20th century; when companies like Google started showing up, the way email was treated and utilized and revolutionized. It would have been a commercial bid to get more and more subscribers, but these companies decided to open their servers to store email data for you, free of charge. However to access that data, you have to use their applications like Gmail, Yahoo Mail and so many others. Practically, this is what cloud computing is all about using other people's servers to run apps for your organization, remotely. Into the 21st century, the concept of the cloud is the same, but more than ever before, Cloud computing is bigger. It's now becoming possible to use bigger apps that will meet your business goals and functions easily in the cloud e.g. with cloud computing, you can run all your computer networks and programs as a whole without ever buying an extra piece of hardware or software. The Cloud technology has these advantages:- 1st Companies can save a lot of money, 2nd they are able to avoid the mishaps of the regular server protocols. For instance, when a company decides to have a new piece of software, whose license can only be used once and it's pretty expensive, they would not have to buy software for each new computer that's added to the network. Instead, they could use the app installed on a virtual server somewhere and share it in the cloud. These capabilities are going to be even more personalized today and there are such a few solutions that permit the usage of mobile in the cloud. Of course there are a few people who are not willing to lose control of the little physical tools they are used to: However, largely any business that considers cutting costs and wants to move forward in this new age needs to embrace the cloud computing basics or at least give it a shot to survive. In simple terms, using Cloud computing means storing your data on a place that is not there on your local hard drive e.g. if you are at work and you want to use a file that was not on your PC, you had to carry floppy disk or CD or USB sticks, Zip drives which is inconvenient. But now by storing the data file on cloud service like Google Drive, it does not matter where you are, you can access that file. You also don't have to take tension about backups because the cloud will do that for you.

Data in cloud computing is of three types. First type data is the data which is transmitted. Second type of data is storage data. And third type of data is processing data. Cloud computing is more important because of device and location independency. Cloud computing is more secure and more reliable. We can add or remove users and also resources in cloud computing. Cloud computing brings everything on cloud. Example whenever you travel by a bus or train, you need to take a ticket for your destination and hold back to your seat till you reach the destination. Likewise other passengers also take tickets and travel in the same bus with you and it hardly bothers you where they are. When your destination comes, you get out of the bus thanking the driver. Cloud computing is just like that bus, carrying data and information for different users and allows to use its service with minimum cost.

Examples of current cloud apps:

- **Software as a service:** Refers to apps delivered to the end user through a web browser or any other web rich client e.g. includes MS Office Live, Drop Box etc.

- **Platform as a service:** provides more customization room, for e.g.; for a developer to acquire a platform such as O.S, that is used to carry out a very specific task for example Google App Engine.
- **Infrastructure as a service:** provide maximum control where a computing infrastructure can be assembled from the O.S. upward for example Amazo, EC 2, IBM cloud burst.

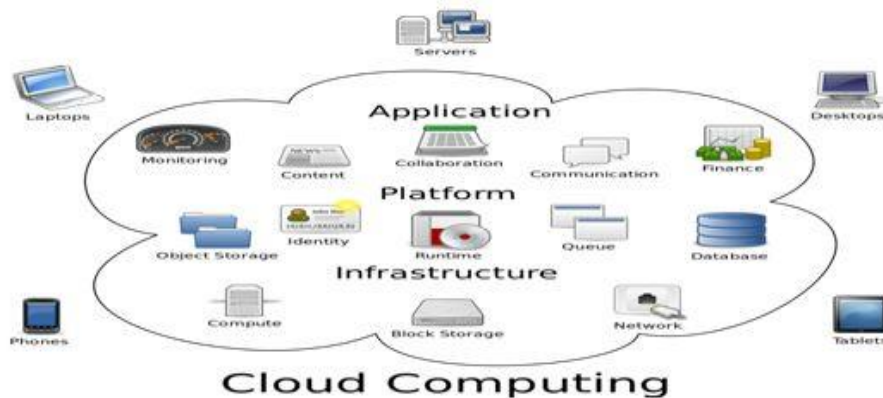


Figure 1: Cloud Computing Services

## RELATED WORK

Most of researchers are operating within the space of load equalization in cloud computing for enhancing the general performances of the clouds. A number of these works contain the improved ancient algorithms to realize load equalization. Therefore on appreciate their contribution and higher perceive the work ahead.

Al-Rayis et al. [1] explains that primarily, load balancers will be deployed supported 3 totally different architectures. The centralized load equalization design which incorporates a central load balancer to create the choice for the complete system relating to that cloud resource ought to take what work and supported that algorithm(s). Within the class-conscious load equalization design, a main load balancer (parent) receives all job requests, then it spreads them to alternative connected load balancers (children) wherever every load balancer within the tree could use a unique rule. The special characteristics of cloud environments that result from the complexities of cloud computing virtual infrastructure need advanced load equalization solutions that are capable of dynamically adapting the cloud platform whereas providing continuous service and performance guarantees. These difficulties have created totally different views on that load equalization design would higher suites cloud computing. Bhoi et al. [2] mentioned that in increased Max-Min Task programming rule in cloud computing helps in supply a high performance computing supported protocols that allowed shared computation and storage over long distances. It depends upon expected execution time rather than completion time. Max-Min rule assign task with most execution time to resource produces minimum completion time whereas increased Max-min assign task with average execution time to resource produces minimum execution time. Bhadani et al. [3] projected a Central Load equalization Policy for Virtual Machines (CLBVM) that balances the load equally in an exceedingly distributed virtual machine/cloud computing surroundings. Bendiab et al. [4] introduced the Map Reduced primarily based Entity Resolution load equalization technique in networking that relies on massive datasets. During this technique, 2 main tasks are done: Map task and scale back task that the author has delineated. For mapping task, the half methodology is dead wherever the request entity is partitioned off into components. Then COMP methodology is employed to check the components and at last similar entities are sorted by cluster methodology and by exploitation scale back task. Map task reads the entities in parallel and method them, so overloading of the task is reduced.

Birattari et al. [5] projected troubleshoot of load balance in Cloud computing exploitation random Hill ascension. Load balancing is a computer network method for distributing workloads across multiple computing resources, for example computers, a computer cluster, network links, central processing units or disk drives. Load balancing plans to optimize resource use, maximize throughput, minimize response time, and evade overload of any one of the resources. Buzato et al. [6] projected Bee Life rule that was used for programming in Cloud computing. Bee Life rule is galvanized by the behavior and copy of bee to search out food supply. The rule evaluated the performance of the resources and it's the aim to scale back time and quality of labor. Babu et al. [7] projected a Honey Bee Behavior galvanized Load equalization [HBB-LB] technique that helps to realize even load equalization across virtual machine to maximize output. It considers the priority of task waiting in queue for execution in virtual machines. Then work load on VM calculated decides whether or not the system is full, beneath loaded or balanced. And supported this VMs are sorted. New in keeping with load on VM the task is scheduled on VMs. Task that is removed earlier. To search out the proper low loaded VM for current task, tasks that are removed earlier from over loaded VM are useful. Forager bee is employed as a Scout bee within the next steps.

Dorigo et al. [8] has projected a load equalization technique referred to as colony of cooperating agents in ants supported soft computing for resolution the improvement downside. This method solves the matter with high likelihood. It's an easy loop occupancy direction of accelerating worth that is uphill. And this build minor modification in to original assignment in keeping with some criteria. Deldari et al. [9] projected a unique load equalization rule referred to as VectorDot in intelligent ants. It handles the class-conscious quality of the datacenter and multidimensionality of resource hundreds across

servers, network switches, associate degree storage in an agile knowledge center that has integrated server and storage virtualization technologies. Desai et al. [10] discusses regarding the rising technology i.e. a brand new normal of enormous scale distributed computing and parallel computing. It provides shared resources, info or alternative resources as per clients' needs at specific times. For higher management of obtainable smart load equalization techniques are needed. And thru higher load equalization in cloud, performance is redoubled and user gets higher services. Therefore during this author has mentioned many alternative load equalization techniques won't to solve the problem in cloud computing surroundings. Elzeki et al. [11] mentioned in Improved Max-Min rule in Cloud Computing that focuses on the cloud computing that additional deals with the allocation of the tasks to the resources whereas observant totally different parameters like waiting time, Average waiting time, turn time, process value. So, associate degree rule named as Max-Min in improved manner from load equalization has been shown to beat such sorts of issues. The rule calculates the expected completion time of the submitted tasks on every resource. Then the task with the general most expected execution time is appointed to a resource that has the minimum overall completion time. Fahringer et al. [12] introduced a static load reconciliation technique referred to as ant Colony optimization. During this technique, AN ant starts the movement because the request is initiated. This system uses the Ants behavior to gather info of cloud node to assign task to the actual node. During this technique, once the request is initiated, the ant and also the secretion starts the forward movement within the pathway from the "head" node. The ant moves in forward direction from A full node searching for next node to ascertain whether or not it's A full node or not. Currently if ant notice below loaded node still it move in forward direction within the path. And if it finds the full node then it starts the backward movement to the last below loaded node it found antecedently. Within the rule if ant found the target node, ant can kill in order that it'll forestall supererogatory backward movement. Fang et al. [13] mentioned a two-level task programing mechanism supported load reconciliation to fulfill dynamic necessities of users and procure high resource utilization. It achieves load reconciliation by 1st mapping tasks to virtual machines so virtual machines to host resources thereby raising the task latent period, resource utilization and overall performance of the cloud computing setting. Gellerb et al. [14] introduced a static well-known load reconciliation technique referred to as spherical Robin, during which all processes are divided amid all obtainable processors. The allocation order of processes is maintained domestically that is freelance of the allocation from the remote processor. During this technique, the request is distributed to the node having least variety of connections, and since of this at some purpose of your time, some node is also heavily loaded and different stay idle. This downside is solved by CLBDM (Central Load reconciliation call Model). Guo et al. [15] given a spotlight on energy potency in datacenter by considering the task programing to physical server and reducing energy within the system. The criterion accustomed live the performance of AN energy loss was to arrange laptop exceeds the need of the duty. Experiment methods for allocating server to a sequence of jobs were a largest machine 1st heuristic, a best work technique, and a mixed technique. The experiment results indicated that each one 3 algorithms waste less energy consumption in over-provision. Hu et al. [16] planned a programing strategy on load reconciliation of VM resources that uses historical knowledge and current state of the system. This strategy achieves the simplest load reconciliation and reduced dynamic migration by employing a genetic rule.

## RESEARCH GAP

Cloud computing so involving distributed technologies to satisfy a spread of applications and user desires. Sharing resources, software and applications are the most functions of cloud computing that are having an objective to reduce the overall cost like the capital cost and operational costs attached with them. Moreover performance in terms of processing time, execution time, turnaround time and the waiting time should be achieved considerably. So there are various measures and numerous technical challenges that has to be addressed like fault tolerance, virtual machine migration, high accessibility, server consolidation and measurability. However central issue is that the load equalization, it's the mechanism of distributing the load among numerous nodes of a distributed system to boost each resource utilization and job time interval whereas additionally avoiding a state of affairs wherever a number of the nodes square measure heavily loaded whereas alternative nodes square measure idle or doing little or no work. It additionally ensures each one) the processor within the system or every node within the network will about the equal quantity of labor at any instant of your time. Load balancing [5] is completed with the assistance of load balancers wherever every incoming request is redirected and is clear to consumer who makes the request. Supported preset parameters, reminiscent of accessibility or current load, the load balancer uses numerous planning rule to work out that server ought to handle and forwards the request on to the chosen server.

The random arrival of load in such an environment can cause some server to be heavily loaded while other server is idle or only lightly loaded. Equally load distributing improves performance by transferring load from heavily loaded server. Efficient scheduling and resource allocation is a critical characteristic of cloud computing based on which the performance of the system is estimated. The considered characteristics have an impact on cost optimization, which can be obtained by improved response time and processing time.

- The capacity of the VM should be considered before allocating the cloudlet to the virtual machine. There can be a scenario where a cloudlet with high credit can be assigned to a VM of lower capacity.
- The existing papers specify that the system will work only in homogeneous environment where all the virtual machines will carry the similar configurations.
- No sorting mechanism is applied on the virtual machines.
- Current load of Virtual machine is not computed before allocating the request to it.
- Resource specific demands of the each task has not been considered.

## OBJECTIVES

- To study the performance of existing load balancing algorithm.
- To propose a new efficient credit based load balancing at both the sides.
- To arrange the virtual machines according to their capacities (MIPS)
- To check the current load of the virtual machine before allocating the task to it.
- To compute the credit of the cloudlet based on task length, priority and cost.
- To implement the proposed algorithm in cloudsim simulator and evaluate the performance of proposed algorithm with existing algorithm.

## RESEARCH METHODOLOGY

The main objective of this paper was to answer the question: in identical cloud environments, which load balancing architecture: centralized, decentralized or hierarchical architecture will give the best results in terms of response time and server load

To answer this question a robust evaluation framework [7] was implemented which includes the following steps:

- To balance the load equally among different VMs.
- Fetch all the available virtual machines in the datacenter/host.
- Retrieve the processing capacity of the available virtual machines.
- Retrieve all the cloudlets and fetch the instruction size, priority and cost of all the cloudlets/tasks.
- Compute the sum of the instruction lengths of all the cloudlets and find the average instruction length.
- Find the TLD (Task Length Difference) of all the cloudlets by taking the absolute of the difference of TLD and cloudlet instruction length.
- After finding the difference in task lengths of each task, credits are assigned to each task. In this algorithm there are 5 credits and these credits are given to each tasks for different conditions. Before these steps, 4 different values are found from the length array. These 4 values forms the condition for assigning the credits. We can't simply choose 4 values. These values should be in a range of task length. The computations are given below.

$$\begin{aligned} \text{value\_1} &= \text{high\_len}/5 \\ \text{value\_2} &= \text{high\_len}/4 \\ \text{value\_3} &= \text{value\_2} + \text{value\_1} \\ \text{value\_4} &= \text{value\_3} + \text{value\_2} \end{aligned}$$

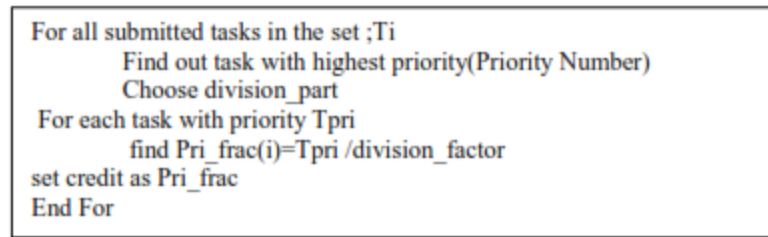
```
For all submitted tasks in the set ;Ti
  |  $TLD_i = \text{len}_{\text{avg}} - \text{Tlen}_i$  |
  If  $TLD_i \leq \text{value\_1}$ 
    then credit =5
  else if  $\text{value\_1} < TLD_i \leq \text{value\_2}$ 
    then credit =4
  else if  $\text{value\_2} < TLD_i \leq \text{value\_3}$ 
    then credit =3
  else if  $\text{value\_3} < TLD_i \leq \text{value\_4}$ 
    then credit =2
  else  $\text{value\_4} > TLD_i$ 
    then credit =1
End For
```

Figure 2. Credit system based on Task Length

### Task priority credit

Task priority is also important for scheduling tasks. Each task may have different priorities, which are represented as values assigned to each task and the value can be the same for more than one task. The scheduling algorithm based on task priority has the problem of treating tasks with similar priority. In the proposed approach this does not arise as a problem because even though we are giving credits to each tasks based on their priority, the final scheduling will be

based on total credit which is based on task length, priority and cost. The primary step in the above algorithm is finding the highest priority number. Second step in the algorithm is choosing the division factor for finding Pri\_frac for each task. For example if highest value of priority is a two digit number then choose division\_part as 100. If it is 3 digit then division\_part is 1000. Third step in the algorithm is calculating the Pri\_frac for each task. This can be calculated by dividing priority value of each task with division factor of corresponding task. Finally this value(Pri\_frac) will be assigned to each task as priority credit.



**Figure 3. Credit system based on Task Priority**

#### Cost priority credit

- Compute the sum of cost of all the cloudlets and find the average cost of all the cloudlets. Find out the percentage cost of each cloudlet by using the below mentioned equation.

$$Cost_{Credit} = \frac{Cloudlet\_Cost}{Total\ Cost\ of\ all\ the\ Cloudlets}$$

All the credits are calculated separately and the final credit of the cloudlet is computed by using the product of all the credits i.e. length credit, cost credit and priority credit.

$$Total_{Credit} = Length_{Credit} * Priority_{Credit} * Cost_{Credit}$$

## SIMULATION IN CLOUD: CLOUDSIM

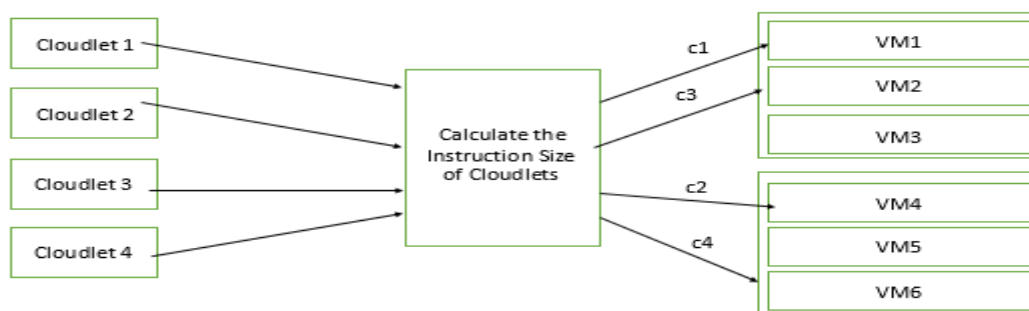
In CloudSim, cloud computing infrastructures and application services allowing its users to focus on specific system design issues that they want to investigate [8]. Simulation in a CloudSim means implementation of actual environment towards benefit of research. The users or researcher actually analyse the proposed design or existing algorithms through simulation. Resources and software are shared on the basis of client's demand in cloud environment. Essentially, dynamic utilization of resources is achieved under different conditions with various previous established policies. Sometime it is very much difficult and time consuming to measure performance of the applications in real cloud environment. In this consequence, simulation is very much helpful to allow users or developers with practical feedback in spite of having real environment. In this research work, simulation is carried out with a specific cloud simulator, CloudSim [7].

**Data centre:** Data centre encompasses a number of hosts in homogeneous or heterogeneous configurations (memory, cores, capacity, and storage). It also creates the bandwidth, memory, and storage devices allocation.

**Virtual Machine (VM):** VM characteristics comprise of memory, processor, storage, and VM scheduling policy. Multiple VM can run on single hosts simultaneously and maintain processor sharing policies.

**Host:** This experiment considers VM need to handle a number of cores to be processed and host should have resource allocation policy to distribute them in these VMs. So host can arrange sufficient memory and bandwidth to the process elements to execute them inside VM. Host is also responsible for creation and destruction of VMs.

**Cloudlet:** Cloudlet is an application component which is responsible to deliver the data in the cloud service model. So the length, and output file sizes parameter of Cloudlet should be greater than or equal to 1. It also contains various ids for data transfer and application hosting policy.



**Figure 4. Assignment of Cloudlets to Virtual Machines**

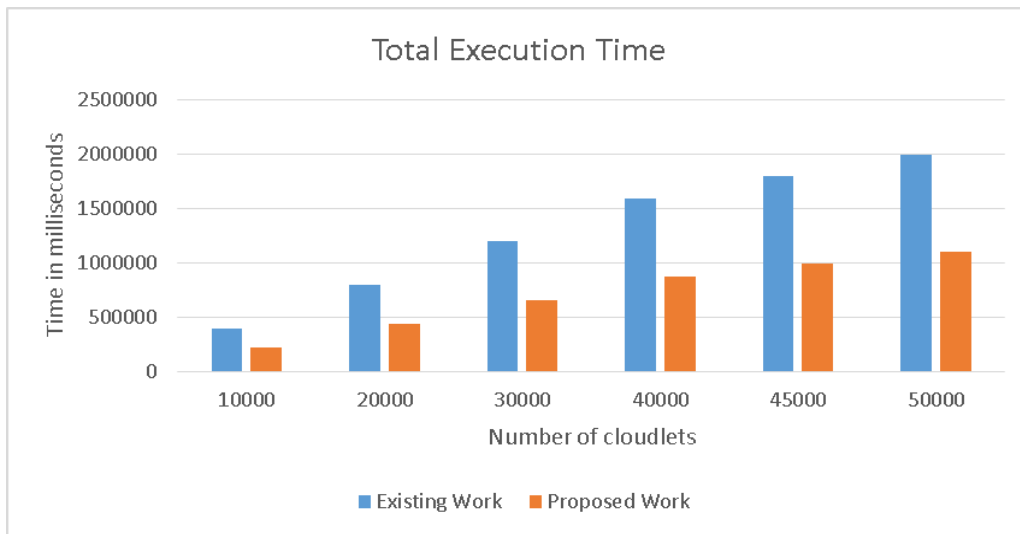
## EXPERIMENTAL RESULTS

For the simulation, we focus on average waiting time, the total processing time, and processing cost. (1) The waiting time and total finishing time is computed in milliseconds. CloudSim is a new generation and extensible simulation platform which enables seamless modeling, simulation, and experimentation of emerging Cloud computing infrastructures and management services to be. CloudSim is used to verify the correctness of the proposed algorithm. CloudSim toolkit is used to simulate heterogeneous resource environment and the communication environment. The experiments are conducted several number of times by taking different number of cloudlets like 1000,2000,3000 and so on. The average waiting time, total processing time, processing cost have been computed as below in Table 1.

**Table 1. Experimental Results for different number of Cloudlets**

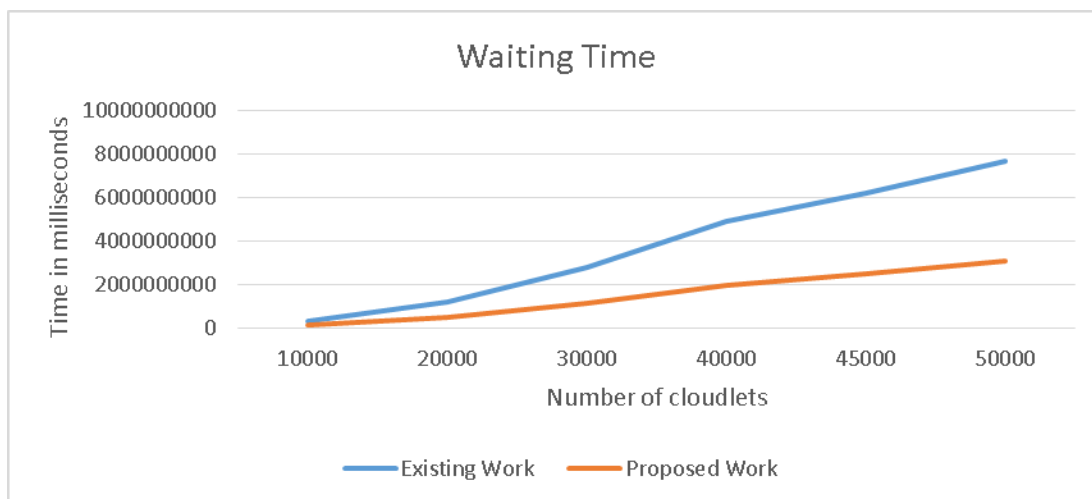
| No of Cloudlets | Total Turn Around Time | Total Execution Time | Total Waiting Time | Total Processing Cost |
|-----------------|------------------------|----------------------|--------------------|-----------------------|
| 5               | 51                     | 50                   | 1                  | 153                   |
| 7               | 81                     | 80                   | 1                  | 245                   |
| 10              | 197                    | 196                  | 1                  | 598                   |
| 20              | 681                    | 421                  | 260                | 1286                  |
| 40              | 2397                   | 868                  | 1529               | 2649                  |
| 60              | 5102                   | 1318                 | 3784               | 4022                  |
| 80              | 8710                   | 1739                 | 6971               | 5307                  |
| 100             | 13409                  | 2187                 | 11222              | 6671                  |
| 150             | 29413                  | 3287                 | 26125              | 10030                 |
| 175             | 38459                  | 3777                 | 34682              | 11523                 |
| 200             | 51395                  | 4376                 | 47019              | 13351                 |
| 300             | 114280                 | 6591                 | 107689             | 20110                 |
| 400             | 201600                 | 8777                 | 192823             | 26779                 |
| 450             | 254682                 | 9878                 | 244804             | 30138                 |
| 500             | 313460                 | 10967                | 302493             | 33461                 |
| 700             | 611703                 | 15368                | 596335             | 46888                 |
| 1000            | 1243681                | 21959                | 1221722            | 66996                 |
| 1500            | 2790349                | 32954                | 2757395            | 100542                |
| 2000            | 4951918                | 43919                | 4907999            | 133997                |
| 3000            | 11128156               | 65908                | 11062247           | 201085                |
| 4000            | 19766333               | 87861                | 19678472           | 268064                |
| 5000            | 30869084               | 109821               | 30759263           | 335064                |
| 7000            | 60476696               | 153766               | 60322931           | 469139                |
| 7500            | 69419559               | 164758               | 69254801           | 502675                |
| 8000            | 78973550               | 175723               | 78797827           | 536132                |
| 9000            | 99944019               | 197709               | 99746310           | 603210                |
| 9500            | 111345350              | 208674               | 111136677          | 636663                |
| 10000           | 123372341              | 219665               | 123152675          | 670199                |
| 20000           | 493259545              | 439330               | 492820214          | 1340397               |
| 30000           | 1109690924             | 659024               | 1109031900         | 2010683               |
| 40000           | 1972604577             | 878678               | 1971725899         | 2680845               |

**Processing Time/ Execution Time:** The time from the submission of a request to the time of first response by the cpu. It is the amount of time it takes to get first response for the request submitted. Figure 5 depicts the total processing time / execution time for the base algorithm and proposed algorithm for the given number of cloudlets.



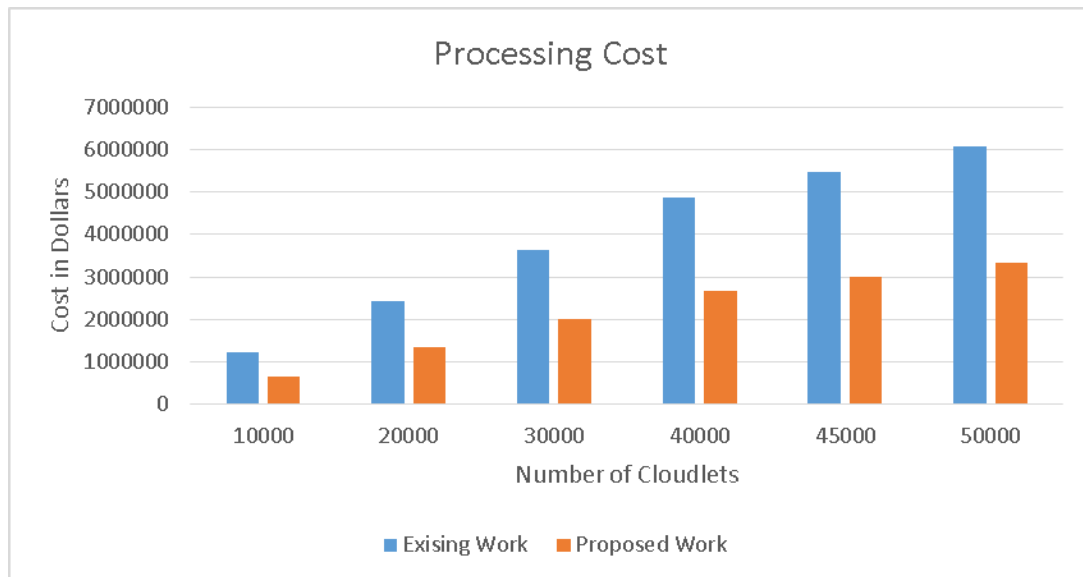
**Figure 5. Execution Time for existing work and proposed algorithm.**

**Waiting Time:** Waiting time is the amount of time that is taken by a process in ready queue. Average waiting time is the time for which the cloudlets were kept in the queue for their execution.



**Figure 6. Waiting time for existing work and proposed algorithm.**

The processing cost is calculated based on the actual CPU time when the tasks are accomplished to execute on the cloud resources and the cost of resources per second. From the figure 7, it is clear that the processing cost occurred to the client is lesser in the proposed algorithm than in the base algorithm. The processing cost is the main factor which has been occurred to the client. Lesser the processing cost, more will be the benefit to the client and thereby increasing the overall client's weightage at the cloud provider. The experiment has been conducted on multiple cloudlets and on multiple VM's which depict that the proposed algorithm has been working consistently on any type of configuration and parameters.



**Figure 7. Processing cost for existing work and proposed algorithm.**

## CONCLUSION

In this paper, a new enhanced and efficient Load balancing algorithm is proposed and then implemented in cloud computing environment using cloudSim toolkit, in java language. The research work involved developing an efficient VM load balancing algorithm for the cloud and conducting a comparative analysis of the proposed algorithm with the existing algorithms on identified parameters. By visualizing the cited parameters in graphs and tables we can easily identify that overall processing time, processing cost and waiting time is improved in comparison to the existing scheduling parameters.

## REFERENCES

- [1] Wu, H.-S., Wang, C.-J., and Xie, J.-Y. (2013a). Terascaler elb-an algorithm of predictionbased elastic load balancing resource management in cloud computing. In *Advanced Information Networking and Applications Workshops (WAINA), 2013 27th International Conference on*, pages 649-654. IEEE.
- [2] Wu, X., Deng, M., Zhang, R., Zeng, B., and Zhou, S. (2013b). A task scheduling algorithm based on qos-driven in cloud computing. *Procedia Computer Science*, 17:1162-1169.
- [3] Sharma, A. and Peddoju, S. K. (2014). Response time based load balancing in cloud computing. In *Control, Instrumentation, Communication and Computational Technologies (ICCICCT), 2014 International Conference on*, pages 1287-1293. IEEE.
- [4] Ren, H., Lan, Y., and Yin, C. (2012). The load balancing algorithm in cloud computing environment. In *Computer Science and Network Technology (ICCSNT), 2012 2nd International Conference on*, pages 925-928. IEEE.
- [5] Raju, R., Amudhavel, J., Kannan, N., and Monisha, M. (2014). A bio inspired energy-aware multi objective chiropteran algorithm (eamoca) for hybrid cloud computing environment. In *Green Computing Communication and Electrical Engineering (ICGCCEE), 2014 International Conference on*, pages 1-5. IEEE.
- [6] Mesbahi, M., Rahmani, A. M., and Chronopoulos, A. T. (2014). Cloud light weight: A new solution for load balancing in cloud computing. In *Data Science & Engineering (ICDSE), 2014 International Conference on*, pages 44-50. IEEE.
- [7] Domanal, S. G. and Reddy, G. R. M. (2013). Load balancing in cloud computing using modified throttled algorithm. In *Cloud Computing in Emerging Markets (CCEM), 2013 IEEE International Conference on*, pages 1-5. IEEE.
- [8] Domanal, S. G. and Reddy, G. R. M. (2014). Optimal load balancing in cloud computing by efficient utilization of virtual machines. In *Communication Systems and Networks (COMSNETS), 2014 Sixth International Conference on*, pages 1-4. IEEE.
- [9] Delavar, A. G. and Aryan, Y. (2014). Hsga: a hybrid heuristic algorithm for work flow scheduling in cloud systems. *Cluster computing*, 17(1):129-137.
- [10] O.M. Elzeki, "Improved Max-Min Algorithm in Cloud Computing". *International Journal of Computer Applications* (0975-8887) Volume 50-No.12, July 2012.
- [11] Vishwas Bagwaiya, "A HYBRID APPROACH USING THROTTLED AND ESCE LOAD BALANCING ALGORITHMS IN CLOUD COMPUTING".
- [12] Amandeep Kaur Sidhu. "Analysis of load balancing techniques in cloud computing". *International Journal of computers & technology* volume 4 No. 2, March-April, 2013, ISSN 2277-3061.





- [13] Ektemal Al-Rayis. "Performance Analysis of load balancing Architectures in Cloud computing" 2013 European Modeling Symposium. 978-1-4799-2578-0/13\$31.00@2013 IEEE.
- [14] Haozheng Ren, "The load balancing Algorithm in cloud computing Environment" 2nd International Conference on computer science and network technology, 2012.
- [15] Tushar Desai. "A survey of various load balancing techniques and challenges in cloud computing" International Journals of scientific and technology research volume 2, Issue11,Nov2013.
- [16] Upendra Bhoi. "Enhanced max-min Task scheduling Algorithm in cloud computing". International Journal of Application or Innovation in Engineering & management(IJAIEM), April 2013.
- [17] Klaithem Al Nuaimi, "A survey of load balancing in cloud computing challenges and algorithm". 2012 IEEE second symposium on network cloud computing and applications.
- [18] Gytis Vlutis, "Model of load balancing and scheduling in cloud computing". Proceedings of the ITI 2012 34th Int. Conf. on Information Technology Interfaces, June 25-28,Cavat,Croatia.
- [19] L.D.Dinesh Babu<sup>a</sup>. "Honey Bee Behaviour inspired load balancing of tasks in cloud computing environments, Jan 30,2013.
- [20] Kumar Nishant, " Load Balancing of Nodes in Cloud Using Ant Colony Optimization". 2012 14th International Conference on Modelling and Simulation.
- [21] Mr. M. Ajit, " VM Level Load Balancing in Cloud Environment", IEEE – 31661.



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